Journal of Achievements in Materials and Manufacturing Engineering
VOLUME 44
ISSUE 2
February
2011

Formulation and identification of First-Principle Data-Driven models

P. Czop [a,*], G. Kost [b], D. Sławik [a], G. Wszołek [b]
[a] Tenneco Automotive Eastern Europe, Eastern European Engineering Center (EEEC), ul. Bojkowska 59 B, 44-100 Gliwice, Poland
[b] Institute of Engineering Processes Automation and Integrated Manufacturing Systems, Silesian University of Technology, ul. Konarskiego 18a, 44-100 Gliwice, Poland
* Corresponding author: E-mail address: piotr.czop@labmod.com

Received 20.12.2010; published in revised form 01.02.2011

Analysis and modelling

ABSTRACT

Purpose: The paper consists of two parts. The first part presents and discusses a process of formulation and identification of First-Principle Data-Driven (FPDD) models, while the second part demonstrates numerical examples of identification of FPDD models.
Design/methodology/approach: First-Principle (FP) model is formulated using a system of continuous ordinary differential equations capturing usually nonlinear relations among variables of the model. The considering model applies three categories of parameters: geometrical, physical and phenomenological. Geometrical and physical parameters are deduced from construction or operational documentation. The phenomenological parameters are the adjustable ones, which are estimated or adjusted based on their roughly known values, e.g. friction/damping coefficients.
Findings: A few phenomenological parameters were successfully estimated from numerically generated data. The error between the true and estimated value of the parameter occurred, however its magnitude is low at level below 2%.
Research limitations/implications: Adjusting a model to data is, in most cases, a non-convex optimization problem and the criterion function may have several local minima. This is a case when multiple parameters are simultaneously estimated.
Practical implications: FPDD models are an excellent tool for understanding, optimizing, designing, and diagnosing technical systems since they are updatable using operational measurements. This opens application area, for example, for model-based design and early warning diagnostics.
Originality/value: First-Principle (FP) models are frequently adjusted by trial-and-error, which can lead to non-optimal results. In order to avoid deficiencies of the trial-and-error approach, a formalized mathematical method using optimization techniques to minimize the error criterion, and find optimal values of tunable model parameters, was proposed and demonstrated in this work.
Keywords: First-Principle model; Data Driven model; Grey-box; Servo-hydraulic system

Reference to this paper should be given in the following way:
P. Czop, G. Kost, D. Sławik, G. Wszołek, Formulation and identification of First-Principle Data-Driven models, Journal of Achievements in Materials and Manufacturing Engineering 44/2 (2011) 179-186.

1. Introduction

The initial phase in modeling of a technical system is collecting and systematic treatment of available knowledge [1-6]. The a priori knowledge about a given phenomenon comes from the analysis, comprising of finding all possible connections to other phenomena and physical laws, preceding the modeling [7-12]. The a priori knowledge is of key importance in modeling although its availability is always limited by the complexity of the physical system [3,8]. Even if the governing physical principia are known, it is usually difficult to formulate the specific relationships and obtain particular values of the parameters. Availability of the a priori knowledge and the modeling purpose determine the following: (i) the final type of the model, (ii) the accuracy requirements, (iii) the type of specific modeling procedure, (iv) the complexity of the model and lastly, (v) the method and the cost of its realization. According to the degree to which the a priori knowledge is available, then either a first-principle or a data-driven model, or a combination of both, can be applied (cf. Fig. 1). First-principle (FP) models use understanding of the system underlying physics to derive its mathematical representation. FP models are expensive in development since expertise in the area of knowledge at the advanced level is required to derive equations from physical laws, while data-driven (DD) models use system test data to derive its mathematical representation. The advantage of the former approach is the depth of the insight into the behavior of the system and thus ability to predict the performance, while the advantage of the latter is the speed in which an accurate model can be constructed and confidence gained thanks to the use of the data obtained from the actual system. The difficulty of the former approach lies in the determination of the phenomenological parameters like the damping or the heat transfer coefficient. FP models are frequently adjusted by trial-and-error, which can lead to non-optimal results. On the other hand, the disadvantage of DD models is the need to handle multiple data sets in order to cover the range of system operation.

