Towards an Operationally Meaningful, Explainable Emulator for the Boussinesq equation

Eloisa Bentivegna
IBM Research Europe
eloisa.bentivegna@ibm.com

ICLR 2021 Workshop
Deep Learning for Simulation (simDL)

Introduction

The demand for emulators of expensive computational models is rapidly surging, as search, optimization, and uncertainty quantification tasks become increasingly more relevant than the computation of single simulation configurations. I investigate the feasibility of an emulator for a 2D coastal inundation model, based on a Variational Autoencoder (VAE) architecture proposed recently. The VAE's ability to predict the relationship between the model controls and the observed water levels at a predefined location is tested, and its applicability towards the construction of an emulator is discussed.

Simulation model

The base configuration involves a two-level coastal profile, representing land and sea on either side of a straight stretch of coast, shielded in part by a straight, infinite-height levee parallel to the coast. The water height's initial configuration is that of a plane wave, also parallel to the coast. The simulation computes the evolution of the water level everywhere on the domain, as the initial profile marches inwards toward sthe coast and waves are scattered around the domain. The model specification is summarised in the table below, and a snapshot of the evolution is shown in Figure 1.

Parameter	Value
Domain boundaries	$[0, 50m] \times [0, 50m]$
Spatial resolution	$0.2m$
x coordinate of the levee, $\bar{x}_L$	$25m$
y extension of the levee, y_L	$[10m, 40m]$
x interval of sea-land transition region	$[26m, 36m]$
Initial x location of wave peak	$15m$
Coastal elevation	$0.5m$
Initial peak height	$0.55m$
Physical time of evolution	$20s$

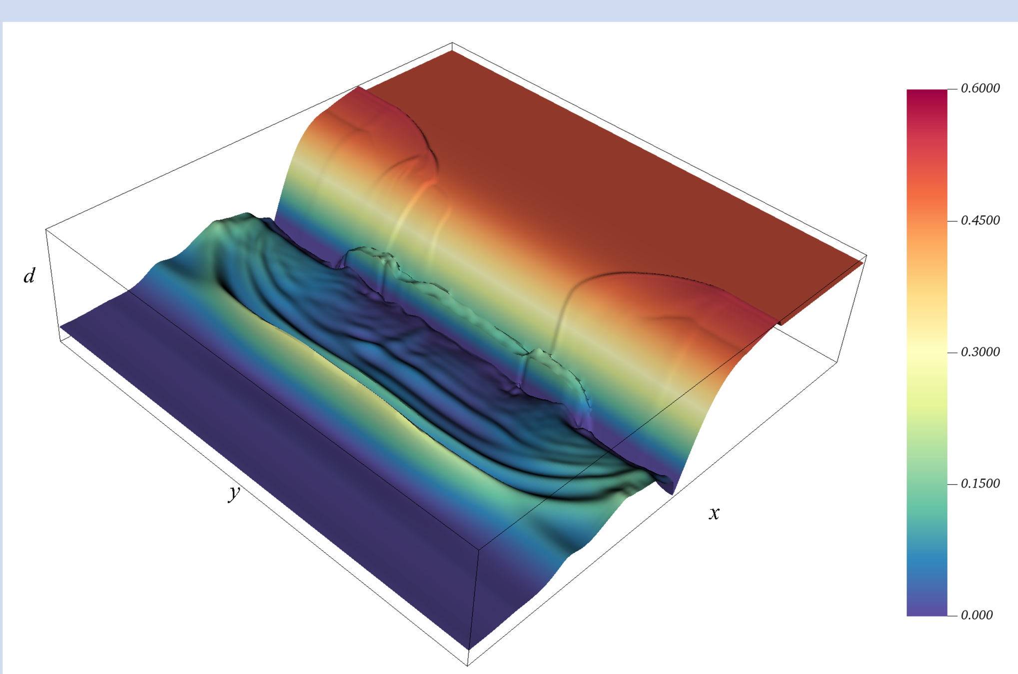

Figure 1

I construct a dataset of 12,006 simulations by perturbing this basic setup. The variation consists of a parametrized deformation of the levee, modelled by a sinusoidal displacement:

$$x_L = \bar{x}_L + a \sin(\pi n y_L)$$

A few, randomly chosen simulations are shown in Figure 2.

