Artificial Intelligence for Engineering Design, Analysis and Manufacturing

cambridge.org/aie

Graph models for engineering design: Model encoding, and fidelity evaluation based on dataset and other sources of knowledge

Eric Coatanéa[1], Hari Nagarajan[1], Hossein Mokhtarian[2], Di Wu[1], Suraj Panicker[1], Andrés Morales-Forero[2] and Samuel Bassetto[2]

[1]Manufacturing Systems Research Group, Faculty of Engineering and Natural Sciences, Tampere University, Tampere, Finland and [2]Mathematics and Industrial Engineering (MAGI), Université de Montréal (Polytechnique), Montreal, Canada

Research Article

Cite this article: Coatanéa E, Nagarajan H, Mokhtarian H, Wu D, Panicker S, Morales-Forero A, Bassetto S (2023). Graph models for engineering design: Model encoding, and fidelity evaluation based on dataset and other sources of knowledge. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **37**, e6, 1–23. https://doi.org/10.1017/S0890060422000269

Received: 6 April 2022
Revised: 7 October 2022
Accepted: 16 December 2022

Key words:
Causality; dimensional analysis conceptual modeling (DACM); machine learning; model fidelity; oriented graphs

Author for correspondence:
Eric Coatanéa,
E-mail: eric.coatanea@tuni.fi

Abstract

Automatically extracting knowledge from small datasets with a valid causal ordering is a challenge for current state-of-the-art methods in machine learning. Extracting other type of knowledge is important but challenging for multiple engineering fields where data are scarce and difficult to collect. This research aims to address this problem by presenting a machine learning-based modeling framework leveraging the knowledge available in fundamental units of the variables recorded from data samples, to develop parsimonious, explainable, and graph-based simulation models during the early design stages. The developed approach is exemplified using an engineering design case study of a spherical body moving in a fluid. For the system of interest, two types of intricated models are generated by (1) using an automated selection of variables from datasets and (2) combining the automated extraction with supplementary knowledge about functions and dimensional homogeneity associated with the variables of the system. The effect of design, data, model, and simulation specifications on model fidelity are investigated. The study discusses the interrelationships between fidelity levels, variables, functions, and the available knowledge. The research contributes to the development of a fidelity measurement theory by presenting the premises of a standardized, modeling approach for transforming data into measurable level of fidelities for the produced models. This research shows that structured model building with a focus on model fidelity can support early design reasoning and decision making using for example the dimensional analysis conceptual modeling (DACM) framework.

Introduction

In data-driven engineering design, the production of data across a product or system's life cycle is becoming abundant (McAfee and Brynjolfsson, 2012). Data are generated from multiple sources, and it is becoming a common design practice to integrate sensors in products as well as product manufacturing. Simultaneously, with the advances in data analytics, the practices followed at different stages of product design are undergoing a transition. The discipline relies extensively on simulations to replace expensive prototyping. Simulation-based design evaluation is increasingly used when the cost of prototyping complex systems is high (Li *et al.*, 2012). However, simulations may become tedious and expensive when a large design space needs to be evaluated (Choi *et al.*, 2014). In that context, generating metamodels is often a suitable alternative. Metamodels are simplified models representing the essence of complex phenomena or complex simulation models (Bhosekar and Ierapetritou, 2018). In a production environment, it is advantageous to produce metamodels using machine learning methods because of its automation capabilities (Cai *et al.*, n.d.; Sun and Wang, 2019).

In data-driven design, the product development process should benefit from tools and approaches that can extract and integrate useful knowledge embedded in the data to produce models (Berman, 2013; Brunton and Kutz, 2019; Lee *et al.*, 2022). It will be beneficial for developed models to support reasoning through simulation. Simulation-based reasoning is a significant contribution to the early decision-making process (Felsberger *et al.*, 2016). The decision-making process relies on human judgment and its global acceptance in any organization requires the production of rationales supporting the decisions (Chaudhuri *et al.*, 2011). Models should facilitate the human cognitive processes (Miller, 1956; Dadi *et al.*, 2014). In human cognition, a natural tendency for humans is to develop a causal interpretation of events. Such causal interpretations can be valid or erroneous, but they represent a strong human cognitive pattern. Any tools developed to support decision making should also consider the human characteristic. Approaches such as graph-based representations and causal ordering for parsimonious models produced can facilitate the model's transparency and enable

© The Author(s), 2023. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (http://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

efficient human decision-making by being aligned with the human cognition (Goel *et al.*, 2012; Shekhawat *et al.*, 2021).

The current research aims to leverage the potential of machine learning for the extraction and integration of knowledge gathered from different data sources in the development of engineering models matching human cognition and supporting automatic reasoning. In general terms, knowledge can be defined as awareness of facts or as practical skills. It may also refer to familiarity with objects or situations. Knowledge of facts is distinct from opinion because it benefits from justification provided through rationales. Such rationales are produced using a representation of the knowledge or the structure of knowledge which can be easily shared and understood. This research exploits two sources of knowledge, namely, data collected from a system and the fundamental units of the variables of the system. The article presents a two-part method for generating a structure of knowledge. First, oriented graphs representing the variables of the dataset as nodes and their interrelations as edges are developed. Second, a power-law representation is developed for the variables of the system. The power laws and the oriented graphs are composed together to construct the structure of knowledge wherein, the exponents of the power laws being associated with the nodes (i.e., the variables) and the multiplicative connections between variables are associated with the edges (i.e., variable interrelationships).

This research aims to extend the scope of the pre-existing dimensional analysis conceptual modeling framework (DACM) to develop oriented graphs. The goal is to exploit datasets as a new form of inputs for the framework. The DACM framework has been initially developed for generating oriented graphs combined with power-law equations using functional representation as a starting point for modeling of systems (Mokhtarian *et al.*, 2017). Initially, it was developed for bringing simulation to the very early stage of the engineering design process. In the framework, a mathematical machinery based on qualitative reasoning using power laws has been implemented to incorporate qualitative objectives in the developed models (Bhaskar and Nigam, 1990). The DACM propagation algorithm can propagate qualitative objectives into oriented graphs and detect physical contradictions as defined in theory of inventive problem solving (TRIZ) (Savransky, 2000). This is allowing the detection of design weaknesses at the early design conceptual stage (Eckert *et al.* 1999; Shahin, 2008; Jin and Benami, 2010). TRIZ separation principle (Savransky, 2000) is used to modify the topology of the oriented graphs and consequently generate innovative solutions minimizing or removing the contradictions for ultimately creating better designs (Kryssanov *et al.*, 2001; Mayda and Börklü, 2014). However, to be able to generate a valid reasoning with this approach, a pre-requisite is to build the oriented graph to accurately represent the causal relationships in the system of interest. In the case of the exploitation of a dataset as the initial starting knowledge, classical machine learning methods cannot guarantee the validity of the causal ordering. Thus, the proposed method enables the development of oriented graphs with an accurate causal ordering of variables which can then be used in tandem with the DACM framework for qualitative reasoning.

The integration and exploitation of data in the DACM framework can provide a standardized way to approach early design reasoning; however, the usability of the models may be affected by the model fidelity. The objectives of the model and the designer, and relevant applications must be considered when developing engineering models (Ponnusamy *et al.*, 2014). As mentioned above, a valid causal ordering is a necessary condition for qualitative reasoning using the DACM framework. Similarly, a lack of standardized approaches to integrate fidelity in modeling approaches can have detrimental effects, weakening the usability of the engineering models. Like fields such as metrology, which is constructed around standardized measurement approaches and fundamental concepts, the design of simulation models for engineering purposes, must also be facilitated by formalization and conceptualization to enable fidelity assessments. The problem has been recognized and an effort to formalize the discipline is ongoing in literature (Gross, 1999; Gross *et al.*, 1999; Ponnusamy *et al.*, 2014). In line with existing research, the current article develops a standardized process for machine learning-based modeling to develop simulation models and measure fidelity. The developed models are based on a causal representation of the system variables and can be used for engineering design reasoning and decision making.

The remainder of the article is organized as follows: Section "Background" presents the background on the development of simulation models and evaluation of fidelity. Section "Method" presents the modeling method with the help of a case study. DACM combined with a machine learning module is presented as a model creation and fidelity measurement approach. Section "Case study" presents the findings of this research and the potential applications of the developed approach. Section "Discussion and future work" discusses the conclusions and future development efforts.

Background

In this section, the importance of early decisions in engineering design are emphasized. DACM as a causal ordering framework is briefly introduced. Next, the concept of fidelity in simulation is analyzed and different methods to support the evaluation of some fidelity aspects are presented.

Early design decision support and the role of discovering causal ordering in engineering design and other disciplines

Herbert Simon introduced two concepts relevant for the entire design and manufacturing disciplines. First, the concept of *bounded rationality* was initially introduced in an economic context as a criticism of the classical economic theory and asserting the existence of a purely rational decision-maker, the *homo economicus* (Moon, 2007). Research in fields such as economics or psychology has demonstrated that rationality is often an exception rather than a norm in human decision processes. Hypothesizing a purely rational behavior is a limitative factor to understanding the complexity of the human decision processes and consequently its branch of design decision making. The second important concept is the concept of *satisficing* (Simon, 1956). The term satisficing is a combination of satisfying and suffice. This concept is important for practical reasons. Indeed, the design activity is a problem-solving activity, where solutions to problems are found in form of tangible or non-tangible artifacts. An artifact is an artificially created object. A *satisficing* solution to a design problem is a feasible solution produced within specified time limits and using specified and constrained bounded resources.

When working on solutions, certain stages of design can have a higher influence and induce constraints on the final performances of the designed artifacts. For example, design decisions taken at the early stages of the process are less costly but concurrently can also heavily constrain later decisions. Therefore, the

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

early design stage is a strategic phase of the design activity, but also a challenging one, due to the limited availability of early knowledge combined with a high level of associated uncertainty. Early design decisions combine the potential for high added value and high risk due to the ill-defined nature of the early knowledge. Early knowledge combines characteristics such as fuzziness, scarcity, and qualitative nature. Despite the inherent uncertainty, early knowledge can provide crucial information during product development and the potential of this resource is rarely fully exploited in current design practices.

A central aim of the scientific activity is to develop methods to systematically unveil cause–effect relationships between variables of a problem. This is particularly relevant to the engineering design and manufacturing discipline. The discovery of potential confounding variables has been a major source of investigation (VanderWeele and Shpitser, 2013). In statistics, a confounding variable is a variable that influences both the dependent and independent variables. This is the source of a false association between variables. Confounding is a concept related to cause–effect relationships and this should not be mixed with the concepts of correlations or relations. The existence of confounders provides rationale for instances when a correlation does not always imply a causal relationship. A variety of approaches have been developed to generate causal ordering in literature (Finkbeiner *et al.*, 2015; McCaffrey and Spector, 2018; Bhatt *et al.*, 2021). The method proposed in this article contributes to the body of knowledge in causal ordering using oriented graphs and equations. Despite the success of existing causal ordering approaches to be effective and robust in an engineering context, a limiting factor could be the scope of the analysis that could be performed using the causal ordering. Methods such as bond graph causal ordering (Gawthrop and Smith, 1996) or causal ordering of equations via the direct use of Iwasaki and Simon algorithm (Iwasaki and Simon, 1994) or its extensions (Trave-Massuyes and Pons, 1997), as well as work done in qualitative physics (Bhaskar and Nigam, 1990), are few examples of causal ordering methods belonging to such a classification. Another approach applied more specifically to the social science context is also interesting for this limited literature review. This is a method searching for cause and effect relationships via models using structural equations modeling (SEM) (Pearl, 2000; Spirtes *et al.*, 2000). SEM derives from path analysis which was developed by the biologist S. Wright in the 1920s and has been often applied to analyze causal relations of non-experimental data using an empirical approach. Path analysis extends regression analysis by analyzing simultaneously many endogenous and exogenous variables of a system or problem. Path analysis was combined with factor analysis and latent variables to form the SEM approach to causal analysis. In path analysis, variables can be endogenous or exogenous. Relations between variables can be represented by a one-sided arrow when a relationship exists between variables and a two-sided arrow when a correlation exists between variables as presented in Figure 1. Factor analysis is integrated with latent variables in the SEM approach. Factor analysis is a statistical method used to describe variability among observed, correlated variables with the central idea that the variation of observed variables can also be reflecting the variations of unobserved variables name latent variables. Factor analysis may help to deal with datasets where there are large numbers of observed variables that are thought to reflect a smaller number of underlying/latent variables (Harman, 1976). However, SEM is of confirmatory nature and researchers have first to model the true causal relationships based on other background knowledge before collecting or analyzing data (Goldberger, 1972). This task is difficult, especially at the beginning of a design process because of the lack of background initial knowledge.

Consequently, using SEM alone, it is often difficult to unveil the initial causal structure. Additionally, SEM suffers from another limitation because SEM assumes normality of the statistical distribution and only uses the covariance structure and cannot find causal direction between two highly correlated variables because the models produced will be equivalent using the SEM approach. Nevertheless, the SEM method constitutes the conceptual basis of several algorithms that are successful in the discovery of causal structure, especially in complex systems studied by sciences such as economics and sociology. This has also been used in medicine and more specifically in epidemiology.

The fundamental concepts of formalism are summarized in the following manner by Halpern (2000). The construction of structural equations (SE) models requires three key steps. First, the problem being studied is represented by a finite set of variables, corresponding to the features of the problem. This can also be the function of a system to be designed as a solution for a specific design problem (Mobus, 2022). There are two sorts of variables. Endogenous variables are such that their values are determined by other variables within the model, whereas the values of exogenous variables are determined in a way that is independent of other variables of the system. The structural equations describe the functional dependence of the endogenous variables on other variables (endogenous and exogenous) in the model (Kaplan, 2008).

SEM use graphs to represent relations between observed variables and latent variables. Latent variables are not observed.

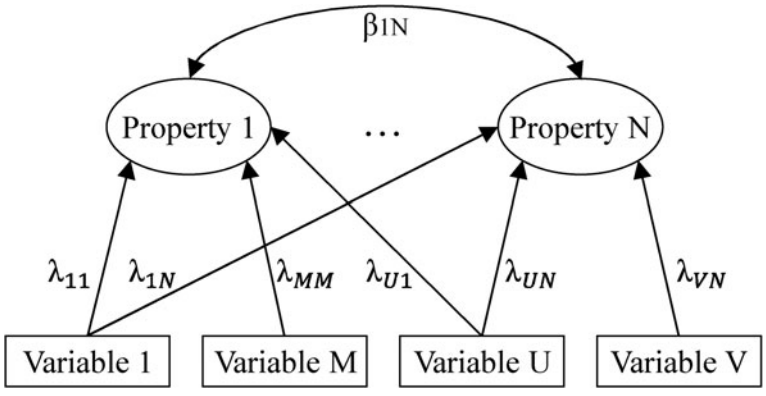


Fig. 1. Example of an SEM model.

