Received: 25 June 2021 | Accepted: 23 September 2021

DOI: 10.1002/pamm.202100238

Workflow concepts to model nonlinear mechanics with computational intelligence

Arnd Koeppe[1,2,*], **Franz Bamer**[1], **Michael Selzer**[2,3], **Britta Nestler**[2,3], and **Bernd Markert**[1]

[1] Institute of General Mechanics, RWTH Aachen University, Eilfschornsteinstr. 18, 52062 Aachen, Germany
[2] Institute of Applied Materials (IAM-CMS), Karlsruhe Institute of Technology (KIT), Straße am Forum 7, 76131 Karlsruhe, Germany
[3] Institute of Digital Materials Science (IDM), Karlsruhe University of Applied Sciences, Moltkestr. 30, 76133 Karlsruhe, Germany

Methods from computational intelligence, such as (artificial) neural networks, have become an active research direction in mechanics, leading to the development of intelligent constitutive models, surrogate models, and meta elements. Therein, many neural network architectures are inspired by mechanical domain knowledge in the form of physics-informed or physics-guided neural networks. Complementary approaches that systematically analyze and compare neural networks trained on mechanical data, i.e., physics-informing neural networks, have not yet been established. As a step in that direction, we propose a workflow concept to describe neural networks in mechanics, as well as a workflow concept to systematically search and train neural network architectures on mechanical data. The workflow concepts will be presented in the scope of Kadi4Mat. Following these workflow concepts, neural network design can be unified, compared, and interpreted, which enables explainable artificial intelligence for mechanics in future works.

1 Introduction

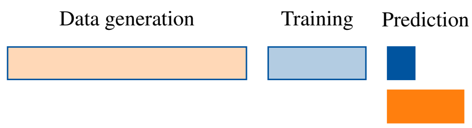
Model Order Reduction (MOR) techniques speed up numerical simulations, e.g., with Computational Intelligence (CI) enhanced surrogate models or proper orthogonal decomposition. The individual computations are accelerated, but an initial (computational) overhead that consists of three parts is often necessary. First, samples or snapshots need to be created from simulations or experiments to gain states of mechanical models. Second, the best surrogate model needs to be identified or designed, usually in an iterative process. Finally, once that surrogate model has been designed, it has to be trained or fitted on the available data. Due to those three contributions, the computational overhead often limits the practical application of MOR.

Therefore, this work aims at describing some concepts that reduce the computational overhead of MOR techniques leveraging the scientific workflow.

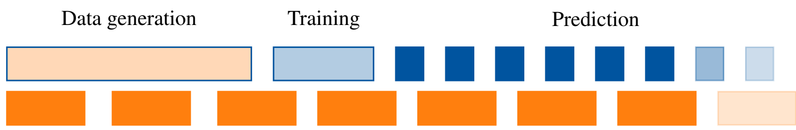
2 Workflow concepts to reduce computational overhead

In the following, we will tackle the three contributions — data generation, surrogate design, and surrogate training — to the computational overhead separately.

The computational overhead of data generation can be mitigated if multiple similar simulations need to be executed with only minor changes to the model. Data generation becomes a by-product of the first part of the original simulation study, as visualized in Figure 1. This approach can be used, e.g., in large parameter sensitivity studies [1] or Monte Carlo simulations [2], to share the overhead over the entire scope of all simulations.



(a) Qualitative comparison of computation times for a single simulation run (bottom) and a trained surrogate model prediction (top).



(b) Qualitative comparison of computation times for multiple simulation runs (bottom) and trained surrogate model predictions (top).

Fig. 1: Reusing neural networks in large-scale simulation studies mitigates the computational overhead of data generation and training.

* Corresponding author: arnd.koeppe@kit.edu

The second approach to reduce the computational overhead incurred by MOR focuses on the surrogate model design. Physics-informed neural networks [3–5] use mechanical knowledge to reduce the required amount of training data and help to find physically motivated neural network architectures. Likewise, surrogate model architectures can be designed by mimicking physical and numerical equations [6] or using self-design approaches [7]. Alternatively, by interpreting this design process as a search, algorithms such as Hyperband [8] can find optimal surrogate models to a given mechanical dataset. Such search algorithms can be implemented and supported by data management infrastructures, such as Kadi [9, 10], which allows automated harvesting of simulation and experimental data from diverse sources. By leveraging machine learning workflows implemented in data management platforms (Figure 2), specific search results, as well as data from previous studies, can be used to find better surrogate models. Both the design and search approaches help to derive mechanically motivated surrogate model architectures using existing data or expert knowledge.