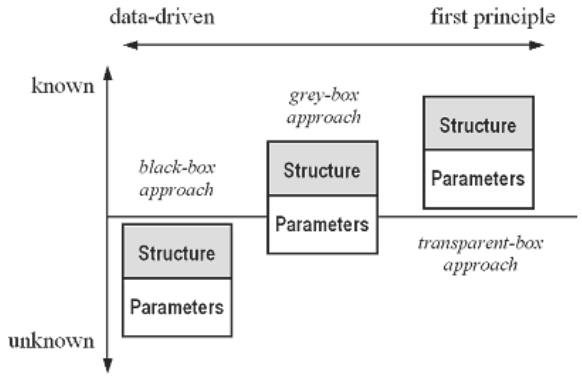

Fig. 1. A servo-hydraulic test-rig used in experimental investigations

The goal is therefore to find a compromise and propose a combined first-principle data-driven (FPDD) model. FPDD models require a formal approach which allows the model parameters to be updated according to the operational data. In order to avoid the drawbacks of trial-and-error approaches, a formalized mathematical method using optimization techniques to minimize the error criterion is proposed for FPDD models. It is believed that the smaller the number of updating model parameters, the more accurate the model and the faster the convergence of the algorithm used for model adjustment. In turn, the application scope of an FPDD model is wide-ranging, including the optimization and diagnostics of technical systems where a method of fast model update is essential.

2. Taxonomy of models

The taxonomy of models regarding a priori available knowledge of the technical system was proposed in [13] and relies on the observation that, in many industrial processes, first-principle or heuristic, yet incomplete knowledge about the system exists. The taxonomy focuses on the way of incorporating a priori knowledge into a model (Table 1).

The first methodology, constrained black-box identification based on regression techniques, originates from the black-box identification frame, where an a priori knowledge is incorporated by imposing constraints on the model parameters. The following is the justification of this approach; a simple continuous model can be transformed into a corresponding discrete time model and known restrictions of the continuous model, such as process stability and step response, can be used to define limits placed on the static gain and the time constants, which are imposed on the parameters of the discrete model. A second methodology, the semi-physical modeling, makes use of case specific nonlinear transformations of measured input/output process signals [13], e.g. a nonlinear sensor characteristic. A Wiener-Hammerstein model is a representative of this class of models [14]. Transformed signals are then used to estimate unknown parameters of a linear black-box model, for instance an ARMAX-type model.

A third methodology, namely the analytical modeling, is based on a basic model originating from mathematical relations derived from the first-principle equations. Analytical modeling deals with lumped and distributed parameter systems. Lumped-parameter models are the most commonly considered in this approach. Nonetheless, spatially distributed phenomena have significant influence on many chemical and thermodynamic processes, for instance on processes involving mass or energy transport by convection or diffusion. Mathematical representation of a distributed-parameter system involves partial differential equations (PDEs) [13].

The fourth of the methodologies, the hybrid modeling, separates the model into a white/transparent box part, the first-principles equations model, and a black-box part represented by a data-fitted, typically, nonlinear data model, e.g. neuro-fuzzy model.

Table 1.
Grey-box modeling methodologies

	Constrained black-box and semi-physical modeling	Analytical modeling	Hybrid modeling
Unknown-to-known parameters ratio	high	low	moderate
Structure complexity	low	high	moderate
First-principle contents	no	yes	yes/no
Application area	model-based control, fault detection, adaptative filtering	system understanding, diagnostics, training simulators	fault detection, process tracking
System identification	standard LS or PEM methods	multi-criteria and general optimization techniques	second-order methods like the Levenberg-Marquardt method
A priori known model structure	not necessary	yes always	yes/no
Artificial intelligence model structures, e.g. neural nets, fuzzy sets, evolutionary computing	yes	no	yes
Partial differential equations (PDEs)models	no	yes	no
On-line estimation	yes	no	yes
Dynamic nonlinearities	yes	yes	yes
Static nonlinearities	yes	yes	yes

2.1. Constrained black box and semi-physical modeling

Black-box models are formulated based on experimental data sets or field operation data sets, and require no a priori knowledge (such as fluid dynamics, thermal dynamics or chemical reactions). They are widely used in industry [15]. Black-box models may include static and dynamic, linear and nonlinear regression models. These models are families of basic ARX (AutoRegressive model with eXogenenous input) or NARX (non-linear ARX) model structures [16]. For the ARX/NARX structure and other related structures, i.e. OE/NOE, ARMAX/NARMAX, BJ/NBJ, the previous outputs return into the model formulating a general linear/nonlinear regressor. Recurrent artificial neural network models are mostly used as nonlinear regressors.

