33rd European Modeling & Simulation Symposium
18th International Multidisciplinary Modeling & Simulation Multiconference

ISSN 2724-0029 ISBN 978-88-85741-57-7 © 2021 The Authors.
doi: 10.46354/i3m.2021.emss.012

An integrated framework for co-simulation of white-box models and black-box models

Wenjia Zhang[1], Wenzheng Liu[1] and Heming Zhang[1,*]

[1]Tsinghua University, Shuangqing Road, Beijing, 100084, China

[*]Corresponding author. Email address: hmz@mail.tsinghua.edu.cn

Abstract

Integration of heterogeneous models can achieve interconnection between multiple types of simulation systems and realize reusability of model components. Recently data-driven modeling is becoming more and more common with the popularity of machine learning. It is a representative of black-box models which are totally dependent on data and need no disciplinary knowledge. From this perspective, models can be divided into white-box models, grey-box models and black-box models. Few researchers have considered the integrated issue under this mode. In this paper, we propose an integrated framework for scenarios where white-box models and black-box models are both involved. We discuss the structures of corresponding proxy models and then introduce a modified advancing strategy for general optimistic methods. It can greatly avoid possible rollback for black-box models and achieve efficient simulation by adjustment of simulation sequence.

Keywords: Model integration; Black-box models; White-box models; Modeling and simulation

1. Introduction

In recent years, due to the advent of Industry 4.0, modeling and simulation have become an important means to support the design and development of complex products. Additionally, the concept of cyber-physical system (CPS) further motivates researchers to combine computing, network and physical environments to achieve real-time perception, dynamic control, and information services of large engineering systems. In fact, a large engineering system is usually composed of heterogeneous models from different disciplines. To apply CPS, the integration of physical model should also be concerned. Multidisciplinary models with diverse sources and complicated coupling relationships lead to the increase in complexity of integrated modeling and simulation (Dolk and Kottemann, 1993). Proper design of synchronization methods and advancing strategy has been a critical problem.

From the perspective of the way system states change, any systems can be generally divided into continuous systems and discrete event systems. They correspond to different simulation advancing methods. For continuous cases, R.Kubler proposed iterative method and filter method, solving the problem of instability when an algebraic loop exists (Kübler and Schiehlen, 2000); Lin discussed the stage data synchronization method with a finer granularity (Lin, 2006); Liang introduced major step method and convergent integration step method, which takes both simulating speed and accuracy into account (Liang, 2009; Liang et al., 2011). For discrete cases, David presented virtual clock management method and allowed probable simulation rollback (Jefferson, 1985); K.venkatesh proposed global-clock controlling method, keeping all local clocks the same value at every time (K.Venkatesh, 1986); B.D.Lubachevsky put forward the static time-window method to determine upcoming events (Lubachevsky, 1989) and L.M.Sokol proposed the dynamic time-window method based on that (Sokol et al., 1991). All methods mentioned above can achieve reasonable simulation results within their range.

But from another perspective, models can be divided into white-box model, grey-box model and black-box model due to different cognitive levels (Beghi et al., 2007; Leifsson et al., 2008; Duun-Henriksen et al., 2013). White-box models are based on prior knowledge, which usually involve a set of deterministic functions, while black-box models are obtained from data and built on statistical information(Garcia et al., 2008). Grey-box models are combinations of both types. Owing to different internal characteristics, these models also need to be treated differently when performing model integration. It is unreasonable to classify them as continuous or discrete in general. However, researchers rarely consider this issue or propose an integrated framework under this mode. With the popularity of machine learning, data-driven modeling will become more and more common. It is necessary to put forward a feasible integrated framework for co-simulation of white-box, grey-box, and black-box models. Since grey-box model is an intermediary between white-box and black-box models, it is more complicated and will be discussed in the future research. In this paper, we will mainly focus on the integration of white-box models and black-box models.

The rest of paper is organized as follows: the detailed descriptions of white-box models and black-box models are discussed in section2, proposed integrated framework is introduced in section3, case study is shown in section4, and conclusions are summarized in section4.

2. Model descriptions

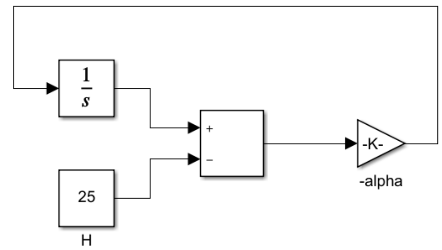
2.1. White-box models

White-box models refer to models based on fundamental principles. They are built according to expertise in different disciplines, and usually comprise a set of deterministic functions, including ordinary differential equations (ODEs), differential algebraic equations (DAEs), etc. White-box models are models of professional knowledge. Thus, configuration parameters or system states usually have specific physical meanings.