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

Latent variables can be also residues or errors in the modeling and measurement process. Software specifically developed for SEM are existing to form the model. An SEM can take the following form presented in Figure 1. Each of the generic situations presented above must form each time a specific model. The β is the element of a correlation matrix. The λ is the coefficient allowing to model quantitatively the different properties in Figure 1.

The work presented in this article is an extension of previously developed framework namely, dimensional analysis conceptual modeling (DACM). DACM was used to build initial causal relationships from the available background knowledge using a framework integrating engineering design concepts such as functions, organs, elementary organs laws, variables' classification, and dimensional analysis theory (DAT) (Coatanéa, 2015).

The DACM framework has been initially developed for generating oriented graphs combined with power-law equations. Currently, DACM uses as a starting modeling point, a functional representation. The framework has been used as a foundation of several publications, including one paper published in AIEDAM (Mokhtarian *et al.*, 2017). Initially, it was developed for developing simulation capabilities at the very early stage of the engineering design process. In the framework, a mathematical machinery based on qualitative reasoning using power laws (Bhaskar and Nigam, 1990) has been implemented to reason via qualitative objectives. The DACM propagation algorithm can propagate qualitative objectives into oriented graphs and detect physical contradictions as defined in TRIZ (Savransky, 2000). Contradictions is a way to detect design weaknesses at the early design conceptual stage. Via the use of TRIZ separation principle (Savransky, 2000), it is possible to modify the topology of the oriented graphs and generate innovative solutions minimizing or removing the contradictions for creating better designs. However, to be able to generate a valid reasoning with this approach, an initial condition needs to be met. In such models, the dimensional homogeneity of models is one property required to support reasoning. A second required property is the right causal ordering of the variables. The DACM method combines some SEM characteristics with dimensional analysis (DA) and qualitative physics and reasoning (Mokhtarian *et al.*, 2017). This is an approach of choice for bringing automatic reasoning capabilities to the early design stages. Causal graphs can find applications in analyzing model fidelity as explained in the next subsection. In the case of the exploitation of a dataset as the initial starting knowledge, classical machine learning methods cannot guaranty the validity of the causal ordering. This is the key purpose of this article to overcome this limitation to ensure the production of a valid causal ordering in a dataset context.

The integration and exploitation of data in the DACM framework open new and interesting research domains but also generate new research difficulties and questions. One among them, is how to evaluate and measure model fidelity?

Models are developed to serve specified purposes and use. The objectives of the model, the designer, and relevant applications must be considered when developing engineering models (Ponnusamy *et al.*, 2014).

Fidelity analysis of models

The process of designing a model, especially for applications in simulation domain includes understanding the type of performances and scenario to be evaluated, choosing a reference, and deciding how many details from this reference should be integrated into the model. The choice of a reference indicates a description of the context of the model and subsequently implies describing the purpose and intended use of the model. Ensuring the above-mentioned features during simulation model design and analysis is necessary to ensure fidelity in simulation results. Fidelity is at the center of what distinguishes computer simulations from other computer programs (Gross *et al.*, 1999). A computer simulation differs from other computer programs because it intends to represent a perceived reality (Horváth and Vroom, 2015). A specification for a simulation is similar to a specification for any other computer program except that a simulation should also describe its objects of interest (performance variables) in the real or perceived world. The accepted level of accuracy for the result of the virtual representation of the performance variables should also be specified. The measure of exactness of the representation in simulation is often referred to as fidelity. To measure fidelity, adequate referencing and quantification are necessary for the modeling process (Ponnusamy *et al.*, 2014).

Using an analogy with metrology, the measurement process is a comparison of an unknown quantity with a known fixed quantity (i.e., a reference) (Ostwald and Muñoz, 2008). A reference is a multifaceted entity that can include a system of fundamental units and their derived units, a specified procedure of measure, an object of interest to be measured, an instrument of measure, and/or a specified environment. All explicit details of an object of interest are not captured by a metrological process but rather only the required features of the object are created for measurement to decide if the object of interest matches the required expectations/specifications. The decision process following a metrological control is universally accepted despite the gap existing between the virtual representation and the real object of interest due to the standardization in methods. Standardization allows for referencing and quantification of fidelity in metrology and can similarly support fidelity in simulation (Roca, 2021). Both computer simulations and metrology processes create virtual representations of the objects of interest. Both simulation and metrology models are the results of interpretations of certain objects of interest. This is also the case for a machine learning (ML) models. Different representations are used to support the decision-making processes. Standardized measurement and specification of fidelity can support the validation of the virtual representations. Fidelity specification standardization initiatives for modeling and simulation have been investigated in literature for fidelity analysis and formalization in literature (Roza *et al.*, 1999; Shephard *et al.*, 2004; Ponnusamy *et al.*, 2014; Roca, 2021).

Ponnusamy *et al.* (2014) present a simulation fidelity assessment framework to enable the verification and validation of simulation products to comply with the specifications and fitness for their intended use. The developed framework mathematically synthesizes abstractions (equations, relations, and numerical entities) consistent with the simulation objectives. The authors assess fidelity by a formal abstraction compatibility criterion between allowable and implemented abstractions. The consistency of abstractions to be compatible with the simulation objectives are developed in the perspective of semantic and syntactic compatibility based on the behavior and structure of the models, respectively. The authors propose that consistent and continuous improvement of simulation products will improve the product development life cycle by keeping the cost and risks in check.

Roca (2021) proposes a formal taxonomy for evaluating fidelity in simulation in the form of a semantic fidelity prism to explore the concept of fidelity beyond arbitrary measures of

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

quality. The author draws on concepts from dimensional analysis, metrology, applied ontology, and other axiomatic formalisms that can define a semantic fidelity taxonomy for classifying quantifiable expressible entities used in computable models. Roca in his work defines fidelity in the simulation context as, "the specification of dimensional, metrological, and mathematical entities and relations that account for relevant axiomatic and hypothetical referent selections of simuland (i.e., things to be simulated) to be encoded into simulation-system, computable models aimed at informing decisions, questions, and/or skill development". The author proposes that the transdisciplinary nature of modeling and simulation needs to be supported by standardized taxonomy promoting semantic fidelity by leveraging tested formalisms such as DA and metrology which are amenable to most engineers. Furthermore, the research proposes that fidelity measurements based on reusable modeling primitives using numerical entities and relations can promote interoperability in applications as opposed to arbitrary and subjective objects measuring accuracy for a single application.

Roza *et al.* (1999) discuss the role of fidelity requirements in simulation and propose a formalized approach for the identification, specification, and validation of fidelity requirements in the simulation development process. The authors develop a guideline for deriving fidelity requirements in an iterative process. The research proposes that fidelity requirements represent the level of realism the simulations must encompass to fulfill the user needs and objectives without which validation and verification approach to measure simulation accuracy may become questionable.

Shephard *et al.* (2004) present the technologies required to enable simulation-based design analysis in CAD and CAE tools. The authors specify that a direct integration of data transfer approaches in CAD tools cannot support the simulation-based design and can affect the accuracy of results during decision making. The authors propose functional components to be integrated into the simulation environment for engineering design such as, simulation model manager, simulation data manager, control tools (static and adaptive), and simulation model generators to carry out specific steps in the process of accurately evaluating design performance parameters (Chakrabarti *et al.*, 2005). The authors suggest that the functional components are required to carry out specific processes in the design evaluation and provide a suitable level of automation and accuracy in decision making.

The studies on fidelity specifications and standardization reiterate the concepts that the development of simulation models must integrate simulation context in the form of fidelity requirements, level of adherence to the real system, and development of references to measure and evaluate fidelity. In addition, the development of a reference that can enable interoperability in applications through numerical entities and relations can support multidisciplinary modeling and simulation. The use of AI and ML presents a unique challenge wherein, the lack of interoperability or explainability of models developed can also hinder the measurement of fidelity. Thus, this research aims to develop a machine learning approach combined with supporting frameworks and methods, such as DACM (Mokhtarian *et al.*, 2017), Singular Value Decomposition (SVD) (Hotelling, 1933), least absolute shrinkage and selection operator (LASSO) (Zou and Hastie, 2005), and ordinary least square regression (OLS) (Goldberger, 1964) to model complex systems in the form of power laws, directly from data. The integration of DA and dimensionless primitives using the Vashy-Buckingham theorem (Szirtes, 2007) in this research, enables the development of reusable modeling primitives for evaluating fidelity and to combine them in the form of a graph model. In the next section, two levels of fidelity at the variable and function level of the system are analyzed, respectively. The proposed approach is explained in the next section with the help of a case study.

Method

The methodology developed in this research introduces a new machine learning approach integrating two levels of fidelity to facilitate the development of engineering simulation models. We hypothesize that interrelationships between variables in a complex system can be modeled using a combination of power laws and oriented graphs. Power laws model proportional relationships between variables are exhibited by general classes of mechanisms present in science and engineering (Sornette, 2006). Power laws can be observed in seemingly unrelated systems. Subsequently, power laws can be used to identify the underlying natural phenomena occurring in a system of interest through the general transformations, interfaces, and storage mechanisms observed in artificial systems. Such an approach is presented in the path 1 of Figure 2. Power laws are highly suitable in modeling complex system behavior with nonlinear relationships. Power laws are combined with the DACM framework to leverage the multiplicative and additive relationships of bond graph organs in path 1 from Figure 2.

The DACM framework is a modeling approach developed by the authors to produce models combining causally ordered graphs and power laws. In its functional-based usage, the DACM framework uses functional modeling combined with an algorithm-derived bond graph organs to create a causally ordered graph of a system associated with equations in form of power laws. It can also support the automatic search for physical contradictions as defined in TRIZ (Jordan, 1973; Savransky, 2000). In the functional part of the DACM theory, multiplicative relationships are used for all the modeling organs except between junctions (i.e., interfaces) where additive relationships are used.

Power laws can also be used to study natural phenomena in the form of probability distributions of variables. This is exemplified in the Bayesian usage of DACM in Figure 4. The overarching goal of this research is to provide a methodology for creating knowledge models combining the structure of causally ordered graphs and power laws. The sources of knowledge are dataset itself and the units of this dataset.

To do so, we need to measure and evaluate the semantic fidelity of simulation models as well as producing parsimonious and explainable models because those models are used later to reason and to propagate qualitative objectives. Non-causally ordered models may generate wrong predictions using this qualitative approach (Mokhtarian *et al.*, 2017) and should consequently be avoided. Subsequently, the proposed modeling method can be viewed as a machine learning approach to develop integrated models with different levels of fidelity.

Figure 2 presents two separate modeling processes separated in two distinct paths, where path 2 is the focus of this paper. Additionally, Figure 3 summarizes possible usages of the resulting models that may help the readers to envision the potential applications of the research work presented in this article. The result of both paths 1 and 2 processes is a model combining an oriented graph and a set of power laws equations. In the case of path 2, two distinct models are produced using the approach presented in this work. One is a model produced using exclusively the

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

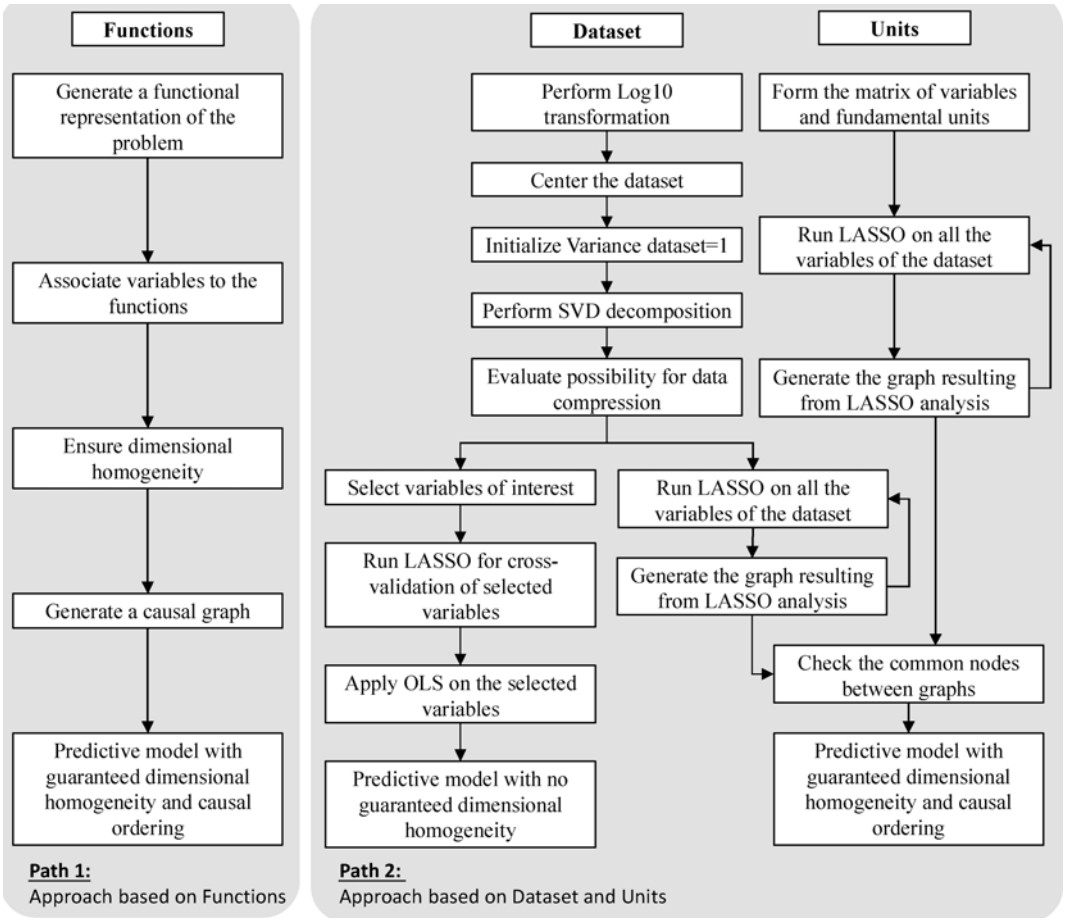


Fig. 2. General processes in DACM. Path based on functions and path 2 based on dataset and units.

information from the dataset but is not guaranteed to be dimensionally homogeneous. The other is a model formed of a causally oriented graph and power laws and is dimensionally homogeneous.

The implement of the method to generate models from dataset and its unit (i.e., path 2 shown in Fig. 2) is described step-by-step as follows. When working with generating power laws from dataset, it is practical to transform the initial dataset in order to ensure working with a linearized form. This is done by using the $\log_{10}$ transformations during the modeling process. The linear transformation allows using a large set of mathematical methods derived from linear algebra. The $\log_{10}$ transformation plays another role in the normalization phase that it allows reducing the scales of the different variables, thus, increasing the efficiency of the variable selection approach from data (i.e., the LASSO method which is an L1 regularization method). After the $\log_{10}$ transformation, the dataset is centered. This is followed by two classical operation in the process of preparing datasets, first centering the distribution and second obtaining a normalized variance of 1 for each column of the dataset (Freedman *et al.*, 2007). This is leading to the matrix presented in Appendix C (Variance analysis). This is useful especially when the numerical values associated with the variables exist in different order of magnitudes. After normalization, the method is followed by matrix decomposition, L1 regularization (i.e., LASSO), and L2 regularization (i.e., OLS) methods to enable complex system modeling. A more detailed account of the L1 and L2 regularization can be found in these publications (Xu *et al.*, 2011; Pennington *et al.*, 2014).