A purely black-box model is not reliable when the process or system exhibits significant nonlinear behavior moving into new operating conditions which may result from configuration changes, new operating practices, or external factors [17,18]. For cases where first-principle modeling is not feasible, it can be possible to make use of physical insight to transform the input and output variables to new variables which are used as regressor to develop a black-box model. If the nonlinear static input/output characteristics are known, it is possible to use the convention of Hammerstein models, Wiener models or general feedback block oriented models (Fig. 2). In [19], the semi-physical grey-box modeling approach has been utilized to distinguish between linear dynamic blocks and static non-linearities in the servo-hydraulic system. The proposed approach has been applied to the hydraulic actuators of a flight simulator motion system, using a well-conditioned experimental setup, with the double-concentric hydraulic actuator placed in a test rig, provided with a three-stage servo-valve and various transducers [19].

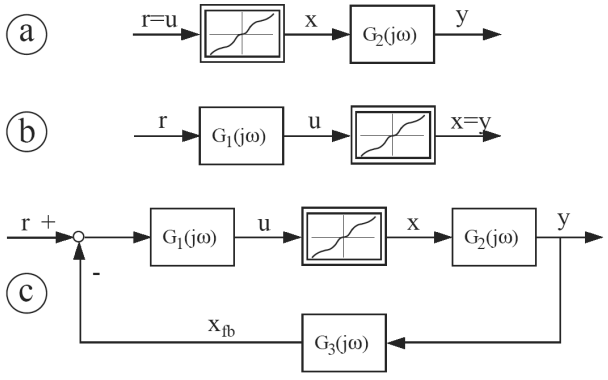

Fig. 2. Block scheme representations of three basic structures of nonlinear grey-box model [18]; Hammerstein model (a); Wiener model (b); general model (c)

Based on the results of this real-life application of the non-linear identification and validation approach, it can be concluded that the proposed method properly utilizes the a-priori knowledge of the model's physical structure to identify the non-linear dynamics of the system. Starting with the non-linear dynamic models of the subsystems of the hydraulic servo-system, the aim of the experimental identification and validation has been to identify the unknown model parameters. Therefore, not only the relevant dynamics, but also the dominant non-linearities, of the hydraulic servo-system have been explicitly taken into account.

2.2. Analytical modeling

In cases where a fundamental insight into the mechanisms that underlie the behavior of a process exists, relevant balance equations can be formulated as a set of first order equations [17]. In general, the initial model structure formulated based on a physical insight must be refined to match the experimental data. The activity of developing an analytical model consists of basic modeling, experiments, estimation, expanded modeling and model validation. The process may cover the following steps [20]:

- A phenomenological description in the form of a verbal, graphical, or other mental description involving the modeling object and experiment conditions.
- Variables and causality enable the description to be translated into a system of causal dependencies between defined variables, for instance in the form of a block diagram. This may introduce internal variables that are not measured, or random disturbances. The step serves to eliminate a number of otherwise mathematically possible relations between the object's input and output variables.
- Mathematical modeling specifies known relations between variables, including parameterization, or chooses structures for unknown relations, including disturbances. If it is not known how many, or what, relations are needed, instead several hypothetical model structures with increasing detail need to be created.
- Calibration allows the simplest models that are not falsified by experimental data to be found. The step involves fitting to the data and testing significance. The results allow uncertainty and credibility to be evaluated, and usually cause a return to the previous step.
- Validation enables the model to be confronted with independent data. If the calibrated model is more complex than the purpose required, then it must be reduced.