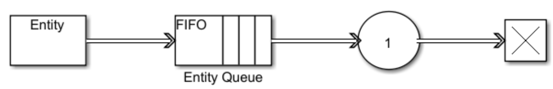
Take Newton cooling model and M/M/1 queuing model as examples for continuous cases and discrete cases. They are built in Simulink as figure 1 shows. The former model is based on Newton's law of cooling and the latter model is based on queuing theory.

2.2. Black-box models

Different from white-box models, black-box models are entirely built on data, with no need for any strict disciplinary theorems. Black-models tend to fit or imitate the relationship between input and output with techniques in data science. It is a useful method when modeling system with not fully understood mechanism, but needs a large amount of data to support. Since sometimes the states in these models merely refer to some computed results during data processing, it is uncertain that states in black-box models have physical meanings.

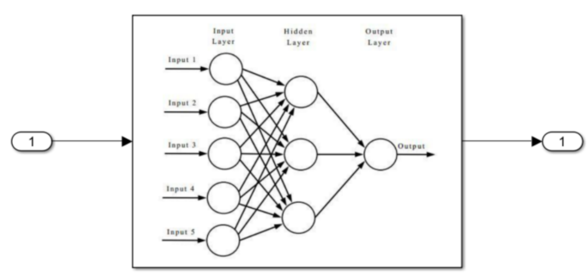


(a) Newton cooling model: $T'(t) = -\alpha(T(t) - H)$

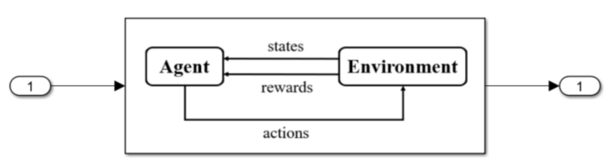


(b) M/M/1 queuing model

Figure 1. Examples of white-box models



(a) fitting model based on neural networks



(b) decision-making model based on RL

Figure 2. Examples of black-box models

For example, fitting models based on neural networks and decision-making models based on reinforcement learning can be seen as black-box models, shown in figure 2. Both models are trained from experimental data, with no prior knowledge.

3. Integrated framework

When performing model integration and collaborative simulation, each model needs a proxy model to connect the model from different sources with integrated environment. All proxy models are connected to data bus and control bus, and an integration console is responsible for management of them. To be more specific, the integration console needs to regulate the simulation process of each subsystem and conduct necessary data analysis and storage. The overall integrated framework is illustrated as figure 3.

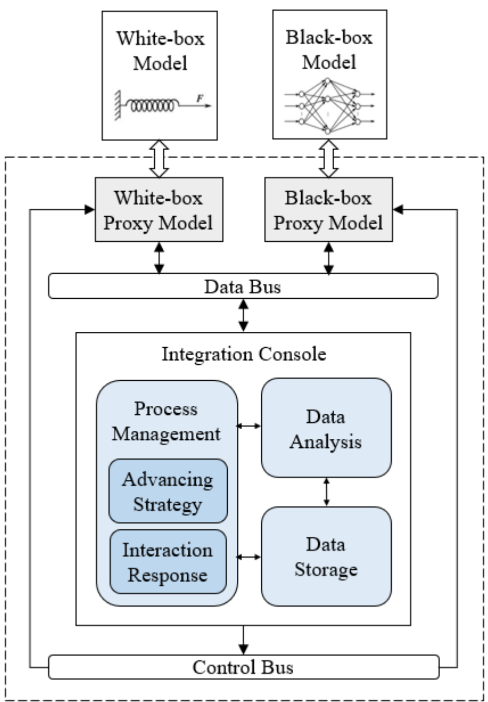


Figure 3. Illustration of overall integrated framework for white-box models and black-box models

3.1. Proxy models

Construction of proxy models is the basis of integration process. The different characteristics of white-box models and black-box models are also reflected in their proxy models.

For white-box models encapsulation, the structure of model is transparent, and all the parameters and system states are readable. It is also allowed to artificially change the value of inputs and states. Thus, there should be three data ports in proxy model connecting the original model to data bus, namely input port, output port and state port. Each proxy model also has a control port to transfer control command and related parameters. Due to the public feature, white-box model can achieve simulation rollback when an optimistic simulation method is adopted, which will be discussed in the next subsections. The control port needs to support control commands involving data control (get or set the value of inputs and states), general process control (start/pause/stop), run-to-time control, and step-size control. Most simulation software will directly provide external interfaces of these four instructions. The construction structure and bus hierarchy of white-box proxy models are shown in figure 4.