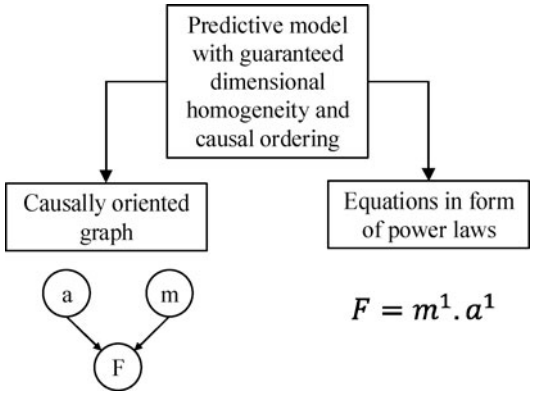


Fig. 3. Nature of the resulting models formed of a combination of an oriented graph and a power-law equation. The other predictive model with no guaranteed homogeneity does not differ in nature.

The matrix decomposition phase is using an SVD decomposition (i.e., an orthogonal decomposition ensuring that principal components are decoupled). The aim of the decomposition is to assess the potential benefit of a dimension reduction that could reduce the noise in the dataset and verify the validity of the hypothesis leading to the choice of the modeling in form of power laws. When this analysis is done, two actions are performed separately, first exploring the left branch of the path 2 process in

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

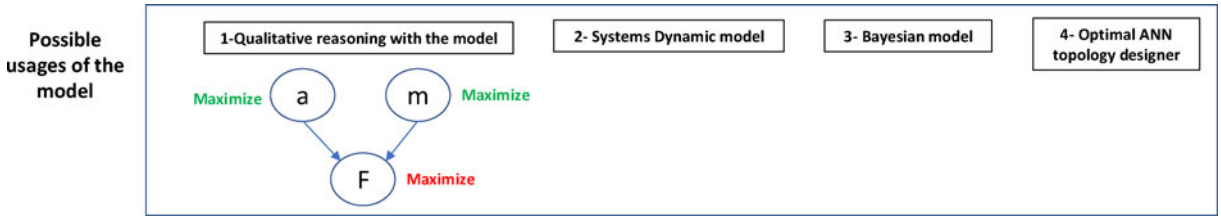


Fig. 4. List of possible design usages of the model produced.

Figure 2. LASSO is applied to select variables influencing the outputs of interest and the final regression using an OLS regression is applied to the set of influencing variables previously selected using LASSO. This process is leading to a model useful for making numerical predictions but not qualified for causal reasoning.

The right branch of the fork in path 2 is combined with the unit branch from path 2 and this combination is exploring the integration of the knowledge coming from the dataset and the fundamental units of the variables in the dataset to guarantee the dimensional homogeneity for the final model. This is done via a recursive use of LASSO on both the normalized dataset from Appendix C (Variance analysis and their fundamental units). The exploitation of Table 3 (the fundamental set of dimensions) in the case study is bringing several interesting properties such as sparsity, bringing stability of Table 3 to the LASSO selection process. As a result of the case study, two graphs are produced separately on the dataset and the dimensional unit's matrix of the variables in the case study in Figures 9 and 11. Those graphs are combined by retaining the common edges between graphs (i.e., shared connections between variables). This is allowing to generate a causally ordered graph integrating the additional knowledge source of the dimensional analysis.

Figure 3 presents an elementary exemplification of the final model produced by the DACM approach using the example of Newton's second law of motion. This law can be rediscovered using the approach described in path 2 of Figure 2.

The possible usages of such type of models are briefly summarized in Figure 4. The key purpose is to support synthesis and generation of novel solutions. The first usage is a qualitative usage. Using the mathematical machinery created by Bhaskar and Nigam (1990), it is possible to propagate backward the red objective in Figure 4 to find the green objectives that are resulting in our case from the positive exponents in the power-law relation $F = m^1 \cdot a^1$. This mechanism allows propagation in complex graphs and possible detection of physical contradictions (Savransky, 2000) appearing on individual nodes. It is possible using the TRIZ separation principle as well as other principles to reduce the contradictions. This is leading in the DACM framework to a modification of the graphs corresponding to a modification of the design and behavior of the designed system.

The second usage of the graphs is a direct application of the model in the system dynamics framework (Sterman, 2001). The concept of stock and flow are not used but replaced directly by the variables of the model. The equations associated with the graph in Figure 3 are directly used to populate the system dynamics model. A third application is a direct usage as a Bayesian Network (BN) (Pearl, 2000). The training of the probability distributions in the BN uses the produced equation too. The topology of the BN is a direct result of the causal ordering from the DACM approach. Another related application is to use the DACM approach as a tool for optimizing the topology of artificial neural networks (ANNs). The equations produced in DACM can be used to generate a dataset itself employ to the train the ANN using a traditional supervised learning approach (Nagarajan *et al.*, 2018). The next section presents via a case study the path 2 from Figure 2. An initial analysis of the dataset is also done using a functional modeling perspective. This is allowing to reason about the transferability of the dataset to the desired case study. The message is that it is possible under certain conditions to use a dataset for design case slightly different from its initial production context.

Case study

Let us consider a physical phenomenon in which a stationary solid body is placed in a moving fluid. A drag force (w) is exerted on the sphere of diameter (l) due to friction generated by the moving fluid having the following characteristics of density (ρ), viscosity (ν), and velocity (V). The nonlinear physical phenomenon is illustrated in Figure 5.

The case study example is a classical problem in fluid dynamics in which the drag force resulting from the velocity of the flow of a certain fluid must be evaluated and modeled. In the example, the solid body is stationary whereas the fluid is in motion. To aid in the modeling and to facilitate the fidelity analysis, we can envision the case study example in the form of a product design case study. From a design context, the data regarding the variables of the system can be used to derive a design case study wherein, the solid body can be envisioned as a spherical underwater drone moving in seawater. The key difference with the initial case study description is that the drone (or solid body) is in motion with a velocity V and not the seawater (or fluid).

The underwater drone represented by the sphere of diameter (l) has a key function which is to navigate in the seawater. This movement is a source of friction due to the contact between the drone and seawater. Thus, from a system view, the desired function *to move into the water* can be defined, while an undesired function provided by the water can be defined *to resist the movement of the drone*. Using a taxonomy of functions presented in the DACM framework, a functional model of the original case study and the derived case study is presented in Figure 6. In both situations, the functions and their interaction remain the same.

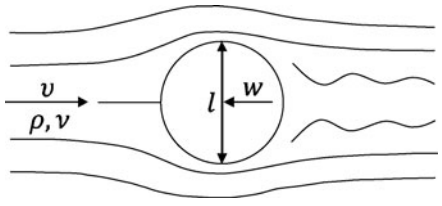


Fig. 5. Pictorial representation of a stationary solid spherical body in a moving fluid.

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

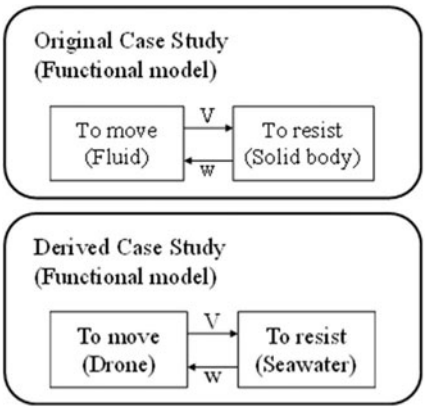


Fig. 6. Functional representation of the case study in its original form (top) and derived form (down).

For the derived case study, the environment is made up of seawater with fundamental properties such as the density of the fluid (ρ) and the viscosity of the fluid (ν). The design objective for the drone would be to reduce the drag force (w) to facilitate the movement within the environment medium (seawater). By reducing the drag force, the energy consumption and maneuverability of the drone can be improved. The model to be created should use the experimental data collected (Appendix A) to derive relationships between the different variables of the system to compute the drag force as a function of different variables as presented in Eq. (1). It should be noted from the dataset (Appendix A) and Figure 6 that a unique dataset can be used to model two radically different engineering situations. In the original case study where the fluid is moving around a solid body while in the derived case study, a drone is traversing through a stationary seawater medium. Nevertheless, in both cases, the variables of the system and its environment remain the same, the difference is in the functional representation. From the perspective of fidelity, it is important to ensure that we would be able to differentiate between the two cases since they have different intended uses. Even though the model structure may appear similar, the individual variables are associated with different functions and parts of the system. Such a distinction is supported by the DACM framework by integrating a generic taxonomy of the variables (Mokhtarian *et al.*, 2017). In DACM, the variables are classified as power variables and state variables. Power variables are decomposed into generalized flow and effort variables representing connections between different functions. Alternately, state variables are decomposed into generalized displacement, momentum, and connecting variables within a function. Using the variables of the system from the case study dataset (Appendix A), the functional representation of the derived case (Fig. 2) can be modified as shown in Figure 7. This dataset is a dataset produced by Prandtl and his team (Prandtl *et al.*, 1927) and used as it is for this article.

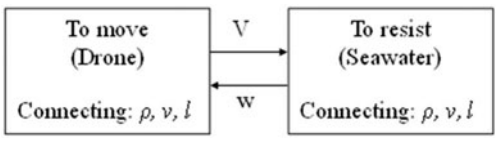


Fig. 7. Updated functional representation of the derived case study underwater drone moving in seawater.

Figure 7 can be presented mathematically [Eqs (1) and (2)] to show the behavior of each technical function described using verbs of action, *to move*, and *to resist*.

$$V = f\ (w, \rho, \nu, l), \tag{1}$$

$$w = f\ (V, \rho, \nu, l), \tag{2}$$

where V is the output of the desired function "to move" and w is the output of the non-desired function "to resist". Using Eqs (1) and (2) and Appendix A, supplementary knowledge can be extracted about the fundamental dimensions (mass, length, and time) of the variables of the system. The variables of the system represented in the form of their fundamental dimensions are shown in Table 1. The encoding of the table is done in the following manner. For example, if the density ρ is considered, then its unit is in kg/m^3. kg is a unit mass (M), and m is a unit of length (L). The exponent for kg is 1 and since kg is divided by m^3, the exponent for m is −3. This is explaining the second columns of Table 1. The rest of the columns are constructed using a similar approach. For the last column, the force w measured in Newton (N), the reader should remember that a force, according to Newton's second law of motion, is a product of a mass (M) and an acceleration ($L{\cdot}T^{-2}$). As for w, the representation of variables using a combination of fundamental dimensions is also, for some variables, integrating the knowledge present in elementary laws of physics.

From Table 1, no columns are a linear combination of one another. The rank of the matrix in Table 1 corresponds to the maximal number of linearly independent columns of this matrix with a rank of five (5). Having three fundamental dimensions M, L, and T, it is possible to form 5–3 = 2 relations between the variables in the columns. Using the updated functional representation in Figure 7 and the dataset in Appendix A, a simulation model can be developed for the derived case study. The taxonomy developed in the DACM framework enables the clustering of variables and functions. A machine learning approach using the DACM framework is presented in this work to automate the generation of parsimonious models from datasets for the derived case study of the drone. First, the development of a simulation model from data is presented followed by an integration of dimensional homogeneity principles to measure and evaluate the fidelity of the developed model.

Modeling using power laws

The DACM framework was initially developed by the authors to generate oriented graph models for product-process modeling and metamodeling (Mokhtarian *et al.*, 2017). The approach initially starts from a functional model which is used as the base upon which the oriented graphs are developed. The current article intends to move a step further to generate models directly from

Table 1. Variables of the system represented in the form of fundamental dimensions

	ν (m^2/s)	ρ (kg/m^3)	l (m)	V (m/s)	w (N)
Mass (M)	0	1	0	0	1
Length (L)	2	−3	1	1	1
Time (T)	−1	0	0	−1	−2

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

datasets to match current machine learning approaches for unsupervised and supervised learning applications (Pelleg and Moore, 1999). The article leverage some key ideas, one of them, is to exploit the knowledge provided by the decomposition in form of fundamental dimensions presented in Table 1 as a supplementary knowledge complementing datasets and powered by the theory of dimensional analysis and the theorem of Vashy-Buckingham (Bhaskar and Nigam, 1990; Szirtes, 2007) presented in different forms in Eqs (3), (5), and (6) to detect potential confounders and to form valid causal ordering of the problem variables (Gilli, 1992). The final models produced by the approach presented in this article are in the form of a combination of oriented graphs associated with power laws (general form shown in Eq. (3)). The power laws provide a powerful tool to evaluate the dimensional homogeneity.

$$Y = A \cdot X_1^{\alpha_1} \cdot X_2^{\alpha_2} \cdots X_n^{\alpha_n}, \tag{3}$$

where Y represents the output variable, A is a constant, X_i are the covariates, and α_i represents the exponents of the covariates. The methodology developed in the article relies on classical tools from linear algebra facilitated by a normalization step using logarithmic transformation of the dataset. The log transformation proves advantageous for improving the efficiency of the different steps in the modeling approach. The normalization also aims at reducing the scale of the dataset using either a decimal or natural log transformation. However, it is important to ensure that the dataset does not contain negative or null values. Following the normalization procedure, the power-law expression in Eq. (3) can then be formulated as shown in Eq. (4).

$$\log Y = \log A + \alpha_1 \cdot \log X_1 + \alpha_2 \cdot \log X_2 + \cdots + \alpha_n \cdot \log X_n. \tag{4}$$

Subsequently, the equations for the model can be constructed in two ways as shown in Eqs (5) and (6). The knowledge extracted from the dataset in Appendix A is used to formulate Eq. (5) whereas, alternately, using the fundamental dimensions of the variable (shown in Table 1), the model can be formulated as Eq. (6).

$$\pi_y = y_i \cdot x_j^{\alpha_{ij}} \cdot x_l^{\alpha_{il}} \cdot x_m^{\alpha_{im}}, \tag{5}$$

$$\pi_y = [D]_i \cdot [D]_j^{\alpha_{ij}} \cdot [D]_l^{\alpha_{il}} \cdot [D]_m^{\alpha_{im}}. \tag{6}$$

In Eq. (5), y_i and $x_{j,l,m}$ represent the variables of interest. In Eq. (6), $[D]$ represents the fundamental dimensions of the variables $x_{j,l,m}$. In both equations, α_i represents the exponents of the power law's form. The exponents in both equations can be similar or dissimilar depending on the nature of the source of information selected (i.e., the dataset or the fundamental dimensions). The exponents can be calculated directly from the experimental dataset, or the exponents can be calculated through dimensional analysis using the fundamental dimensions of the variables in the study. The fundamental dimensions are used in this research to improve the fidelity of the models produced as explained further in the section titled, Deriving a machine-learning model combining data and dimensions assources of information, starting with the analysis provided in Table 4. Thus, for the derived case study, two types of models can then be created using Eqs (5) and (6).