Lumped parameter systems are most often considered in this approach. Nevertheless, spatially distributed phenomena are important in many chemical, and energy transfer related processes, and including for instance any process involving mass or energy transport by convection, radiation or diffusion. For such processes, a priori knowledge will result in a model structure involving partial differential equations (PDE) [17]. A common approach for calibrating and validating PDE models is to perform using spatial discretization to obtain an ordinary differential equations (ODE) model [21,22]. Unfortunately, if the physically derived PDE model structure is completely replaced by an approximate ODE model, then important information about the underlying process has been discarded [17]. On the other hand, if the ODE model is used only for calibration and validation, then the model error introduced by spatial discretization can easily lead to falsely validated PDE models [17]. As a simple example of the latter, it was shown in [21] how finite difference discretization introduces artificial diffusion effects in the ODE model, artifacts that cannot be distinguished from the physical diffusion modeled in the PDE structure. A robust identification scheme was proposed in [22] for distributed parameter systems based on integrating the spatial discretization with calibration and validation of the ODE model. This integration is made feasible by estimating the model error imposed by discretization and formulating simple hypothesis tests relating the parameters of the discretization mesh to the estimated error. The proposed method ensures that there is no interference between model reduction errors and model-data discrepancies. In general, PDE models are not suitable for direct real-time applications without spatial discretization. There are two-dimensional approaches available [23] that, it is claimed to produce real-time PDE models suitable for use in operation training and for control testing design. However, there is the inevitable loss of detail (both internal detail and input output relationships) and a requirement for experimental data in order to identify unknown parameters [23].

2.3. Hybrid modeling

Thanks to using a hybrid model, the predictions tend towards results obtainable from a first-principle model when new operating conditions are encountered and additionally, the data-driven models are used when in already encountered and known operating conditions [18]. There are two major methodologies: the "serial approach" and the "parallel approach".

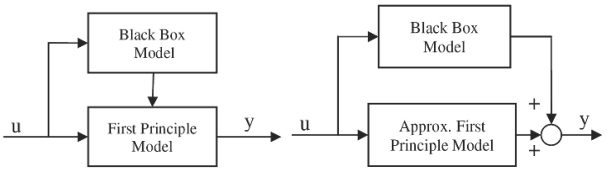

Fig 3. Block scheme representations of three basic structures of nonlinear grey-box model [19]; Hammerstein model (a); Wiener model (b); general model (c)

The serial approach uses a data-based model to construct missing inputs, or in the serial hybrid modeling approach, nonlinear system identification methods are used to model parameters in the first-principle models, and the first-principle model is used to model the system [18]. These parameters may be unknown, unmeasurable, change with time, or otherwise uncertain [18]. In the parallel hybrid modeling approach, a nonlinear system identification method is used to predict the residuals not explained by the first principle model. When in operation, the predicted residuals are added to the first principle model output, resulting in a total prediction that is much closer to the actual system.

The paper [18] presents a physical model and a parallel hybrid modeling architecture, both of which are capable of modeling a heat exchanger. The physical modeling did not perform well at all steady state conditions, therefore the neural network standard architectures (multi-layer perceptron) were used to improve model performance. This allows the hybrid models to revert back to the first-principle models when new conditions are encountered [18]. For comparison purposes, the authors will develop the hybrid serial model and its performance will be compared to the parallel hybrid model [18].

A novel cost-effective evolutionary hybrid model identification technique was presented in [24]. It was proposed to utilize the physical law based white-box dominated global

structure, with local black-boxes to include immeasurable and local nonlinearities of a system. The method offers an accurate, uncluttered and robust model with insightful representation of nonlinearities, which cannot be matched by conventional means [24]. It was shown that the hybrid model identification may fail by conventional gradient guided model fitting methods, but can be easily established through global optimization techniques by evolutionary computing [24]. The evolution can start from empirical models, making the best use of existing knowledge on a practical system [24]. In addition, the evolutionary technique is capable of accommodating multiple objectives to examine different trade-offs between the model complexity and fitting accuracy before final determination of a suitable hybrid model structure with optimized parameters. Two practical applications have shown the good feasibility and accuracy of the hybrid models [24].