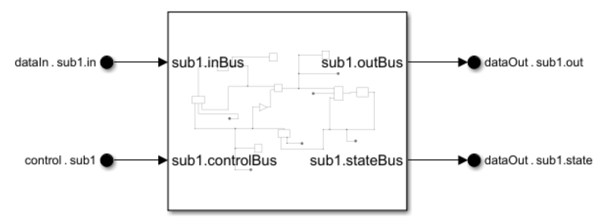


Figure 4. Proxy model for white-box models

For black-box models encapsulation, since there is no clear necessity for accessibility, states are usually maintained inside the models and are unreadable. It only has input port and output port connected to data bus in black-box proxy models. Therefore, it does not support simulation rollback, either. The control port is similar to white-box models, involving four general instructions: data control(get or set the value of inputs), general process control (start/pause/stop), run-to-time control, and step-size control. The construction structure and bus hierarchy of black-box proxy models are shown in figure 5.

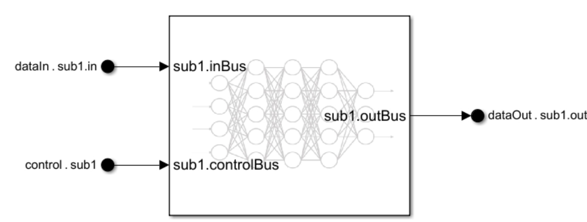


Figure 5. Proxy model for black-box models

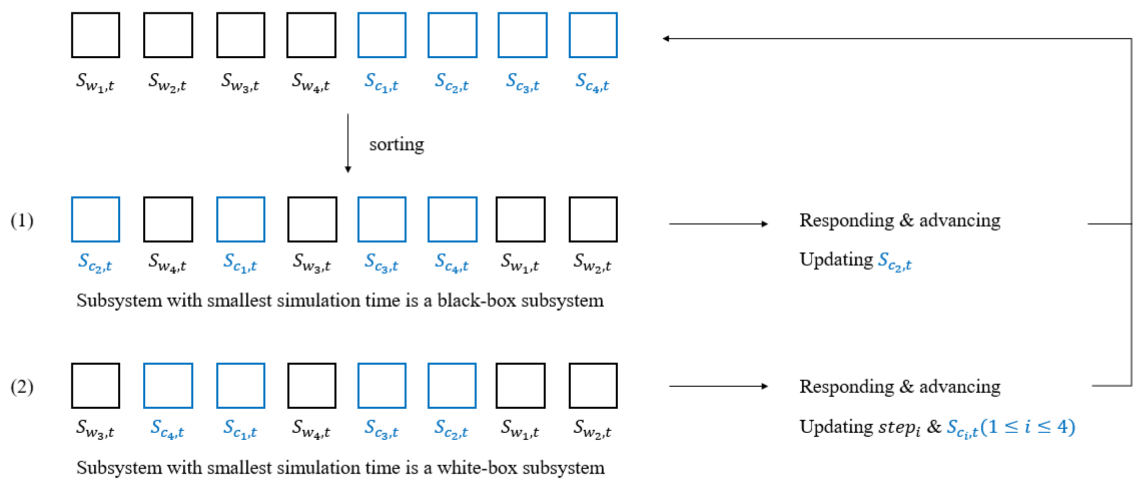


Figure 6. Illustration of the modified advancing strategy. Take a global system which consists of four white-box models and four black-box models for an example. Black squares represent white-box models, and blue squares represent black-box models.

3.2. Advancing strategy

Generally, advancing strategy can be divided into conservative ways and optimistic ways. Conservative methods usually require application-specific information, while optimistic methods have much more flexibility. For conservative ways, each step forward is cautious. Only when the subsystem determines that there will be no possible errors or the errors can be approximately ignored, it can advance to next step. For optimistic ways, subsystems can advance under broader constraints and when errors detected, they are supposed to perform simulation rollback according to the states before errors happen. When a subsystem needs rollback, it should not only restore the states itself, but also revoke the impact on other subsystems during this period. The completion of simulation rollback requires cooperation of all instructions. It is based on a large memory of previous states and happened instructions of each subsystem.

As shown in the previous subsection, due to the differences in model encapsulation, advancing methods might have to be modified in some cases. For optimistic advancing methods, such as virtual clock method, white-box proxy models also follow the normal form of original methods. But for black-box proxy models, due to the inaccessibility of system states, it is impossible to achieve simulation rollback. There should be a mechanism to avoid rollback and coordinate the simulation process for these models.