In the next subsection, the development of the simulation model directly from data is explained and demonstrated. The authors would like to highlight that the model developed in the next subsection uses only the experimental data and does not make use of the knowledge available regarding the dimension of the variables. Additionally, the clustering resulting from the use of the DACM taxonomy is not used in the next section. Instead, the DACM clustering is used as a cross-validation method to validate an automated clustering algorithm using LASSO directly from data.

Deriving a machine-learning model exploiting only the data source

In this section, a machine learning approach combining power laws and oriented graphs is presented to model the derived case study using the data from Appendix A. The modeling method follows the following steps namely, normalization of the dataset leading to the data in Appendix C, singular value decomposition (SVD), compression analysis, selection of the performance variables, selection of the variables influencing the performance variables, computation of the optimal relationships between selected parameters and performance parameters, and finally, the generation of the power models.

The normalization phase aims at linearizing the dataset to reduce the scale of the data but also to prepare the dataset for the use of linear algebra in the subsequent steps of the method. The SVD decomposition and compressibility analysis aim at applying compression, when possible, on the dataset to form a latent space. For many complex systems, the distribution of the principal components follows a long-tail distribution. Compression to the latent space can be beneficial for complex systems as it can remove noise in the signal and allow for a more efficient selection of important variables in the following data analytics phases. The compression to the latent space can be useful when the distribution of the variables is a long-tail distribution, otherwise, the compression step can be skipped.

The LASSO selection and its cross-validation are used to accurately select variables from the data which have a high influence on the performance variables thus, allowing for the development of parsimonious models. The performance variables are the variables selected by the modeler to assess the performance of the system.

The LASSO approach can be viewed as a variation of the OLS regression in which covariates not contributing to the output of the variables of interest are gradually discarded.

LASSO stands for Least Absolute Shrinkage and Selection Operator, in Eq. (8) below, the second term is the sum of the absolute value of the magnitude of exponents w_j. A problem occurs since $|w_j|$ is not differentiable in 0. Updates of the coefficients w_j are done in the following manner, when $|w_j|$ is becoming close to zero, a threshold condition is used to set w_j to 0. The condition can be $w_j = 0$ if $|w_j| < \delta_{\text{threshold}}$.

LASSO provides sparse solutions. It is an excellent approach for modeling situations where the number of parameters to be analyzed is very high.

$$\text{Min}\, \varepsilon(w) = \sum_{i=1}^{n} \frac{1}{\sigma_i^2} \cdot (\hat{y}_i - y_i)^2 + \lambda \sum_{j=0}^{m} |w_j|. \tag{7}$$

Thus, the approach works as a variable selection method contributing to the development of parsimonious models. Parsimony is an important principle in machine learning approaches due to their ability to support the generalization of models. LASSO acts

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

as an L1-regularized regression method to support parsimony in the model development process but often at the cost of stability of the selection process. In the context of this paper, the stability of the selection algorithm can be seen as the method providing identical outputs on two datasets differing on only one sample value (Xu *et al.*, 2011). Thus, to improve on the stability of the model development method, an L2-regularized regression (OLS) step is introduced to leverage the parsimony principle from LASSO but simultaneously improve the stability of the modeling approach.

The OLS regression phase brings stability to the model by providing an L2 regularization (i.e., computation of the optimal values of the coefficients in equations). The final power-law equations of the model are obtained by reversing the normalization steps.

The dataset follows the normalization approach presented in path 2 of Figure 2 and in the text above. The transformation of the dataset following the different steps in the modeling process is shown in Appendix B. The SVD is performed on the dataset after normalization at which point the variance of the dataset is equal to one (1). The goal of the SVD is to highlight the most important phenomena in the initial dataset and remove any secondary information or noise which is impertinent to the performance variables. The SVD decomposition for an input matrix $[X]$ is shown in Eq. (8).

$$[X] = [U] \cdot [W] \cdot [V]^T. \tag{8}$$

Here, $[X]$ is the input data matrix and $[U]$, $[W]$, and $[V]^T$ are the decomposed matrices resulting from the SVD. Following the SVD, the compressibility of the dataset is evaluated. Compression is possible if the decay of the eigenvalues of the input data matrix $[X]$ follows a Pareto distribution. A Pareto distribution is a power-law distribution also called a long-tail distribution. Such distributions are often found in complex systems when hierarchy and subsystems are present. If such a distribution is not found, then LASSO variable selection is performed on a non-compressed dataset with the assumption that the noise or complexity of the dataset does not hinder the LASSO selection process. The compressibility check is performed by computing a linear fit (Fig. 8) for the decay of eigenvalues in the decomposed matrix $[W]$ from Eq. (8). A log-log scale is used to ascertain the presence of a power law. An *R*-squared value computed for the fit can determine how well the regression fits with the observed data. A value closer to one (1) suggests that the regression fit matches well with the observed data and hence, can also suggest in our case the presence of power-law distribution (i.e., Pareto distribution).

For the derived case study, the linear fit resulted in an *R*-squared value of 60.6% suggesting that the decay of eigenvalues does not follow a power-law (i.e., Pareto distribution). Thus, we can conclude that the benefit of applying compression to the current data set is limited or null. It can also be noticed that the design problem that we are studying is not integrating a hierarchy of subsystems at this level of analysis as reflected in the decay of Eigenvalues. This observation was verified using different compression levels during the LASSO selection parameter selection process explained further in the text. The compression appeared not to influence the LASSO process. Consequently, during the LASSO selection as well as the steps located after LASSO usage in path 2 of Figure 2, the dataset from Appendix C is exploited without compression. The result of the LASSO variable selection has been applied recursively to all the variables in the dataset is shown in Table 2.

It can be noted that the SVD decomposition and the linear fit for the decay of eigenvalues in log-log scale are also a measure of the conditioning of the dataset used in this article. This is impacting the numerical process and especially the matrix inversions and requires in specific cases to evaluate the number of digits required in the numerical process. Furthermore, sparse, well-structured, and well-conditioned matrixes are properties that are desired when processing matrixes (Kaveh, 2018). The present article does not address those aspects but the exploitation of the dimensional analysis information from Table 1 has connections with those properties and is a practical way to use in addition of experimental datasets another source of information exhibiting those properties.

From Table 2, we can see that *V*, the speed of the underwater drone is shown to be the most influencing and influenced variable in the dataset. Thus, using Table 2, a graph-based representation is developed as shown in Figure 9 with *V* in the center.

Table 2 and Figure 9 will be reused for the fidelity analysis section of this article further. The LASSO selection is followed by the L2-regularized regression approach (i.e., traditional OLS) to model two outputs namely, *V* and *w* to represent performances of two key functions of the simulation model. Equations (9) and (10) show the general form in which the regression approach

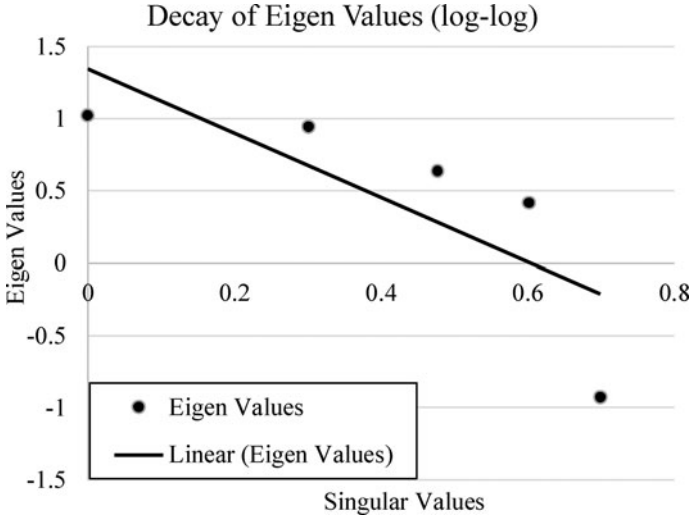


Fig. 8. Linear fit for the decay of eigenvalues in log-log scale (vertical axis: eigenvalues and horizontal axis: singular values from first to fifth).

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

Table 2. Variable selection using LASSO for the derived case study

Outputs \ Influencing Variables	ν	ρ	l	V	w	Influenced
ν	0	1	0	1	0	2
ρ	1	0	0	1	0	2
l	0	0	0	1	1	2
V	1	0	1	0	1	3
w	0	0	1	1	0	2
Influencing	2	1	2	4	2	

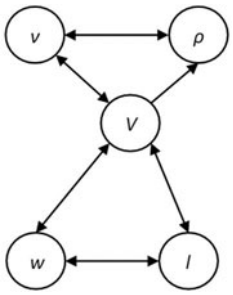


Fig. 9. Graph-based representation of selected variables from LASSO for the derived case study.

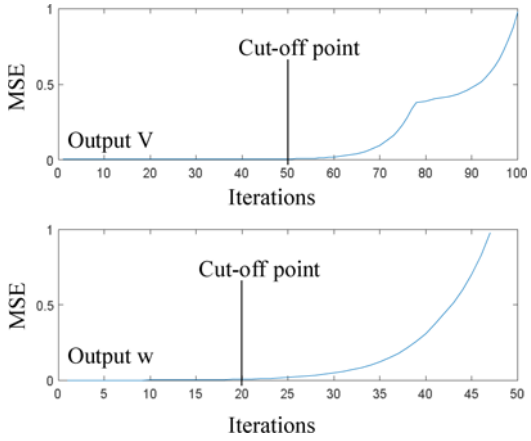


Fig. 10. Iterations and their impact on the mean squared error (MSE) for output variable (*V* – top and *w* – bottom).

using OLS is computed.

$$\log V = \log A + \alpha_1 \cdot \log \nu + \alpha_2 \cdot \log l + \alpha_3 \cdot \log w, \quad (9)$$

$$\log w = \log B + \beta_1 . \log l + \beta_2 . \log V, \quad (10)$$

Where log A and log B are the constants in the equation. Coefficients α_i and β_i are computed using the L2-regularized regression.

LASSO is an iterative approach wherein, a cut-off value for iterations must be specified to maximize the parsimony without reducing the precision of the model developed. The cut-off value for iterations is determined automatically at the point where the mean squared error (MSE) during selection starts to increase as shown for outputs V and w in Figure 10. The last values before having the MSE raising from 0 are selected so for V, the iteration 50 is selected and for w, the value 20 is selected in Figure 10.

Following the OLS computation step, the final models for the output V and w with coefficients computed are shown in Eqs (11) and (12), respectively.

$$\log V = -0.039 \cdot \log \nu - 1.980 \cdot \log l + 2.668 \cdot \log w, \quad (11)$$

$$\log w = 0.747 \cdot \log l + 0.370 \cdot \log V. \quad (12)$$

The MSE for Eq. (11) was found to be 0.0039 with an R-square value of 0.996 and the adjusted R-square value of 0.9957. For Eq. (12), the MSE was found to be 7.46×10^{-04} with an R-square value of 0.999 and an adjusted R-square value of 0.999. The values suggest a very good fit to the observed data for both outputs. In both cases, the intercept values are null because of the normalization procedure performed in Appendix C. The final step in the model development method aims at converting Eqs (11) and (12) in the form of power laws. A denormalization approach is performed to enable the conversion. Equations (11) and (12) are first multiplied by the standard deviations for V and w, followed by adding to both equations the mean values for V and w. Finally, both equations must be exponentiated to obtain their respective power-law representation as shown in Eqs. (13) and (14) for outputs V and w, respectively.

$$V = 7.328 \cdot \nu^{-0.01} \cdot l^{-0.57} \cdot w^{0.71}, \quad (13)$$

$$w = 0.107 \cdot l^{1.032} \cdot V^{0.511}. \quad (14)$$

The computation of V and w for prediction purposes can be solved using a simple linear algebra method when the equations are in the linear format of Eqs (11) and (12), respectively. A simple algorithm is presented at the end of the section to ensure the computation of the unknown outputs for the derived case study.

From Eqs (13) and (14), it is visible that both equations are not dimensionally homogeneous and cannot become dimensionally homogeneous even if a fundamental dimension is associated with the constants in the model as summarized in Table 3 for performance variable V.

Deriving a machine-learning model combining data and dimensions as sources of information

From a fidelity perspective, a limitation of the models formed by Eqs (13) and (14) becomes apparent. The equations are not

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

Table 3. Dimensional homogeneity analysis for performance variable V in Eq. (13)

Variables	V	ν	l	w
Exponents	1	−0.01	−0.53	0.709
Fundamental dimensions				
M	0	0	0	M^1
L	L^1	L^2	L^1	L^1
T	T^{-1}	T^{-1}	0	T^{-2}

dimensionally homogeneous despite their usefulness for providing a good numerical prediction of the output variables. This is also a sign that the causality is not correct.

This can be due to the presence of confounding variables and relationships and/or from the incompleteness of the set of considered variables. Thus, this research further hypothesizes that integrating dimensional homogeneity can improve the fidelity of the developed models by supporting the detection of confounding variables and relationships. To do so, this research uses the DAT and Vashy-Buckingham theorem with the result of the LASSO analyses to improve the fidelity of the developed models. But first, we must tackle a fundamental problem faced by the DAT and the Vashy-Buckingham theorem. In the Vashy-Buckingham theorem, each Pi number produced [Eq. (5)] is the product of a performance variable and a subset of repeating variables. The Vashy-Buckingham theorem provides a necessary condition to create dimensionless numbers from a subset of variables. However, two problems not solved by the Vashy-Buckingham theorem remain: (1) How to select the performance variables meaningful to the study? and (2) How to build the associated subset of repeating variables, especially when multiple possible subsets are available? Figure 9 and Table 2 offer the start of an answer to both problems. From Table 2, we noticed that V is the variable having the most incoming edges and outgoing edges. From a DAT perspective, V is a desirable choice for a performance variable. In DAT, the problem is described using five variables and three fundamental dimensions. This is implying that two dimensionally homogeneous equations can be formed. Usually, two performance variables are selected to build those equations. Nevertheless, Figure 6 provides a useful piece of information. The speed V of the drone is provided first before getting feedback in form of a drag force w generated by the water. From Figure 6, it can be seen that the entire behavior of the system and the environment reaction described in the functional models are initiated by the speed V. w is then the consequence of V and one unique performance variable V can be considered in this case study. Figure 9 and Table 2 have been able to unveil this aspect automatically. Thus, the first question can be answered to determine the performance variable as highlighted using Table 2. The centrality measure in graph theory will allow to highlight such performance variables after producing Table 2 (Tutte, 2001). The authors would like to clarify that this result requires more extensive validation efforts using more case studies to be generalized. In our context, V is used as a performance variable for the two dimensionless equations that can be produced from the list of variables and their dimensions (i.e., five variables and three dimensions resulting in two equations). The next step is to select the appropriate repeating variables for which Figure 9 and Table 2 provide base knowledge. The performance variable must be connected with other repeating variables. Figure 9 can be used for this analysis but in its current state does not integrate any dimensional homogeneity principles. Thus, another iterative LASSO selection applied to the entire set of variables represented in the form of their fundamental dimensions from Table 3 is performed to create Table 4.