An application example of the serial approach is the model development process of an experimental batch distillation column [25]. The moderately complex model describes the product quality and production over time including the start-up phase of the distillation column. The model structure is based on a simple physical framework, which is augmented with fuzzy logic. This provides a way to use prior knowledge about the dynamics, which have a general validity, while additional information about the specific column behavior is derived from measured process data [25]. The model framework is applicable for a wide range of columns operating under a certain control policy. The model framework for the particular column under study makes a priori assumptions about the specific behavior superfluous. In addition, a detailed description of the internal dynamics is not required, which reduces modeling effort. Three different hybrid model structures are compared; the model that uses the available sources of information most effectively can be used to simulate production including part of the start-up by applying constant quality control [25]. Another interesting example of the hybrid parallel approach is composition of a first-principle model derived from dynamic mass, energy and momentum balances used in the Globally Linearizing Control (GLC) strategy capable of using nonlinear process models directly [26]. When the process is not perfectly known, the unknown parts of the first-principle model should be represented by black-box models, e.g. by neural networks. In this paper, it is shown that the first principles part of the model determines the dominant structure of the controller, while the black-box elements of the hybrid model are used as state and/or disturbance estimators. For the identification of the neural network elements of the hybrid model a sensitivity approach based algorithm has been developed [26].

3. Formulation of FPDD model

This section introduces FPDD model referring to previous section where general classification overview was given.
Process of formulation of a first-principle data-driven model consists of three phases, namely (i) formulation of mathematical representation of first-principle laws, (ii) model adjustment and calibration process, and (iii) model evaluation phase.

3.1. Definition of an FPDD model

To derive first-principle mathematical representation of the technical system, three important aspects have to be considered; these are (i) system isolation, (ii) inputs-outputs selection and (iii) model economy. System isolation allows various types of interactions existing between the system and its surrounding environment to be discerned. System isolation enables only the most relevant and important interactions to be selected and represented in the form of inputs and outputs. Using principles of isolation and selection, a model is always simplified in accordance to the purpose of modeling. The number of inputs and outputs of a model is limited due to economical-type constraints (model economy). The principle of economy implies the simplicity of a structure and that a minimal number of parameters and state variables are considered.

A FP model requires an efficient modeling and simulation language to perform simulation. This kind of language differs from general-purpose programming language like FORTRAN or C in a functionality efficiently supporting of continuous and/or discrete event simulations. Nevertheless, a dedicated software environment is required to execute simulation and therefore, modeling and simulation languages are frequently integrated with simulation environments, e.g. Matlab/Simulink [27], or developed independently from the simulation environment, e.g. Modelica. In this respect, Modelica is a language that enables creation of open code, which can be executed in numerous third party simulation environments like CATIA Systems, Dymola, LMS Imagine.Lab AMESim, JModelica, MapleSim, MathModelica, OpenModelica, SCICOS, SimulationX.

3.2. Model adjustment and calibration

The most important phase when formulating the model is its calibration, which consists of (i) selection of representative data sets, including a sufficiently broad operating range that the model proved could work correctly using a data fit measure and (ii) adjustment of model parameters to fit a model response to data. First-principle models are frequently adjusted by a trial-and-error approach manipulating the values of parameters as a result of performed sensitivity analysis with the model, i.e. a series of model responses corresponding to a combination of values of selected model parameters. Nevertheless, such an approach can lead to non-optimal and non-repeatable results due to a lack of any systematic approach in making changes to the parameter values. In order to avoid the deficiencies of the trial-and-error approach, a formalized mathematical method using optimization techniques to minimize the error criterion, and find optimal values of tunable model parameters, is proposed in this respect. It is believed that the smaller the number of tunable parameters compared to the number of known parameters, the more accurate the model and the faster the convergence of the optimization algorithm used for model adjustment.

The procedure for adjusting a model consists of two in-a-loop phases: (i) simulation of a model by solving differential equations numerically, and (ii) numerical minimization in the parameter space with respect to an error-related criterion function. The function describing the error has to be a positive and decreasing function of the differences between reference and modeled outputs. A model of a technical system to be calibrated is typically represented as a set of nonlinear state-space equations formulated in the continuous-time domain as

$$\begin{aligned} &\frac{d}{dt}x(t) = f\left(t, x(t), u(t), w(t); \theta\right) \\ &y(t) = h\left(t, x(t), u(t), v(t); \theta\right) \\ &x(0) = x_0 \end{aligned} \tag{1}$$

where vector f(.) is a nonlinear, time-varying function of the state vector x(t) and the control vector u(t), while vector h(.) is a nonlinear measurement function; w(t) and v(t) are sequences of independent random variables and θ denotes a vector of unknown parameters. In nonlinear systems, the state vectors and the measurements vectors are not Gaussian distributed.