To solve this problem, we introduce a modified method in which subsystems are promoted by step-size to achieve timing consistency. Previous states and happened instructions of white-box models are saved for the convenience of future rollback. The core innovation of this method lies in the determination of subsystem advancing order. There is no change in the way responding the interactions.

At the beginning, each white-box subsystem is advanced one-step. Suppose the step-size of each white-box subsystem is $(step_1, step_2, \ldots, step_n)$, and $step_i = max(step_1, step_2, \ldots, step_n)$. For each black-box subsystem with simulation time $S_{b,t}$, define $S_{c,t} = S_{b,t} + step_i$ as its secure simulation time. Then get the current simulation time of white-box subsystems, denoted as $S_{w_1,t}, S_{w_2,t}, \ldots, S_{w_n,t}$, and the secure simulation time of black-box subsystems, denoted as $S_{c_1,t}, S_{c_2,t}, \ldots, S_{c_m,t}$.Sort the time series $(S_{w_1,t}, S_{w_2,t}, \ldots, S_{w_n,t}, S_{c_1,t}, S_{c_2,t}, \ldots, S_{c_m,t})$, and the subsystem with smallest time is selected as next proceeding subsystem.If next subsystem is a black-box subsystem, respond the interactions from other subsystems, advance it, and then update its secure simulation time; otherwise, after response and advancing, update $step_i$ and all the secure simulation time of black-box subsystems. Continue to repeat above sorting, advancing and updating process until the end of the simulation, as figure 6 shows.

Actually, $step_i$ represents the secure distance for black-box models and it can also be manually specified. The larger value indicates larger secure interval and fewer possible errors caused by non-rollback in black-box models, but it will require longer time in return.

In this way, black-box models are always a secure distance behind the white-box models, which provides flexibility for white-box models to perform rollback and greatly avoid the situation required rollback for black-box models. It also prevents the occurrence of deadlock. This method can arrange the simulation sequence of non-rigid system efficiently. But for rigid system, for instance, if the average step-size of a certain black-box model is far greater than white-box models, apart from possible problems from numerical integration algorithm, this strategy will fail to keep it behind the general simulating process and thus lose effectiveness.

4. Case study

Since the 21st century, autonomous vehicles are showing a practical trend. Many large technology companies have launched autonomous driving related businesses. Their goal is to be operated automatically and safely without any human operation. Autonomous vehicles are extremely complex products that integrate artificial intelligence, visual computing, global positioning systems, etc., and need a great many sensors to support. Virtual modeling and simulation can assist the development process, making it safe, economical and environmental friendly.

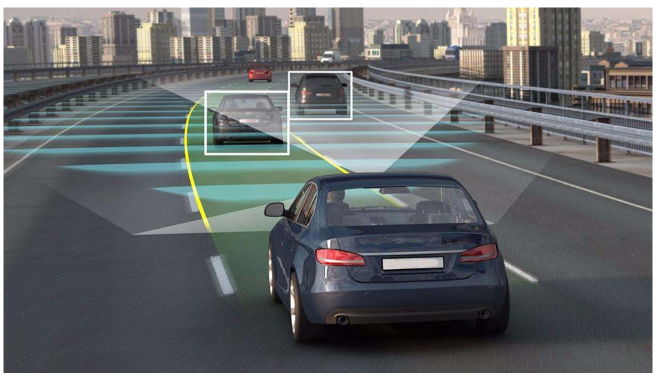


Figure 7. An autonomous vehicle drving on the road[1]

In this scenario, there are usually many white-box models according to mechanics, dynamics, thermodynamics, etc., to achieve precise control of cars, and some black-box models which can perceive higher-level information from obtained sensor data, providing stronger support for decision-making. Each model can be heterogeneous and from diverse sources.

Proposed framework can help tackle the integration problem of these models. Suppose that developers are testing the performance of an autonomous vehicle when smoothly driving on a straight road. In this case, take the integration of a white-box speed control model, a white-box gasoline consuming model and a black-box perceptual model as an example. The perceptual model will perceive environmental information, such as road and weather conditions. The speed control model will constantly modify the expected speed based on the information perceived and adjust the actual driving speed to expected value. The gasoline consuming model will calculate the gasoline consumption based on real-time driving speed and promote the fired power engine. According to proposed integrated framework, each model can be encapsulated and interconnected as figure 8 shows.