The information from Table 4 is presented in the form of a graph in Figure 11. From the table and graph, choosing V as the unique performance variable, we have two directions in which we can move in Figure 11 to select the subset of repeating variables. The performance variable V with fundamental dimensions LT^{-1} has two DOFs, L and T. Thus, we would need to ensure that the choice of repeating variables can match the same dimensions of the performance variable (V). Moving up from speed (V) in Figure 11, the first choice of repeating variable would be the viscosity (ν). Following this, the length (l) would also need to be added to the first subset of repeating variables alongside viscosity (ν) to ensure dimensional homogeneity. Similarly, if we choose to move down from V in Figure 11, the first choice for repeating variable would be the drag force (w). However, the addition of drag force introduces another fundamental dimension which is mass (M) implying an additional DOF.

Thus, two more repeating variables namely density (ρ) and length (l) would be needed to ensure dimensional homogeneity in this case. For the performance variable (V), two dimensionless equations can be developed using the two different subsets of repeating variables as explained above. The method followed in this article provides an automatic mechanism to select the minimal subset of repeating variables required to ensure dimensional homogeneity. However, the authors would like to mention that the performance of the presented approach may be limited when several variables of the system contain the same dimensions and/or are a linear composition of other variables. In which case, the LASSO approach might randomly select one variable using the information available in Table 1. Thus, to ensure accurate selection of variables, a cross-validation approach is also included

Table 4. Variable selection using LASSO for fundamental dimensions of variables in Table 1

		Influencing Variables					
		v	ρ	l	V	w	Influenced
Outputs	v	0	0	1	1	0	2
	ρ	0	0	1	0	1	2
	l	1	1	0	0	0	2
	V	1	0	0	0	1	2
	w	0	1	0	1	0	2
	Influencing	2	2	2	2	2	

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

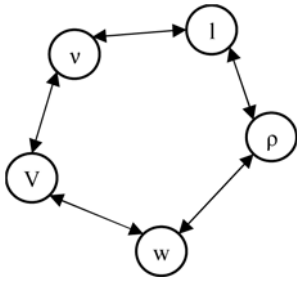


Fig. 11. Graph-based representation of variable selection from Table 4.

in the methodology by combining the results of Tables 2 and 4 as explained below. The idea is to provide a supplementary cross-validation mechanism to detect potential remaining confounders.

The cross-validation of the selection approach is performed by evaluating the shared similarity between Table 2 (selection based on data) and Table 4 (selection based on fundamental dimensions). By doing so, Table 5 can be developed which can validate the immediate connections for each variable.

The "TRUE" values in the different cells in Table 5 implies that shared similarity was identified between the corresponding cells in Tables 2 and 4. Similarly, "FALSE" values imply a lack of shared similarity. Thus, for performance variable (V), the two choices for the first repeating variable would either be viscosity (ν) or drag force (w). Thus, Table 5 provides a validation of the LASSO selection results in Table 4. An alternate way to look at the connections determined by Table 4 and Figure 9 is that the direct link between $\nu \rightarrow l$ in Figure 9, is established with the path $\nu \rightarrow V \rightarrow l$ in Figure 7. Similarly, the direct link between $l \rightarrow \rho$ in Figure 9, is established with the path $l \rightarrow V \rightarrow \rho$ in Figure 7. Finally, the direct link between $w \rightarrow \rho$ in Figure 9 is associated with the path $w \rightarrow V \rightarrow \rho$ in Figure 7 thus, providing the similarity of knowledge extracted directly from data and the fundamental dimensions of the variables in the data. In our case, it reinforces the confidence of the direct connections between variables provided by Figure 9, since it is cross-validated by indirect paths in Figure 7 through the intermediate node (V) in the graph.

Following the cross-validation of the variable selection process, we can develop the two dimensionless equations (i.e., 5 variables − 3 dimensions = 2 equations) for performance variable (V) using the two subsets of repeating variables chosen. The first subset contains viscosity (ν) and length (l), while the second subset contains drag force (w), density (ρ), and length (l) as repeating variables. The fundamental dimensions of each subset of repeating variables identified are shown in Tables 6 and 7, respectively.

The two tables above allow the computation of the exponents α_i for Eq. (8). Two identical approaches can be used to achieve this objective. The vectors of the performance variable V are named $[A]$ and the vectors matrixes of the repeating variables are collected in $[B]$. The following general relationship exists between the matrixes [Eq. (15)].

$$[A] = [B] \cdot [\alpha], \tag{15}$$

where $[\alpha]$ represents the unknown coefficients ensuring the dimensional homogeneity which can be computed using Eq. (16).

$$[\alpha] = [B]^{-1*} \cdot [A], \tag{16}$$

where $[B]^{-1^*}$ represents the Moore-Penrose pseudoinverse. The use of pseudoinverse is required since matrix B is not always a square matrix. The pseudoinverse is calculated as shown in Eq. (17).

$$[B]^{-1*} = ([B]^T [B])^{-1} \cdot [B]^T. \tag{17}$$

The calculation of the $[\alpha]$ using the Moore-Penrose pseudoinverse can also be substituted by OLS regression to obtain the coefficient values. Thus, from the vector and matrix representations in Tables 6 and 7 and using Eqs. (15)–(17), the two dimensionless equations for performance variable (V) are shown in Eqs (18) and (19), respectively.

$$V = \pi_{\mathrm{Re}} \cdot \nu^1 \cdot l^{-1}, \tag{18}$$

$$V = \pi_{\mathrm{Cd}} \cdot w^{1/2} \cdot \rho^{-1/2} \cdot l^{-1}. \tag{19}$$

Both equations can be rearranged in the following form [Eqs (20) and (21)] in which π_{Re} and π_{Cd} are two dimensionless values resulting from the ratio of the variables in Eqs (18) and (19), respectively. From the perspective of fluid dynamics for the derived case study, π_{Re} represents the Reynolds number (Re) commonly used to predict flow patterns. The second one is the drag coefficient, also a dimensionless value used to quantify the resistance of an object in a fluid environment.

$$\pi_{\mathrm{Re}} = \frac{Vl}{\nu}, \tag{20}$$

$$\pi_{\mathrm{Cd}} = \frac{w}{\rho V^2 l^2}. \tag{21}$$

From Eqs (20) and (21), we have four variables which are unknowns namely, V, w (original outputs of the derived case

Table 5. Cross-validation approach evaluating the shared similarity between LASSO selection results from Tables 2 and 4

	Influencing Variables					
	ν	ρ	l	V	w	Influenced
ν	FALSE	FALSE	FALSE	TRUE	FALSE	1
ρ	FALSE	FALSE	FALSE	FALSE	FALSE	0
l	FALSE	FALSE	FALSE	FALSE	FALSE	0
V	TRUE	FALSE	FALSE	FALSE	TRUE	2
w	FALSE	FALSE	FALSE	TRUE	FALSE	1
Influencing	1	0	0	2	1	

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

Table 6. Fundamental dimensions of the first subset of repeating variables

	Performance	Repeating	
Dimensions	V (m/s)	ν (m^2/s)	l (m)
M	0	0	0
L	L^1	L^2	L^1
T	T^{-1}	T^{-1}	0

Table 7. Fundamental dimensions of the second subset of repeating variables

	Performance	Repeating		
Dimensions	V (m/s)	w (N)	ρ (kg/m^3)	l (m)
M	0	M^1	M^1	0
L	L^1	L^1	L^{-3}	L^1
T	T^{-1}	T^{-2}	0	0

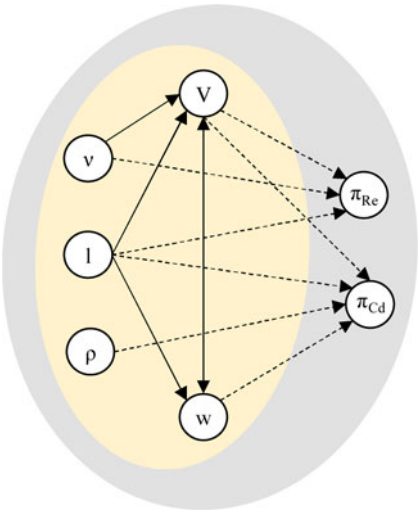


Fig. 12. Final oriented graph model for the derived case study showing two levels of fidelity (1 – lower fidelity, variable level: yellow; and 2 – higher fidelity, function level: gray).

study as shown in Eqs (13) and (14)), π_{Re}, and π_{Cd} (the dimensionless numbers developed in Eqs (20) and (21)). Thus, Eqs (13), (14), (20), and (21) must be computed conjointly to solve the system of equations. The initial modeling objective in this article for the derived case study aimed at modeling the speed (V) and drag force (w) which can be accurately predicted with the parsimonious models developed in Eqs (13) and (14). However, Eqs (20) and (21) developed using DAT overlay an additional layer of information to the model. Evaluating in the context of the functional model developed for the derived case study, Eqs (13) and (14) only assess one parameter in each function of the drone. However, the second set of models was developed using DAT in Eqs (20) and (21) assess globally each of the functions of the drone shown in Figure 4. The current representation in Eqs (20) and (21) provide a dimensionally homogeneous expression fulfilling the required specification of a function thus, suggesting a higher level of fidelity in the developed approach.

The two sets of models [Eqs (13) and (14), and Eqs (20) and (21)] can be represented graphically (Fig. 10) to holistically represent the model. In Figure 10, arrows represent the connection between variables.

Solid arrows represent the lower fidelity connections of the system model where fitting using data and parsimony criteria are fulfilled. The dotted arrows combine the fitting and parsimony criteria but in addition, integrate dimensional homogeneity to enable a causal ordering. The graph-based model shown in Figure 10 can be used as a pretrained topology for an ANN model. The prediction of the unknown outputs V, w, π_{Re} π_{Cd} in Figure 10 is done using the algorithm presented in Tables 6 and 7. The algorithm is generic and can be applied to all problems analyzed using this approach. Figure 12 associated with Eqs (13), (14), (20), and (21) is a specific type of ANN. This type of ANN differs from traditional ANN due to the topology design of the structure of the oriented graph. It differs also in the way the training of the weights is done and in the nature of the equations governing the modeled phenomena. The algorithm for computing unknown output is presented in Table 8.

The graph model (Fig. 10) can also be used to define a BN topology wherein, the equations can be used to determine the conditional relations between different variables (represented as nodes in a BN) to facilitate the Bayesian inference mechanism. The final graph model produced in this section integrates two levels of fidelity. The first level facilitates computation at the variable level and is not dimensionally homogeneous. The second level provides the required granularity to holistically assess a function's performance and behavior concerning the system of interest. Additionally, this second level of fidelity also ensures dimensional homogeneity as well as a valid causal ordering compliant with modeler expected usage of the model. Those aspects will be studied further in future studies. Selection of the variables influencing the performance variables, computation of the optimal relationships between selected parameters and performance parameters, and finally, the generation of the power models.

Discussion and future work

This research presented a modeling and machine learning method to create simulation models with measurable levels of fidelity for engineering design. The method is a machine learning method serving as a basis for evaluating the fidelity of different types of models. The developed approach generates parsimonious explainable models that enable generic knowledge representation in the form of oriented graph with valid causal ordering, associated with power laws. The approach uses data as initial knowledge to model the system using power laws and graph-based representations of the modeled equations. The graph-based representation is a synthetic manner of representing the different equations increasing the interpretability of the model. The research also highlights the use of the graph-based representation as a precursor to generating the graph topology for more classical machine learning approaches such as BNs and ANNs. The graph representation used in this work is an approach to create optimal topologies for BN and ANNs as well as training the specific form of ANN presented in this work.

From the viewpoint of model fidelity, the developed method demonstrated the existence of different levels and sources of knowledge (i.e., the experimental dataset and the dimensions of the variables). The models produced using the two sources of knowledge are different and provide two levels of granularity in the demonstrated case study. The first level of granularity was at a variable level to predict the key performance parameters of

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

Table 8. Algorithm for computing the unknown outputs for the derived case study

Algorithm: Output prediction algorithm

Inputs: *Normalized known inputs values* $\boldsymbol{P}_{in}$ *and unknown outputs* $\boldsymbol{P}_{Out}$ *in a* log_{10} *form, centered and with Variance=1. The coefficients matrix* $[\boldsymbol{A}]$ *contains the* log_{10} *form of computed coefficients. The matrix* $[\boldsymbol{B}]$ *contains the* log_{10} *form of the computed constants.*

Outputs: *Predicted outputs* $\boldsymbol{P}_{out\ final}$

Initiate $\begin{bmatrix} \boldsymbol{P}_{in} \\ \boldsymbol{P}_{Out} \end{bmatrix} . [\boldsymbol{A}] = [\boldsymbol{B}]$

Compute $[\boldsymbol{A}]^{-1*} = ([\boldsymbol{A}]^T [\boldsymbol{A}])^{-1} . [\boldsymbol{A}]^T$

$$\begin{bmatrix} \boldsymbol{P}_{in} \\ \boldsymbol{P}_{Out} \end{bmatrix} = [\boldsymbol{B}] . [\boldsymbol{A}]^{-1*}$$

Exponentiate each $\boldsymbol{P}_{in\,i}$ and $\boldsymbol{P}_{out,i}$ to obtain $\boldsymbol{P}_{out,i\,final}=10^{\boldsymbol{P}_{out,i}}$ and $\boldsymbol{P}_{in,i\,final}=10^{\boldsymbol{P}_{in,i}}$

End

the derived case study presented in Figure 3. The second level of granularity is achieved for the computation of system-level criteria, the key functions (desired and non-desired) of the system of interest for the derived case study. The two levels of granularity in the developed model can be associated as two intricated layers of fidelity, an initial layer, with a lower fidelity and not adhering to the dimensional homogeneity criteria, and a second level matching the homogeneity criteria. Together, the two levels of the model can be represented in form of a synthetic graph. A part of this graph is an oriented graph representing the first layer of fidelity, while the second part is an oriented graph with a validated causal ordering representing the second layer of fidelity. This causal ordering is compliant with the intentional use of the model by the modeler as presented in the background section.

The developed method can be coined as a machine learning method for its ability to automatically model the system based on experimental data. In addition, the method enables simulations using the graph presentation and the output computing algorithm (Table 8) presented in the paper. The developed approach will be applied in future work to the dynamic control of manufacturing systems. One requirement for such an application is the need for the control system to be able to respond to sensors information in close to real-time. The parsimonious nature of the models developed using this method can offer the benefit to be light and can be integrated into decentralized control systems located as close as possible from sensors. Additionally, an interesting characteristic of the developed method is the composability of the models providing a modular approach where elementary building blocks can be assembled to form extended models.