The sum of squared errors is used as an error criterion. This problem is known in numerical analysis as "the nonlinear least-squares problem" [27]. The objective of the estimation is to minimize the $V_N(\theta)$ error function by means of an iterative numerical technique. The error function $V_N(\theta)$ has the form

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N}\frac{1}{2}\varepsilon^2(t,\theta) \tag{2}$$

Three methods of minimizing the error function [27] are feasible as presented in Table 6. These are (i) direct search, (ii) first-order and (iii) second-order methods.

Direct search methods use only the value of the function to find the minimum. The first-order method uses the information provided by the first derivative (gradient) of the error function, while the second-order method uses both, information regarding the first and the second order derivatives (gradient and Hessian form) of the error function. First- and second-order methods update the estimates of the error function iteratively according to general scheme

$$\hat{\theta}^{(k+1)} = \hat{\theta}^{(k)} + \alpha_k \cdot f^{(k)} \tag{3}$$

where $\hat{\theta}^{(k)}$ denotes the k-th iteration point in the search, f(k) is a search direction based on the information about $V(\theta)$ acquired at previous iterations, and the sequence of positive scalars α_i determine velocity in which the value of $V(\theta)$ is decreased. In general, function $f(k)$ has a form

$$f^{(i)} = -\left[V''(\hat{\theta}^{(i)})\right]^{-1} V'(\hat{\theta}^{(i)})^T \tag{4}$$

Table 6.
Available methods to minimize the criterion function

Method	Algorithm	Description
Direct search methods	Simplex	The simplex algorithm is most useful for simple problems and might be faster than function minimization for models that contain discontinuities.
	Evolutionary algorithms	Genetic algorithm (GA) for instance, is a technique used in computing to find exact or approximate solutions of optimization and search problems.
First-order (gradient)	Gradient (steepest-descent)	The algorithm uses only the information about the gradient or the approximate gradient. The weakness of the method is that the algorithm may need multiple iterations before converging towards a local minimum.
Second-order (gradient and Hessian)	Newton-Raphson	Newton's method converges much faster towards a local minimum than gradient descent using full form of the Hessian. A number of quasi-Newton methods, where an approximation for the Hessian is used instead, are known. The Hessian is updated by analyzing successive gradient vectors instead.
	Gauss-Newton	The Gauss–Newton (GN) method is derived by ignoring the second-order derivative terms (the second term in this expression) approximating the Hessian with the form given by Eq. 6.
	Damped Gauss-Newton (Adaptive Gauss-Newton)	If the term $\mu_N^{(i+1)}$ in Eq. (7) is not equal to unity then the equation describes the so-called damped Gauss-Newton algorithm.
	Levenberg-Marquardt	The Levenberg-Marquardt algorithm (LMA) interpolates between the Gauss–Newton algorithm (GNA) and the method of gradient descent. The LMA is more robust than the GNA, which means that in many cases it finds a solution even if it starts very far from the minimum. On the other hand, when well-behaved functions and reasonable starting parameters are considered, the LMA tends to be slower than the GNA.

The first and second derivatives are obtained as follows

$$V'_N(\theta) = -\frac{1}{N}\sum_{t=1}^{N}\varepsilon(t,\theta)\psi(t,\theta) \tag{5}$$

$$V''_N(\theta) = \frac{1}{N}\sum_{t=1}^{N}\psi(t,\theta)\psi^T(t,\theta) - \frac{1}{N}\sum_{t=1}^{N}\psi'(t,\theta)\varepsilon(t,\theta) \tag{6}$$

where $\psi'(t,\theta)$ denotes the d-by-d Hessian matrix of $\varepsilon(t,\theta)$, d = dim(θ), cf. [27].