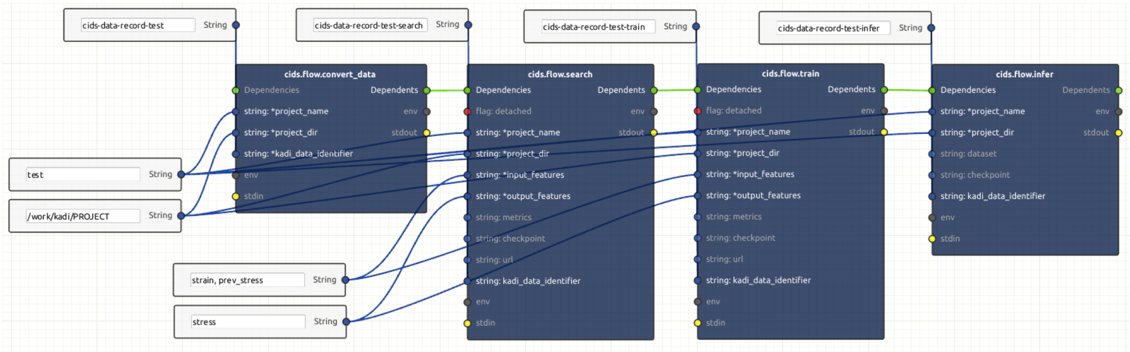


Fig. 2: A workflow of a neural network study on mechanical data in Kadi [9].

The third approach promotes reusing generic surrogate models that have been trained previously [6, 11]. As part of larger models in a Craig-Bampton approach, intelligent meta elements can predict the response of substructures without requiring additional training in new models [6, 11]. Thus, the computational overhead of data generation, surrogate design, and training can be fully eliminated.

3 Concluding remarks and outlook

This work briefly reviewed three approaches to reduce the computational overhead of MOR techniques. Employing the initial phase of large simulation studies for data generation offers a straightforward way to mitigate some computational overhead of surrogate modeling. Implementing the scientific workflow in data management platforms enables search methods to find advantageous surrogate model designs. Finally, reusing previously trained surrogate models in generic substructures, e.g., in intelligent meta elements, allows the effective elimination of the computational overhead.

Future publications will investigate search algorithms and data management to design physics-informing neural networks that help to explain and interpret mechanical behavior from datasets [12].

Acknowledgements The authors gratefully acknowledge financial support from the Federal Ministry of Education and Research (BMBF) in the project FestBatt (project number 03XP0174E) and from the Ministry of Science, Research and Art Baden-Württemberg in the project MoMaF - Science Data Center, with funds from the state digitization strategy digital@bw (project number 57). Open access funding enabled and organized by Projekt DEAL.

References

[1] A. Koeppe, C. A. Hernandez Padilla, M. Voshage, J. H. Schleifenbaum, and B. Markert, Manuf. Lett. **15**(January), 147–150 (2018).
[2] A. Koeppe, F. Bamer, and B. Markert, Acta Mech **230**(9), 3279–3293 (2019).
[3] M. Raissi, P. Perdikaris, and G. E. Karniadakis, Journal of Computational Physics **378**(February), 686–707 (2019).
[4] Y. Yang and P. Perdikaris, Journal of Computational Physics **394**(October), 136–152 (2019).
[5] G. Kissas, Y. Yang, E. Hwuang, W. R. Witschey, J. A. Detre, and P. Perdikaris, Computer Methods in Applied Mechanics and Engineering **358**(January), 112623 (2020).
[6] A. Koeppe, F. Bamer, and B. Markert, Computer Methods in Applied Mechanics and Engineering **366**(July), 113088 (2020).
[7] A. Fuchs, Y. Heider, K. Wang, W. Sun, and M. Kaliske, Computers & Structures **249**(June), 106505 (2021).
[8] L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, J. Mach. Learn. Res. **18**(1), 6765–6816 (2017).
[9] N. Brandt, L. Griem, C. Herrmann, E. Schoof, G. Tosato, Y. Zhao, P. Zschumme, and M. Selzer, Data Sci. J. **20**(1), 8 (2021).
[10] M. Selzer, N. Brandt, S. Ephraim, P. Zschumme, P. Altschuh, C. Herrmann, Y. Zhao, G. Tosato, and B. Nestler, Kadi4Mat, https://kadi.iam-cms.kit.edu/, 2020.
[11] A. Koeppe, F. Bamer, and B. Markert, PAMM **18**(1), e201800283 (2018).
[12] A. Koeppe, F. Bamer, M. Selzer, B. Nestler, and B. Markert, ArXiv210410683 Cs(May) (2021).