The research developed in this article presents a novel combination of known and validated methods to generate an actionable framework contributing to the fidelity evaluation of models. Tangible criteria relying on dimensional analysis and their interrelations with the concept of variables and functions have been leveraged in the developed modeling approach to represent two specific levels of fidelity. Additionally, this supplementary knowledge was used to direct the causal ordering of the second model. It is claimed that a direct link exists between a higher fidelity and a valid causal ordering. It was also demonstrated that the higher fidelity model was corresponding to a representation of the behavior of functions of the studied system. The developed approach demonstrates that simulation at lower granularity level (i.e., variable level) can be performed efficiently with a lower model fidelity while function level models requires a higher model fidelity, at

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

least for the case study exemplified in this research. Higher fidelity in the models can be achieved by using the supplementary knowledge provided by the fundamental dimensions of the variables. The analysis presented in this research has voluntarily been focused on engineering models.

The key contributions of the research are highlighted as follows:

1) A machine learning approach for engineering modeling using data, for generating parsimonious and explainable models has been presented. A cut-off criterion is defined to combine the parsimony and precision of the model. The different steps of the machine learning method have been exemplified in the case study.
2) It has been shown that the LASSO graphs resulting from an experimental dataset and a supplementary information in form of a dimensional dataset (i.e., Table 1) can be used as a cross-validation method for the selection of the variables forming the performance and repeating variables in the higher fidelity model. This contribution is effective in the exemplified case in this research but needs to be validated further for other complex case studies.
3) Two interrelated models have been developed and aggregated via a graph-based representation. Obtaining the highest level of fidelity implies the fulfillment of supplementary constraints requiring a holistic set of variables related to the different functions of a system. The research enables modeling at distinct levels of fidelity, integrated into a single graph-based model which supports simulation.
4) Finally, an algorithm for predicting the unknown outputs for the distinct levels of fidelity in the models has been presented as a new form of ANNs.

The method developed in this article can be viewed as a reverse engineering approach capable of extracting explainable knowledge in a parsimonious form from diverse types of collected datasets. The log transformation used in the dataset normalization enables the use of linear algebra algorithms to analyze highly nonlinear problems. The modeling method combines conjointly L1 and L2 normalization to support the optimal selection between parsimony and stability in the models produced. The developed approach is computationally efficient and is scalable for larger datasets. In the current research, the size of the datasets handled by the method is potentially small. In this respect, the method demonstrates that efficient modeling of problems where data are very limited can be performed. Like human learning, only a small number of experiments are requested for new learnings.

The idea underlying the paper is to propose a method coming closer from the number of cases required by the human learning process. Samples of the size of two digits in our case can be considered as small. Small dataset is associated with poor statistical validity of the models produced. Therefore, we need to exploit other sources of knowledge not directly present in the dataset. We exploited the units of the variables from the dataset. Those units allow us to generate sparse matrixes, well-conditioned matrixes and well-structures matrixes (Kaveh, 2018) bringing stability to the algorithms together with the L2 regularization used by OLS regression. They can compensate for the limited size of the dataset and for the inability of the machine learning methods to integrate the dimensional analysis homogeneity.

A limitation of the current work not discussed in the paper, but of fundamental importance for the scope of the approach, is that metrics are not always available to measure variables in a dataset. Multiple situations in design, and social sciences exist where getting a measure metric for variables in a dataset is a challenge. The authors are developing an approach allowing the exploitation of linguistic definitions to generate the type of matrixes presented in Table 1. This development is a significant expansion of the scope of the work presented in the paper and significantly increase the application domain of the approach.

Future work will assess different case studies and problems where collinearity between variables exists. This is important to make the approach robust and validate the generic principles emerging from this study. Second, the presented approach has established a link between fidelity, the nature of the knowledge, and the nature of the models. Nevertheless, the concept of fidelity is broader, and a real operationalization of that concept needs to be associated with quantified tolerances. For operational use, it is important to define the minimal and maximal fidelity acceptable for a model. Additionally, it is important to be able to measure the fidelity of the model and to compare it with given tolerances. To do so, the authors plan to expand on the framework proposed to quantify fidelity by transforming criteria such as dimensional homogeneity, completeness of variables set, types of datasets, produced models into quantifiable and measurable criteria or levels.

Data availability statement. The data used in the article is reproduced at the end of the article in the appendix. This dataset was produced by Prandtl *et al.* (1927) and reused by the authors.

Financial statement. The research presented in this article was funded for H.N. by the DIGITBrain project from H2020 (EU commission) and for D.W. and S.P. by the Smaragdi project (Business Finland).

Conflict of interests. No competing interests related to the presented research exist for the authors of the article.

References

Berman JJ (2018) *Principles of Big Data: Preparing, Sharing, and Analyzing Complex Information*, 2nd edition. Elsevier. ISBN 978-0-12-815609-4.

Bhaskar R and Nigam A (1990) Qualitative physics using dimensional analysis. *Artificial Intelligence* **45**, 73–111.

Bhatt AN, Majumder A and Chakrabarti A (2021) Analyzing the modes of reasoning in design using the SAPPhIRE model of causality and the extended integrated model of designing. *AI EDAM* **35**, 384–403.

Bhosekar A and Ierapetritou M (2018) Advances in surrogate based modeling, feasibility analysis, and optimization: a review. *Computers & Chemical Engineering* **108**, 250–267.

Brunton SL and Kutz JN (2019) *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press. doi:10.1017/9781108380690

Cai L, Ren L, Wang Y, Xie W, Zhu G and Gao H (n.d.) Surrogate models based on machine learning methods for parameter estimation of left ventricular myocardium. *Royal Society Open Science* **8**, 201121. doi:10.1098/rsos.201121

Chakrabarti A, Sarkar P, Leelavathamma B and Nataraju B (2005) A functional representation for aiding biomimetic and artificial inspiration of new ideas. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing : AI EDAM* **19**, 113–132.

Chaudhuri S, Dayal U and Narasayya V (2011) An overview of business intelligence technology. *Communications of the ACM* **54**, 88–98.

Choi SH, Lee SJ and Kim TG (2014) Multi-fidelity modeling & simulation methodology for simulation speed up. Proceedings of the 2nd ACM SIGSIM/PADS Conference on Principles of Advanced Discrete Simulation - SIGSIM-PADS '14, pp. 131-150. doi:10.1145/2601381.2601385

Coatanéa E (2015) Dimensional analysis conceptual modelling (DACM): a comprehensive framework for specifiying, validating, and analyzing system models from a model-based system engineering perspective (SOW, 4 (4.5),1). US Department of Defence.

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

Dadi GB, Goodrum PM, Taylor TR and Carswell CM (2014) Cognitive workload demands using 2D and 3D spatial engineering information formats. *Journal of Construction Engineering and Management* **140**, 04014001.

Eckert C, Kelly I and Stacey M (1999) Interactive generative systems for conceptual design: an empirical perspective. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* **13**, 303–320. doi:10.1017/S089006049913405X

Felsberger A, Oberegger B and Reiner G (2016) A review of decision support systems for manufacturing systems. SAMI@ iKNOW.

Finkbeiner B, Gieseking M and Olderog E-R (2015) Adam: causality-based synthesis of distributed systems, pp. 433–439.

Freedman D, Pisani R and Purves R (2007) *Statistics: Fourth International Student Edition*. W.W. Norton & Company. ISBN 9780393930436.

Gawthrop P and Smith L (1996) *Metamodelling: Bond Graphs and Dynamic Systems, 1996*. Englewood Cliffs, New Jersey: Prentice-Hall, pp. 2093–2121. Fig, 14.

Gilli M (1992) Causal ordering and beyond. *International Economic Review* **33**, 957–971. doi:10.2307/2527152

Goel AK, Vattam S, Wiltgen B and Helms M (2012) Cognitive, collaborative, conceptual and creative — four characteristics of the next generation of knowledge-based CAD systems: a study in biologically inspired design. *Computer-Aided Design* **44**, 879–900. doi:10.1016/j.cad.2011.03.010

Goldberger AS (1964) Econometric theory (Faks.-repr. d. Ausg Wiley, 1964). UMI.

Goldberger AS (1972) Structural equation methods in the social sciences. *Econometrica* **40**, 979–1001. doi:10.2307/1913851

Gross DC (1999) Report from the fidelity implementation study group. Fall Simulation Interoperability Workshop Papers.

Gross DC, Tucker WV, Harmon SY and Youngblood SM (1999) Fidelity Theory Advances in Distributed Simulation, pp. 415–420.

Halpern JY (2000) Axiomatizing causal reasoning. *The Journal of Artificial Intelligence Research* **12**, 317–337. doi:10.1613/jair.648

Harman HH (1976) *Modern Factor Analysis*. Chicago: University of Chicago Press. ISBN 9780226316529.

Horváth I and Vroom RW (2015) Ubiquitous computer aided design: a broken promise or a sleeping beauty? *Computer-Aided Design* **59**, 161–175. doi:10.1016/j.cad.2014.10.006

Hotelling H (1933) Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology* **24**, 417–441.

Iwasaki Y and Simon HA (1994) Causality and model abstraction. *Artificial Intelligence* **67**, 143–194. doi:10.1016/0004-3702(94)90014-0

Jin Y and Benami O (2010) Creative patterns and stimulation in conceptual design. *AI EDAM* **24**, 191–209.

Jordan ZA (1973) The evolution of dialectical materialism. A philosophical and sociological analysis. *Zeitschrift Für Philosophische Forschung* **27**, 316–322.

Kaplan D (2008) *Structural Equation Modeling: Foundations and Extensions*. SAGE Publications.

Kaveh A (2018) Graph transformations for efficient structural analysis. *Acta Mech* **229**, 659–675.

Kryssanov VV, Tamaki H and Kitamura S (2001) Understanding design fundamentals: how synthesis and analysis drive creativity, resulting in emergence. *Artificial Intelligence in Engineering* **15**, 329–342.

Lee B, Cooper R, Hands D and Coulton P (2022) Continuous cycles of data-enabled design: reimagining the IoT development process. *AI EDAM* **36**, e11. doi:10.1017/S0890060421000299

Li Y, Zhao W and Shao X (2012) A process simulation based method for scheduling product design change propagation. *Advanced Engineering Informatics* **26**, 529–538. doi:10.1016/j.aei.2012.04.006

Mayda M and Börklü HR (2014) An integration of TRIZ and the systematic approach of Pahl and Beitz for innovative conceptual design process. *Journal of the Brazilian Society of Mechanical Sciences and Engineering* **36**, 859–870.

McAfee A and Brynjolfsson E (2012) *Big Data: The Management Revolution*. Harvard Business Review. Available at: https://hbr.org/2012/10/big-data-the-management-revolution

McCaffrey T and Spector L (2018) An approach to human–machine collaboration in innovation. *AI EDAM* **32**, 1–15.

Miller GA (1956) The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review* **63**, 81.

Mobus GE (2022) Principles of systems science. In Mobus GE (ed.), *Systems Science: Theory, Analysis, Modeling, and Design*. Springer International Publishing, pp. 41–87. doi:10.1007/978-3-030-93482-8_2

Mokhtarian H, Coatanéa E and Paris H (2017) Function modeling combined with physics-based reasoning for assessing design options and supporting innovative ideation. *AI EDAM* **31**, 476–500. doi:10.1017/S0890060417000403

Moon FC (2007) *The Machines of Leonardo Da Vinci and Franz Reuleaux: Kinematics of machines from the Renaissance to the 20th Century*, Vol. **2**. Dordrecht: Springer.

Nagarajan HPN, Mokhtarian H, Jafarian H, Dimassi S, Bakrani-Balani S, Hamedi A, Coatanéa E, Gary Wang G and Haapala KR (2018) Knowledge-based design of artificial neural network topology for additive manufacturing process modeling: a new approach and case study for fused deposition modeling. *Journal of Mechanical Design* **141**, 021705-021705–021712. doi:10.1115/1.4042084

Ostwald PF and Muñoz J (2008) *Manufacturing Processes and Systems*, 9th Edn. John Wiley & Sons. Available at: https://app.knovel.com/hotlink/pdf/id:kt011AZNW3/manufacturing-processes/motive-power

Pearl J (2000) *Causality: Models, Reasoning and Inference*, Vol. **29**. Springer.

Pelleg D and Moore A (1999) Accelerating exact k-means algorithms with geometric reasoning. Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '99, pp. 277–281. doi:10.1145/312129.312248

Pennington J, Socher R and Manning CD (2014) Glove: global vectors for word representation, pp. 1532–1543.

Ponnusamy SS, Albert V and Thebault P (2014) A Simulation Fidelity Assessment Framework. Proceedings of the 4th International Conference on Simulation and Modeling Methodologies, Technologies and Applications, pp. 463–471. doi:10.5220/0005105704630471

Prandtl L, Wieselsberger C and Betz A (1927) Ergebnisse der aerodynamischen Versuchsanstalt zu Göttingen. Lfg. 3. Walter de Gruyter GmbH & Co KG.

Roca R (2021) *Semantic Fidelity Nomenclature: An Alternative to Simulation "Low-Medium-High" Fidelity Monikers*. Simulation Interoperability Workshop, 13. https://www.sisostds.org/Workshops/SISOWorkshops/2021SIW.aspx

Roza M, Voogd J, Jense H and Gool PV (1999) Fidelity requirements specification: a process oriented view. In Fall Simulation Interoperability Workshop.

Savransky SD (2000) *Engineering of Creativity: Introduction to TRIZ Methodology of Inventive Problem Solving*. Boca Raton: CRC Press.

Shahin TM (2008) Computerizing conceptual design without hindering creativity. *Computer-Aided Design and Applications* **5**, 548–556.

Shekhawat K, Jain RN, Bisht S, Kondaveeti A and Goswami D (2021) Graph-based approach for enumerating floorplans based on users specifications. *AI EDAM* **35**, 438–459. doi:10.1017/S0890060421000275

Shephard MS, Beall MW, O'Bara RM and Webster BE (2004) Toward simulation-based design. *Finite Elements in Analysis and Design* **40**, 1575–1598. doi:10.1016/j.finel.2003.11.004

Simon HA (1956) Rational choice and the structure of the environment. *Psychological Review* **63**, 129.

Sornette D (2006) *Critical Phenomena in Natural Sciences*. Springer-Verlag. doi:10.1007/3-540-33182-4

Spirtes P, Glymour CN, Scheines R and Heckerman D (2000) *Causation, Prediction, and Search*. MIT Press.

Sterman JD (2001) System dynamics modeling: tools for learning in a complex world. *California Management Review* **43**, 8–25. doi:10.2307/41166098

Sun G and Wang S (2019) A review of the artificial neural network surrogate modeling in aerodynamic design. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* **233**, 5863–5872. doi:10.1177/0954410019864485

Szirtes T (2007) *Applied Dimensional Analysis and Modeling*. Butterworth-Heinemann.

Trave-Massuyes L and Pons R (1997) Causal ordering for multiple mode systems. 11th International Workshop on Qualitative Reasoning, pp. 203–214.