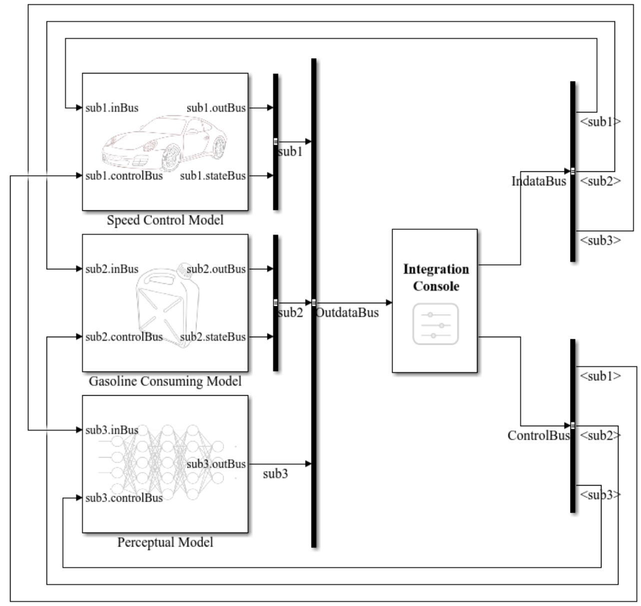


Figure 8. An integrated model from autonomous vehicles scene, which is composed of a speed control model, a gas consuming model and a perceptual model

Integration console is responsible for arranging the simulation sequence and mastering the integration process. It marks different type of subsystems from their bus structure, and then organizes simulation process according to the advancing strategy introduced in previous section. By this means, the global system can perform simulation correctly and efficiently with an integrated form.

5. Conclusions

In this paper, we propose an integrated framework for scenarios where white-box models and black-box models are both involved. We discuss the structures of corresponding proxy models and then introduce a modified advancing strategy for general optimistic methods. It can greatly avoid possible rollback for black-box models and achieve efficient simulation by adjustment of simulation sequence, but it is only suitable for non-rigid systems. Our framework is also demonstrated in the context of autonomous driving for an example. While machine learning and other data techniques are more and more widely used, research on integration of white-box models and black-box models is meaningful. For future research, additional consideration of grey-box models or design of advancing strategies for

1 The figure is from this website: http://k.sina.com.cn/article_6434159667_17f817c33001003rsi.html?from=auto&subch=bauto

rigid system might be the point.

6. Funding

This work is supported by the National Key R&D Program of China under Grant 2018YFB1701600.

References

Beghi, A., Liberati, M., Mezzalira, S., and Peron, S. (2007). Grey-box modeling of a motorcycle shock absorber for virtual prototyping applications. *Simulation Modelling Practice and Theory*, 15(8):894–907.

Dolk, D. R. and Kottemann, J. E. (1993). Model integration and a theory of models. *Decision Support Systems*, 9(1):51–63.

Duun-Henriksen, A. K., Schmidt, S., Roge, R. M., Moller, J. B., Norgaard, K., Jorgensen, J. B., and Madsen, H. (2013). Model identification using stochastic differential equation grey-box models in diabetes. *Journal of diabetes science and technology*, 7(2):431–440.

Garcia, J. D., Prada, L., Fernandez, J., and Nunez, A. (2008). Using black-box modeling techniques for modern disk drives service time simulation. *IEEE*.

Jefferson, D. R. (1985). Virtual time. *ACM Trans. Program. Lang. Syst.*, 7(3):404–425.

K.Venkatesh, T.Radhakrishnan, H. (1986). *Discrete event simulation in a distributed system*. IEEE Computer Society Press.

Kübler, R. and Schiehlen, W. (2000). Two methods of simulator coupling. *Mathematical and Computer Modelling of Dynamical Systems*, 6(2):93–113.

Leifsson, L. T., S?Varsdóttir, H., Siguresson, S. T., and Vésteinsson, A. (2008). Grey-box modeling of an ocean vessel for operational optimization. *Simulation Modelling Practice Theory*, 16(8):923–932.

Liang, S., Zhang, H., and Wang (2011). Combinative algorithms for the multidisciplinary collaborative simulation of complex mechatronic products based on major step and convergent integration step. *CHINESE JOURNAL OF MECHANICAL ENGINEERING -ENGLISH EDITION-*.

Liang, S. (2009). *Research on multidisciplinary collaborative simulation algorithm for complex product*. PhD thesis, Tsinghua University.

Lin, Y. (2006). *Research and implementation on collaborative simulation technology based on HLA*. PhD thesis, Tsinghua University.

Lubachevsky, B. D. (1989). Efficient distributed event-driven simulations of multiple-loop networks. *Communications of the ACM*.

Sokol, L. M., Weissman, J. B., and Mutchler, P. A. (1991). Mtw: an empirical performance study. In *1991 Winter Simulation Conference Proceedings.*, pages 557–563.