A general family of iterative search routines is described by the scheme

$$\hat{\theta}_N^{(k+1)} = \hat{\theta}_N^{(k)} + \mu_N^{(k+1)}\left[R_N^{(k)}\right]^{-1}V'_N(\hat{\theta}_N^{(k)}, Z^N) \tag{7}$$

where $R_N^{(k)}$ is a d-by-d matrix, which modifies the search direction (changes the search direction according to user preferences), while the step size $\mu_N^{(k+1)}$, is chosen so that satisfied in an inequality

$$V_N(\hat{\theta}_N^{(k+1)}, Z^N) < V_N(\hat{\theta}_N^{(k)}, Z^N) \tag{8}$$

It is often necessary to use variable step length to improve the convergence of the algorithm; a sequence of scalars $\mu_N^{(k+1)}$ is used to control the step length. If the term $\mu_N^{(k+1)} \neq 1$, equation (7) describes the so-called damped Gauss-Newton algorithm.

The form of $R_N^{(i)}$ enables different optimization schemes to be classified as [27]:

(i) Gradient or steepest-descent algorithm

$$R_N^{(k)} = \mathbf{I} \tag{9}$$

(ii) Newton-Raphson algorithm

$$R_N^{(k)} = V''_N(\theta) \tag{10}$$

(iii) Gauss-Newton algorithm

$$R_N^{(k)} = H_N\left(\hat{\theta}_N^{(k)}\right) \tag{11}$$

where $H_N(\hat{\theta})$ is the simplified form of the second derivative of error function.

$$V''_N(\theta, Z^N) \approx \frac{1}{N}\sum_{t=1}^{N}\psi(t,\theta)\psi^T(t,\theta) \equiv H_N\left(\hat{\theta}\right) \tag{12}$$

The simplification is feasible under assumption that the prediction errors, in the vicinity of the minimum, are independent of θ, i.e. $\varepsilon(t,\theta_0) = e_0(t)$.

(iv) Levenberg-Marquardt algorithm (a subclass of regularization techniques)

$$R_N^{(k)}(\lambda) = \frac{1}{N}\sum_{t=1}^{N}\psi(t,\hat{\theta}_N^{(k)})\psi^T(t,\hat{\theta}_N^{(k)}) + \lambda\mathbf{I} \tag{13}$$

where λ is a positive scalar used to control the convergence in the iterative scheme rather than the step size parameter (the method reduces to the Gauss-Newton method when λ=0).

3.3. Model evaluation

Validation enables to confront the model with independent data sets which were not used during the calibration phase.

4. Summary

The paper proposes and demonstrates a first-principle data-driven approach towards modeling of technical systems. The FPDD model offers physical insight and sufficient numerical performance to be applicable in understanding underlying physical phenomena, designing control systems, diagnosing and optimizing processes. FPDD models can be used in many areas where physical understanding is critical, e.g. design of new products or early warning diagnostics of large industrial installations. The model is represented by nonlinear state-space equations having geometrical and physical parameters deduced from available documentation, and adjustable phenomenological parameters (i.e. friction, leakage coefficients) that are estimated from measurement data.

This paper discusses three modeling approaches intended for technical systems, namely (i) constrained black-box together with semi-physical modeling, (ii) analytical modeling, and (iii) hybrid modeling. FPDD model was proposed by contrast to the presented taxonomy. A process of formulation of a FPDD model, adjustment and calibration is discussed as well.

The second part of the paper is focused on estimation of FPDD model parameters.

Acknowledgements

The author gratefully acknowledges the financial support of the research project N N502 087838 funded by the Polish Ministry of Science (MNiI).

References

[1] K. Białas, Synthesis of mechanical systems including passive or active elements reducing of vibrations, Journal of Achievements in Materials and Manufacturing Engineering 20 (2007) 323-326.

[2] M. Annarumma, A. Naddeo, M. Pappalardo, Software independence: impact on automotive product development process Journal of Achievements in Materials and Manufacturing Engineering, 31/2 (2008) 725-735.

[3] T. Dzitkowski, A. Dymarek, Design and examining sensitivity of machine driving systems with required frequency spectrum. Journal of Achievements in Materials and Manufacturing Engineering 26/1 (2008) 49-56.

[4] A. Naddeo, M. Annarumma, M. Pappalardo, N. Cappetti, State of the art of the passive pedestrian safety simulation, Journal of Achievements in Materials and Manufacturing Engineering, 30/1 (2008) 51-58.

[5] S. Żółkiewski, Analysis and modelling of rotational systems with the Modyfit application, Journal of Achievements in Materials and Manufacturing Engineering 30/1 (2008) 59-66

[6] K. Białas, Comparison of passive and active reduction of vibrations of mechanical systems, Journal of Achievements in Materials and Manufacturing Engineering 18 (2006) 455-458.