Tutte WT (2001) *Graph Theory*. Cambridge: Cambridge University Press.

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

VanderWeele TJ and Shpitser I (2013) On the definition of a confounder. *Annals of Statistics* **41**, 196.

Xu H, Caramanis C and Mannor S (2011) Sparse algorithms are not stable: a no-free-lunch theorem. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **34**, 187–193.

Zou H and Hastie T (2005) Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **67**, 301–320. doi:10.1111/j.1467-9868.2005.00503

Dr. Eric Coatanéa received his BS degree in mechanical engineering in 1990 from the University of West Brittany, Brest, France, his MS in mechanical engineering from INSA Toulouse, France in1993, and his teaching certification from Ecole Normale Supérieure of Cachan, France in 1994. Dr. Coatanéa worked for 11 years as a teacher of manufacturing engineering at the University of West Brittany, Brest, France while in parallel being very active in track and field. He embarked in joint doctoral studies in 2002 and received his joint PhD degree in mechanical engineering from Helsinki University of Technology, Finland (nowadays Aalto University) and the University of West Brittany, Brest, France in 2005. Currently, Dr. Coatanéa is a Professor of Manufacturing Engineering and Systems Engineering at the Faculty of Engineering and Natural Sciences, Tampere University, Finland. He was Marie-Curie fellow in 2005–2007. He held several visiting professorships at different universities. Dr. Coatanéa is in charge of the Manufacturing research group at Tampere University. His research interests include design and manufacturing activities, system thinking, and system engineering. More specifically, the research focus is given to the early integration of simulations and decision support tools during the initial design phases.

Hari Nagarajan is a research scientist in advanced manufacturing technologies at the Technical Research Centre of Finland, VTT. He received his PhD in mechanical engineering from Tampere University (Finland) in 2022. His research interests include Life Cycle Engineering, Material Processing using Additive Manufacturing, Metamodelling Framework Development for Additive Manufacturing, and Sustainability Evaluation.

Hossein Mokhtarian is a University Lecturer at Tampere University. He holds a double PhD degree from Tampere University (Finland) and Grenoble Alpes University (France) in mechanical engineering and industrial engineering. His research interests include the intersection of digital design and advanced manufacturing. He has a passion and solid pedagogical background.

Di Wu is a post-doctoral researcher at Tampere University. He got his PhD degree from Simon Fraser University (Canada) in mechanical engineering. His research major is multi-disciplinary design optimization, machine learning/deep learning methods, and their applications in engineering designs.

Suraj Panicker is a doctoral researcher working at Tampere University, Finland. His research interests include metal additive manufacturing, sustainable manufacturing, and data analytics in manufacturing.

Andrés Morales-Forero is a PhD candidate in industrial engineering at Polytechnique Montreal, Canada. His research focuses on Causal Machine Learning and Safe, Ethical, and Explainable AI. He is currently working on different projects to find mathematical and statistical models for complex challenges in research, industry, and society.

Samuel Bassetto is a Canadian professor of industrial and system engineering at Polytechnique Montréal. He is born and made his scholarship in France. He earned PhD in 2005, from ParisTech – Arts et Métiers. He got his MSc in artificial intelligence from the University of Lorraine in 2001 and his BIng in mechanical engineering from ParisTech also in 2001. He worked for several industrial companies from 2001 to 2006. Since 2006 in the University of Savoie in France and in 2011 in Polytechnique Montréal in Canada. He has directed more than 100 MSc and PhD students, co-published more than 70 scientific papers, and earned several awards for his research. Now he directs the LABAC, his R&D laboratory of process control. He focuses the research on the design of intelligent alarms (data science and artificial intelligence) and relevant actions (psychiatry and neuroscience). He is also a caver and a climber.

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

Appendix A. Experimental data

Index	ν (m^2/s)	ρ (kg/m^3)	l (m)	V (m/s)	w (N)
1	1.84×10^{-5}	1.183	0.008	2.2	1.8029×10^{-4}
2	1.80×10^{-5}	1.185	0.008	2.81	2.5690×10^{-4}
3	1.80×10^{-5}	1.184	0.008	3.52	4.0279×10^{-4}
4	1.81×10^{-5}	1.193	0.008	4.01	5.2057×10^{-4}
5	1.80×10^{-5}	1.182	0.008	4.5	6.5411×10^{-4}
6	1.80×10^{-5}	1.18	0.008	6.02	1.0482×10^{-3}
7	1.80×10^{-5}	1.185	0.008	6.66	1.3456×10^{-3}
8	1.80×10^{-5}	1.226	0.018	3.22	1.5898×10^{-3}
9	1.80×10^{-5}	1.226	0.018	4.29	2.7926×10^{-3}
10	1.80×10^{-5}	1.227	0.018	5.64	4.8939×10^{-3}
11	1.79×10^{-5}	1.224	0.018	6.84	7.3474×10^{-3}
12	1.81×10^{-5}	1.227	0.018	8.34	1.1254×10^{-2}
13	1.79×10^{-5}	1.201	0.062	3.59	2.6061×10^{-2}
14	1.78×10^{-5}	1.197	0.062	4.36	3.8661×10^{-2}
15	1.78×10^{-5}	1.198	0.062	4.89	5.0544×10^{-2}
16	1.79×10^{-5}	1.2	0.062	5.66	6.7976×10^{-2}
17	1.78×10^{-5}	1.198	0.062	6.45	8.8129×10^{-2}
18	1.79×10^{-5}	1.2	0.062	7.88	1.3376×10^{-1}
19	1.79×10^{-5}	1.2	0.062	9.04	1.7793×10^{-1}
20	1.79×10^{-5}	1.2	0.062	9.86	2.1302×10^{-1}
21	1.79×10^{-5}	1.203	0.062	10.7	2.5254×10^{-1}
22	1.82×10^{-5}	1.199	0.1	7.6	3.3311×10^{-1}
23	1.82×10^{-5}	1.201	0.1	8.88	4.5647×10^{-1}
24	1.83×10^{-5}	1.206	0.1	10.4	6.3133×10^{-1}
25	1.82×10^{-5}	1.201	0.1	12.7	9.4336×10^{-1}
26	1.82×10^{-5}	1.201	0.1	15.6	1.4146×10^{0}
27	1.83×10^{-5}	1.2	0.1	2.35	2.8761×10^{-2}
28	1.83×10^{-5}	1.202	0.1	4.37	1.0674×10^{-1}
29	2.08×10^{-5}	1.566	0.1	5.08	1.8832×10^{-1}
30	1.83×10^{-5}	1.199	0.142	5.29	3.3354×10^{-1}
31	1.83×10^{-5}	1.199	0.142	7.76	7.2647×10^{-1}
32	1.82×10^{-5}	1.186	0.142	10.8	1.3696×10^{0}
33	1.83×10^{-5}	1.192	0.142	12.7	1.8453×10^{0}
34	1.84×10^{-5}	1.203	0.142	15	2.5598×10^{0}
35	1.83×10^{-5}	1.194	0.142	17	3.2006×10^{0}
36	1.83×10^{-5}	1.193	0.142	19.7	4.0984×10^{0}
37	1.83×10^{-5}	1.202	0.142	21.6	4.6476×10^{0}
38	1.76×10^{-5}	1.196	0.142	24.8	5.2507×10^{0}
39	1.83×10^{-5}	1.199	0.142	26.4	5.6279×10^{0}
40	1.82×10^{-5}	1.229	0.283	5.1	1.2033×10^{0}
41	1.82×10^{-5}	1.233	0.283	6.1	1.7197×10^{0}
42	1.79×10^{-5}	1.217	0.283	7.7	2.7508×10^{0}
43	1.80×10^{-5}	1.224	0.283	10.3	4.6072×10^{0}
44	1.82×10^{-5}	1.238	0.283	11.4	5.3733×10^{0}

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

Appendix B. Log_{10} transformation

Index	ν (m^2/s)	ρ (kg/m^3)	l (m)	V (m/s)	w (N)
1	-4.7345×10^{0}	7.2985×10^{-2}	-2.0969×10^{0}	3.4242×10^{-1}	-3.7440×10^{0}
2	-4.7447×10^{0}	7.3718×10^{-2}	-2.0969×10^{0}	4.4871×10^{-1}	-3.5902×10^{0}
3	-4.7442×10^{0}	7.3352×10^{-2}	-2.0969×10^{0}	5.4654×10^{-1}	-3.3949×10^{0}
4	-4.7414×10^{0}	7.6640×10^{-2}	-2.0969×10^{0}	6.0314×10^{-1}	-3.2835×10^{0}
5	-4.7440×10^{0}	7.2617×10^{-2}	-2.0969×10^{0}	6.5321×10^{-1}	-3.1843×10^{0}
6	-4.7452×10^{0}	7.1882×10^{-2}	-2.0969×10^{0}	7.7960×10^{-1}	-2.9795×10^{0}
7	-4.7438×10^{0}	7.3718×10^{-2}	-2.0969×10^{0}	8.2347×10^{-1}	-2.8711×10^{0}
8	-4.7450×10^{0}	8.8490×10^{-2}	-1.7447×10^{0}	5.0786×10^{-1}	-2.7987×10^{0}
9	-4.7450×10^{0}	8.8490×10^{-2}	-1.7447×10^{0}	6.3246×10^{-1}	-2.5540×10^{0}
10	-4.7452×10^{0}	8.8845×10^{-2}	-1.7447×10^{0}	7.5128×10^{-1}	-2.3103×10^{0}
11	-4.7462×10^{0}	8.7781×10^{-2}	-1.7447×10^{0}	8.3506×10^{-1}	-2.1339×10^{0}
12	-4.7435×10^{0}	8.8845×10^{-2}	-1.7447×10^{0}	9.2117×10^{-1}	-1.9487×10^{0}
13	-4.7484×10^{0}	7.9543×10^{-2}	-1.2076×10^{0}	5.5509×10^{-1}	-1.5840×10^{0}
14	-4.7493×10^{0}	7.8094×10^{-2}	-1.2076×10^{0}	6.3949×10^{-1}	-1.4127×10^{0}
15	-4.7488×10^{0}	7.8457×10^{-2}	-1.2076×10^{0}	6.8931×10^{-1}	-1.2963×10^{0}
16	-4.7479×10^{0}	7.9181×10^{-2}	-1.2076×10^{0}	7.5282×10^{-1}	-1.1676×10^{0}
17	-4.7488×10^{0}	7.8457×10^{-2}	-1.2076×10^{0}	8.0956×10^{-1}	-1.0549×10^{0}
18	-4.7469×10^{0}	7.9181×10^{-2}	-1.2076×10^{0}	8.9653×10^{-1}	-8.7367×10^{-1}
19	-4.7467×10^{0}	7.9181×10^{-2}	-1.2076×10^{0}	9.5617×10^{-1}	-7.4976×10^{-1}
20	-4.7476×10^{0}	7.9181×10^{-2}	-1.2076×10^{0}	9.9388×10^{-1}	-6.7159×10^{-1}
21	-4.7476×10^{0}	8.0266×10^{-2}	-1.2076×10^{0}	1.0294×10^{0}	-5.9767×10^{-1}
22	-4.7395×10^{0}	7.8819×10^{-2}	-1.0000×10^{0}	8.8081×10^{-1}	-4.7741×10^{-1}
23	-4.7392×10^{0}	7.9543×10^{-2}	-1.0000×10^{0}	9.4841×10^{-1}	-3.4058×10^{-1}
24	-4.7375×10^{0}	8.1347×10^{-2}	-1.0000×10^{0}	1.0170×10^{0}	-1.9974×10^{-1}
25	-4.7390×10^{0}	7.9543×10^{-2}	-1.0000×10^{0}	1.1038×10^{0}	-2.5321×10^{-2}
26	-4.7402×10^{0}	7.9543×10^{-2}	-1.0000×10^{0}	1.1931×10^{0}	1.5064×10^{-1}
27	-4.7380×10^{0}	7.9181×10^{-2}	-1.0000×10^{0}	3.7107×10^{-1}	-1.5412×10^{0}
28	-4.7385×10^{0}	7.9904×10^{-2}	-1.0000×10^{0}	6.4048×10^{-1}	-9.7168×10^{-1}
29	-4.6824×10^{0}	1.9479×10^{-1}	-1.0000×10^{0}	7.0586×10^{-1}	-7.2509×10^{-1}
30	-4.7366×10^{0}	7.8819×10^{-2}	-8.4771×10^{-1}	$7.2346E \times 10^{-1}$	-4.7685×10^{-1}
31	-4.7366×10^{0}	7.8819×10^{-2}	-8.4771×10^{-1}	8.8986×10^{-1}	-1.3878×10^{-1}
32	-4.7404×10^{0}	7.4085×10^{-2}	-8.4771×10^{-1}	1.0334×10^{0}	1.3659×10^{-1}
33	-4.7371×10^{0}	7.6276×10^{-2}	-8.4771×10^{-1}	1.1038×10^{0}	2.6607×10^{-1}
34	-4.7354×10^{0}	8.0266×10^{-2}	-8.4771×10^{-1}	1.1761×10^{0}	4.0820×10^{-1}
35	-4.7366×10^{0}	7.7004×10^{-2}	-8.4771×10^{-1}	1.2304×10^{0}	5.0524×10^{-1}
36	-4.7375×10^{0}	7.6640×10^{-2}	-8.4771×10^{-1}	1.2945×10^{0}	6.1261×10^{-1}
37	-4.7385×10^{0}	7.9904×10^{-2}	-8.4771×10^{-1}	1.3345×10^{0}	6.6723×10^{-1}
38	-4.7557×10^{0}	7.7731×10^{-2}	-8.4771×10^{-1}	1.3945×10^{0}	7.2021×10^{-1}
39	-4.7366×10^{0}	7.8819×10^{-2}	-8.4771×10^{-1}	1.4216×10^{0}	7.5035×10^{-1}
40	-4.7404×10^{0}	8.9552×10^{-2}	-5.4821×10^{-1}	7.0757×10^{-1}	8.0363×10^{-2}

(*Continued*)

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

(*Continued.*)

Index	v (m^2/s)	ρ (kg/m^3)	l (m)	V (m/s)	w (N)
41	-4.7406×10^{0}	9.0963×10^{-2}	-5.4821×10^{-1}	7.8533×10^{-1}	2.3544×10^{-1}
42	-4.7474×10^{0}	8.5291×10^{-2}	-5.4821×10^{-1}	8.8649×10^{-1}	4.3945×10^{-1}
43	-4.7440×10^{0}	8.7781×10^{-2}	-5.4821×10^{-1}	1.0128×10^{0}	6.6343×10^{-1}
44	-4.7399×10^{0}	9.2721×10^{-2}	-5.4821×10^{-1}	1.0569×10^{0}	7.3024×10^{-1}
Mean	$\mathbf{-4.7412 \times 10^{0}}$	$\mathbf{8.2869 \times 10^{-2}}$	$\mathbf{-1.2157 \times 10^{0}}$	$\mathbf{8.6543 \times 10^{-1}}$	$\mathbf{-9.7118 \times 10^{-1}}$

