Intelligent data collection for efficient model surrogating with deep learning

Advisors: Michael Bauerheim (Isae-Suparo), Guillaume Infantes (Jolibrain)

Contact: michael.bauerheim@isae-supaero.fr, guillaume.infantes@jolibrain.com

Place: Jolibrain office, 10 boulevard d'Arcole, Toulouse, France

Recent advances in Artificial Intelligence like machine learning based on deep neural networks allows the learning of any function of interest, given enough input/output samples. While the offline learning time can be long, the online computation of the output given the inputs takes constant time, generally very short. This paves a way for using such techniques for building surrogate models/functions where long computations are mandatory at high frequency. This is particularly true in elaborate physics models where iterative computation is necessary in every cell of a discretized space, for instance to solve local partial derivative equations as ones encountered in fluid motion, heat exchange among others.

Using surrogate models based on deep learning has already shown interesting results like in [TSSP16], but questions are still opened in order for such methods to spread out. While the learning phase of the neural network is well known, the problem of collecting the data using the original function before learning (or the loop including both of them) is much less investigated. In practice, it may have dramatic consequences for several reasons:

- Deep learning generally needs a lot of data as it does not take into account any modelling hypothesis. If the original function is expensive or slow to compute (for instance very precise simulator in fluid dynamics, or real environment), the global amount of data should be minimized.
- There is a trade-off between the quantity of data used for learning, and the quality of the surrogate model obtained, but it is difficult to assess since the various deep architecture properties are not completly modelled.
- The impact of the quality of the data used, particularly in terms of variety, is still not clearly understood: the data should cover all the cases, and certainly need to be more dense around sensitive places. Yet, methods to identify which data should be generate to complete the dataset are still missing.

Such problems have been studied as hyperparmeter optimization (parameters for data collection can be seen as hyperparameters), for instance with the pioneering ParamILS algorithm [HHLBS09]. Techniques have evolved, for instance, by adding a model of the error of the surrogate model like in the AutoML [FH18] framework, and/or by using the sequential nature of the hyperparameter search, like in the SMAC framework [HHLB11]. All these approaches consider the sequential problem of selecting good (hyper-)parameters, seeing the results in terms of error, and then selecting another set of good (hyper-)parameters and so on. Another approach is the reinforcement learning framework, which has shown very impressive results in the few last years [MKS+13, MGM+18, Ope18]. Such techniques can use deep learning techniques in order to build a model of the expected future errors after choosing some hyper-parameters.

This internship will investigate the first few steps towards algorithms and methodologies for intelligent data collection taking into account the criteria above, using different techniques going from statistical modelling to reinforcement learning. This will be applied to building surrogate models of the Poisson equation resolution used in fluid mechanics.

Applicant profile: Candidate should have high motivation for new challenges and innovative approaches, a background in machine learning and/or its mathemathical foundations, C++ and python development as well as linux environment. Knowledge in fluid mechanics and more genrally physics simulations would be most appreciated.

References

[FH18] Matthias Feurer and Frank Hutter. Hyperparameter optimization. In Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren, editors, *AutoML: Methods, Sytems, Challenges*, chapter 1, pages 3–37. Springer, December 2018. To appear.

[HHLB11] F. Hutter, H. H. Hoos, and K. Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *Proc. of LION-5*, page 507523, 2011.

[HHLBS09] Frank Hutter, Holger H. Hoos, Kevin Leyton-Brown, and Thomas Stutzle. ParamILS: an automatic algorithm configuration framework. *Journal of Artificial Intelligence Research*, 36:267–306, October 2009.

[MGM+18] Piotr Mirowski, Matthew Koichi Grimes, Mateusz Malinowski, Karl Moritz Hermann, Keith Anderson, Denis Teplyashin, Karen Simonyan, Koray Kavukcuoglu, Andrew Zisserman, and Raia Hadsell. Learning to navigate in cities without a map. *CoRR*, abs/1804.00168, 2018.

[MKS+13] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602, 2013.

[Ope18] OpenAI. Openai five. `https://blog.openai.com/openai-five/`, 2018.

[TSSP16] J. Tompson, K. Schlachter, P. Sprechmann, and K. Perlin. Accelerating Eulerian Fluid Simulation With Convolutional Networks. *ArXiv e-prints*, July 2016.