[7] A. Buchacz, Influence of piezoelectric on characteristics of vibrating mechatronical system, Journal of Achievements in Materials and Manufacturing Engineering 17 (2006) 229-232.

[8] A. Buchacz, S. Żółkiewski, Dynamic analysis of the mechanical systems vibrating transversally in transportation, Journal of Achievements in Materials and Manufacturing Engineering 20 (2007) 331-334.

[9] J. Świder, G. Wszołek, K. Foit, P. Michalski, S. Jendrysik, Example of the analysis of mechanical system vibrations in GRAFSIM and CATGEN software, Journal of Achievements in Materials and Manufacturing Engineering 20 (2007) 391-394.

[10] J. Świder, A. Baier, Aided design, virtual development and testing of modern control systems, Proceedings of the 11th International Scientific Conference "Achievements in Mechanical and Materials Engineering" AMME'2002, Gliwice-Zakopane, 2002, 541-544 (in Polish).

[11] J. Świder, G. Kost, J. Gorczyński, Process control technology using logic controllers PLC, Proceedings of the 11th International Scientific Conference "Achievements in Mechanical and Materials Engineering" AMME'2002, Gliwice-Zakopane, 2002, 549-552 (in Polish).

[12] J. Świder, R. Zdanowicz, Application of PLC controller in the systems simulation model Proceedings of the 12th International Scientific Conference "Achievements in Mechanical and Materials Engineering" AMME'2003, Gliwice-Zakopane, 2003, 971-974 (in Polish).

[13] B. Sohlberg, E.W. Jacobsen, Grey Box Modeling – Branches and Experience, Proceedings of the 17th World Congress The International Federation of Automatic Control Seoul, Korea, 2008, 1235-1247.

[14] L. Ljung, System identification - Theory for the User, Prentice-Hall, USA, 1999.

[15] G.W. Irwin, K. Warwick, K.J. Hunt, Neural network applications in control. The Institution of Electrical Engineers, Short Run Press, Exeter, 1995.

[16] K. Li, S. Thompson, Developing a NO, emission model for a coal-fired power generation plant using artificial neural networks. UKACC International Conference on Control, Cambridge, 2000, 347-352.

[17] T. Bohlin, S.F. Grabe, Issues in nonlinear stochastic grey box identification, International Journal Of Adaptive Control and Signal Processing 9 (1995) 465-490.

[18] R.L. Penha, J.W. Hines, Hybrid System Modeling for Process Diagnostics, Proceedings of MARCON 2002, Knoxville, TN, 2002, 242-248.

[19] G. Schothorst, Modeling of Long-Stroke Hydraulic Servo-System for Flight Simulator Motion Control and System Design, PhD Thesis, Technical University of Delft, 1997.

[20] T. Bohlin, Practical Grey-box Process Identification: Theory and Applications (Advances in Industrial Control), Springer-Verlag, London, 2006.

[21] J. Funkquist, Grey Box Identification of a Continuous Digester - A Distributed Parameter Process. Control Engineering Practice 5 (1997) 919-930.

[22] Y. Liu, Grey Box Identification of Distributed Parameter Systems. Doctoral Thesis. Signals, Sensors and Systems, KTH, Stockholm, Sweden, 2005.

[23] G. Ferretti, L. Piroddi, Estimation of NO, emissions in thermal power plants using neural networks, ASME Journal of Engineering for Gas Turbine and Power 123 (2001) 465-471.

[24] K.C. Tan, Y. Li, Grey-box model identification via evolutionary computing. Control Engineering Practice 10/7 (2002) 673-684.

[25] P.F. Lith, B.H.L. Betlem, B. Roffel, Combining prior knowledge with data driven modeling of a batch distillation column including start-up. Computers and Chemical Engineering 27 (2003) 1021-1030.

[26] J. Madar, J. Abonyi, F. Szeifert, Feedback Linearizing Control Using Hybrid Neural Networks Identified by Sensitivity Approach, Engineering Applications of Artificial Intelligence 18/3 (2006) 343-351.

[27] Matlab/Simulink package documentation, The Math Works Inc., Natick 1998.