Center the dataset

Index	v (m^2/s)	ρ (kg/m^3)	l (m)	V (m/s)	w (N)
1	6.7167×10^{-3}	-9.8846×10^{-3}	-8.8126×10^{-1}	-5.2301×10^{-1}	-2.7728×10^{0}
2	-3.5362×10^{-3}	-9.1510×10^{-3}	-8.8126×10^{-1}	-4.1672×10^{-1}	-2.6190×10^{0}
3	-3.0539×10^{-3}	-9.5176×10^{-3}	-8.8126×10^{-1}	-3.1889×10^{-1}	-2.4237×10^{0}
4	-1.7138×10^{-3}	-6.2289×10^{-3}	-8.8126×10^{-1}	-2.6229×10^{-1}	-2.3123×10^{0}
5	-2.8129×10^{-3}	-1.0252×10^{-2}	-8.8126×10^{-1}	-2.1222×10^{-1}	-2.2132×10^{0}
6	-4.0190×10^{-3}	-1.0987×10^{-2}	-8.8126×10^{-1}	-8.5834×10^{-2}	-2.0084×10^{0}
7	-2.5721×10^{-3}	-9.1510×10^{-3}	-8.8126×10^{-1}	-4.1956×10^{-2}	-1.8999×10^{0}
8	-3.7775×10^{-3}	5.6211×10^{-3}	-5.2908×10^{-1}	-3.5757×10^{-1}	-1.8275×10^{0}
9	-3.7775×10^{-3}	5.6211×10^{-3}	-5.2908×10^{-1}	-2.3297×10^{-1}	-1.5828×10^{0}
10	-4.0190×10^{-3}	5.9752×10^{-3}	-5.2908×10^{-1}	-1.1415×10^{-1}	-1.3392×10^{0}
11	-4.9862×10^{-3}	4.9121×10^{-3}	-5.2908×10^{-1}	-3.0374×10^{-2}	-1.1627×10^{0}
12	-2.3315×10^{-3}	5.9752×10^{-3}	-5.2908×10^{-1}	5.5736×10^{-2}	-9.7750×10^{-1}
13	-7.1704×10^{-3}	-3.3263×10^{-3}	8.0432×10^{-3}	-3.1034×10^{-1}	-6.1283×10^{-1}
14	-8.1447×10^{-3}	-4.7752×10^{-3}	8.0432×10^{-3}	-2.2594×10^{-1}	-4.4154×10^{-1}
15	-7.6573×10^{-3}	-4.4125×10^{-3}	8.0432×10^{-3}	-1.7612×10^{-1}	-3.2515×10^{-1}
16	-6.6841×10^{-3}	-3.6881×10^{-3}	8.0432×10^{-3}	-1.1261×10^{-1}	-1.9646×10^{-1}
17	-7.6573×10^{-3}	-4.4125×10^{-3}	8.0432×10^{-3}	-5.5871×10^{-2}	-8.3699×10^{-2}
18	-5.7131×10^{-3}	-3.6881×10^{-3}	8.0432×10^{-3}	3.1096×10^{-2}	9.7518×10^{-2}
19	-5.4707×10^{-3}	-3.6881×10^{-3}	8.0432×10^{-3}	9.0738×10^{-2}	2.2143×10^{-1}
20	-6.4412×10^{-3}	-3.6881×10^{-3}	8.0432×10^{-3}	1.2845×10^{-1}	2.9960×10^{-1}
21	-6.4412×10^{-3}	-2.6037×10^{-3}	8.0432×10^{-3}	1.6395×10^{-1}	3.7352×10^{-1}
22	1.7397×10^{-3}	-4.0502×10^{-3}	2.1565×10^{-1}	1.5383×10^{-2}	4.9378×10^{-1}
23	1.9780×10^{-3}	-3.3263×10^{-3}	2.1565×10^{-1}	8.2983×10^{-2}	6.3060×10^{-1}
24	3.6424×10^{-3}	-1.5220×10^{-3}	2.1565×10^{-1}	1.5160×10^{-1}	7.7144×10^{-1}
25	2.2162×10^{-3}	-3.3263×10^{-3}	2.1565×10^{-1}	2.3837×10^{-1}	9.4586×10^{-1}
26	1.0240×10^{-3}	-3.3263×10^{-3}	2.1565×10^{-1}	3.2769×10^{-1}	1.1218×10^{0}
27	3.1675×10^{-3}	-3.6881×10^{-3}	2.1565×10^{-1}	-4.9436×10^{-1}	-5.7001×10^{-1}
28	2.6921×10^{-3}	-2.9649×10^{-3}	2.1565×10^{-1}	-2.2495×10^{-1}	-4.9583×10^{-4}
29	5.8837×10^{-2}	1.1192×10^{-1}	2.1565×10^{-1}	-1.5957×10^{-1}	2.4609×10^{-1}
30	4.5907×10^{-3}	-4.0502×10^{-3}	3.6794×10^{-1}	-1.4197×10^{-1}	4.9434×10^{-1}
31	4.5907×10^{-3}	-4.0502×10^{-3}	3.6794×10^{-1}	2.4432×10^{-2}	8.3240×10^{-1}
32	7.8521×10^{-4}	-8.7847×10^{-3}	3.6794×10^{-1}	1.6799×10^{-1}	1.1078×10^{0}

(*Continued*)

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

(*Continued.*)

Index	ν (m^2/s)	ρ (kg/m^3)	l (m)	V (m/s)	w (N)
33	4.1168×10^{-3}	-6.5931×10^{-3}	3.6794×10^{-1}	2.3837×10^{-1}	1.2373×10^{0}
34	5.7731×10^{-3}	-2.6037×10^{-3}	3.6794×10^{-1}	3.1066×10^{-1}	1.3794×10^{0}
35	4.5907×10^{-3}	-5.8650×10^{-3}	3.6794×10^{-1}	3.6502×10^{-1}	1.4764×10^{0}
36	3.6424×10^{-3}	-6.2289×10^{-3}	3.6794×10^{-1}	4.2904×10^{-1}	1.5838×10^{0}
37	2.6921×10^{-3}	-2.9649×10^{-3}	3.6794×10^{-1}	4.6902×10^{-1}	1.6384×10^{0}
38	-1.4532×10^{-2}	-5.1382×10^{-3}	3.6794×10^{-1}	5.2902×10^{-1}	1.6914×10^{0}
39	4.5907×10^{-3}	-4.0502×10^{-3}	3.6794×10^{-1}	5.5617×10^{-1}	1.7215×10^{0}
40	7.8521×10^{-4}	6.6825×10^{-3}	6.6744×10^{-1}	-1.5786×10^{-1}	1.0515×10^{0}
41	5.4626×10^{-4}	8.0937×10^{-3}	6.6744×10^{-1}	-8.0100×10^{-2}	1.2066×10^{0}
42	-6.1983×10^{-3}	2.4212×10^{-3}	6.6744×10^{-1}	2.1061×10^{-2}	1.4106×10^{0}
43	-2.8129×10^{-3}	4.9121×10^{-3}	6.6744×10^{-1}	1.4741×10^{-1}	1.6346×10^{0}
44	1.2627×10^{-3}	9.8513×10^{-3}	6.6744×10^{-1}	1.9147×10^{-1}	1.7014×10^{0}
Mean	**0**	**0**	**0**	**0**	**0**
STD	$\mathbf{1.0251 \times 10^{-2}}$	$\mathbf{1.8067 \times 10^{-2}}$	$\mathbf{5.0092 \times 10^{-1}}$	$\mathbf{2.6639 \times 10^{-1}}$	$\mathbf{1.3806 \times 10^{0}}$

Appendix C. Variance analysis

Index	ν (m^2/s)	ρ (kg/m^3)	l (m)	V (m/s)	w (N)
1	6.5520×10^{-1}	-5.4710×10^{-1}	-1.7593×10^{0}	-1.9633×10^{0}	-2.0085×10^{0}
2	-3.4495×10^{-1}	-5.0650×10^{-1}	-1.7593×10^{0}	-1.5643×10^{0}	-1.8971×10^{0}
3	-2.9790×10^{-1}	-5.2679×10^{-1}	-1.7593×10^{0}	-1.1971×10^{0}	-1.7556×10^{0}
4	-1.6718×10^{-2}	-3.4476×10^{-1}	-1.7593×10^{0}	-9.8458×10^{-1}	-1.6749×10^{0}
5	-2.7440×10^{-1}	-5.6743×10^{-1}	-1.7593×10^{0}	-7.9663×10^{-1}	-1.6031×10^{0}
6	-3.9205×10^{-1}	-6.0814×10^{-1}	-1.7593×10^{0}	-3.2221×10^{-1}	-1.4548×10^{0}
7	-2.5091×10^{-1}	-5.0650×10^{-1}	-1.7593×10^{0}	-1.5750×10^{-1}	-1.3762×10^{0}
8	-3.6849×10^{-1}	3.1112×10^{-1}	-1.0562×10^{0}	-1.3423×10^{0}	-1.3237×10^{0}
9	-3.6849×10^{-1}	3.1112×10^{-1}	-1.0562×10^{0}	-8.7454×10^{-1}	-1.1465×10^{0}
10	-3.9205×10^{-1}	3.3072×10^{-1}	-1.0562×10^{0}	-4.2850×10^{-1}	-9.7002×10^{-1}
11	-4.8640×10^{-1}	2.7188×10^{-1}	-1.0562×10^{0}	-1.1402×10^{-1}	-8.4219×10^{-1}
12	-2.2743×10^{-1}	3.3072×10^{-1}	-1.0562×10^{0}	2.0922×10^{-1}	-7.0805×10^{-1}
13	-6.9947×10^{-1}	-1.8411×10^{-1}	1.6057×10^{-2}	-1.1649×10^{0}	-4.4390×10^{-1}
14	-7.9451×10^{-1}	-2.6430×10^{-1}	1.6057×10^{-2}	-8.4816×10^{-1}	-3.1983×10^{-1}
15	-7.4696×10^{-1}	-2.4423×10^{-1}	1.6057×10^{-2}	-6.6113×10^{-1}	-2.3552×10^{-1}
16	-6.5203×10^{-1}	-2.0413×10^{-1}	1.6057×10^{-2}	-4.2273×10^{-1}	-1.4231×10^{-1}
17	-7.4696×10^{-1}	-2.4423×10^{-1}	1.6057×10^{-2}	-2.0973×10^{-1}	-6.0627×10^{-2}
18	-5.5730×10^{-1}	-2.0413×10^{-1}	1.6057×10^{-2}	1.1673×10^{-1}	7.0637×10^{-2}
19	-5.3366×10^{-1}	-2.0413×10^{-1}	1.6057×10^{-2}	3.4062×10^{-1}	1.6039×10^{-1}
20	-6.2833×10^{-1}	-2.0413×10^{-1}	1.6057×10^{-2}	4.8217×10^{-1}	2.1701×10^{-1}
21	-6.2833×10^{-1}	-1.4411×10^{-1}	1.6057×10^{-2}	6.1545×10^{-1}	2.7056×10^{-1}
22	1.6971×10^{-1}	-2.2417×10^{-1}	4.3051×10^{-1}	5.7747×10^{-2}	3.5767×10^{-1}
23	1.9295×10^{-1}	-1.8411×10^{-1}	4.3051×10^{-1}	3.1150×10^{-1}	4.5677×10^{-1}

(*Continued*)

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press

(*Continued.*)

Index	ν (m^2/s)	ρ (kg/m^3)	l (m)	V (m/s)	w (N)
24	3.5531×10^{-1}	-8.4243×10^{-2}	4.3051×10^{-1}	5.6909×10^{-1}	5.5879×10^{-1}
25	2.1618×10^{-1}	-1.8411×10^{-1}	4.3051×10^{-1}	8.9481×10^{-1}	6.8513×10^{-1}
26	9.9893×10^{-2}	-1.8411×10^{-1}	4.3051×10^{-1}	1.2301×10^{0}	8.1259×10^{-1}
27	3.0899×10^{-1}	-2.0413×10^{-1}	4.3051×10^{-1}	-1.8558×10^{0}	-4.1289×10^{-1}
28	2.6261×10^{-1}	-1.6410×10^{-1}	4.3051×10^{-1}	-8.4442×10^{-1}	-3.5915×10^{-4}
29	5.7395×10^{0}	6.1948×10^{0}	4.3051×10^{-1}	-5.9899×10^{-1}	1.7825×10^{-1}
30	4.4781×10^{-1}	-2.2417×10^{-1}	7.3452×10^{-1}	-5.3295×10^{-1}	3.5807×10^{-1}
31	4.4781×10^{-1}	-2.2417×10^{-1}	7.3452×10^{-1}	9.1712×10^{-2}	6.0295×10^{-1}
32	7.6596×10^{-2}	-4.8622×10^{-1}	7.3452×10^{-1}	6.3062×10^{-1}	8.0241×10^{-1}
33	4.0159×10^{-1}	-3.6492×10^{-1}	7.3452×10^{-1}	8.9481×10^{-1}	8.9620×10^{-1}
34	5.6315×10^{-1}	-1.4411×10^{-1}	7.3452×10^{-1}	1.1662×10^{0}	9.9915×10^{-1}
35	4.4781×10^{-1}	-3.2462×10^{-1}	7.3452×10^{-1}	1.3702×10^{0}	1.0694×10^{0}
36	3.5531×10^{-1}	-3.4476×10^{-1}	7.3452×10^{-1}	1.6105×10^{0}	1.1472×10^{0}
37	2.6261×10^{-1}	-1.6410×10^{-1}	7.3452×10^{-1}	1.7606×10^{0}	1.1868×10^{0}
38	-1.4175×10^{0}	-2.8439×10^{-1}	7.3452×10^{-1}	1.9859×10^{0}	1.2252×10^{0}
39	4.4781×10^{-1}	-2.2417×10^{-1}	7.3452×10^{-1}	2.0878×10^{0}	1.2470×10^{0}
40	7.6596×10^{-2}	3.6987×10^{-1}	1.3324×10^{0}	-5.9258×10^{-1}	7.6169×10^{-1}
41	5.3287×10^{-2}	4.4798×10^{-1}	1.3324×10^{0}	-3.0068×10^{-1}	8.7402×10^{-1}
42	-6.0464×10^{-1}	1.3401×10^{-1}	1.3324×10^{0}	7.9058×10^{-2}	1.0218×10^{0}
43	-2.7440×10^{-1}	2.7188×10^{-1}	1.3324×10^{0}	5.5334×10^{-1}	1.1840×10^{0}
44	1.2318×10^{-1}	5.4526×10^{-1}	1.3324×10^{0}	7.1876×10^{-1}	1.2324×10^{0}
STD	**1**	**1**	**1**	**1**	**1**

https://doi.org/10.1017/S0890060422000269 Published online by Cambridge University Press