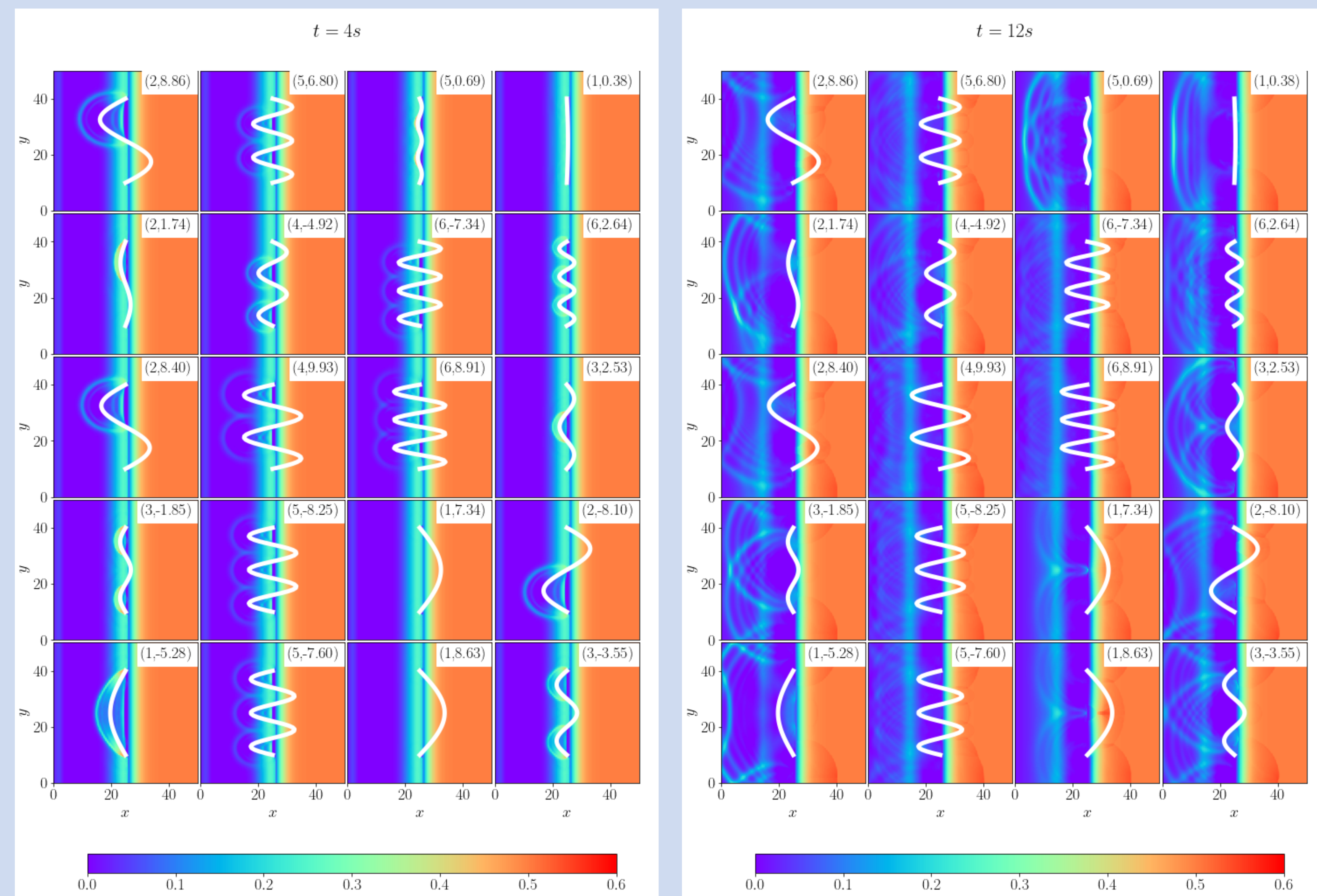

Figure 2

Results

I train a β-VAE with a 60-neuron input layer, a 1-neuron output layer, a two-level encoder and decoder of size n_{layer}, and a latent representation of size n_{latent} (the architecture corresponds to Figure 1(b) in Iten et al. (2020), except for the question neuron, which I omit). I illustrate the result of this procedure, for various values of n_{latent}, n_{layer}, and β, in the table below.

Configuration	N_{train}	n_{latent}	n_{layer}	β	Loss	Reconstruction RMSE	Prediction RMSE
a	10006	2	64	0	$1.(8) \cdot 10^{-4}$	$(6) \cdot 10^{-4}$	$(6) \cdot 10^{-5}$
b	10006	2	256	0	$(2) \cdot 10^{-4}$	$(6) \cdot 10^{-4}$	$2.(1) \cdot 10^{-5}$
c	10006	2	1024	0	$(3) \cdot 10^{-3}$	$(2) \cdot 10^{-3}$	$(5) \cdot 10^{-5}$
d	10006	1	64	0	$(5) \cdot 10^{-4}$	$(8) \cdot 10^{-4}$	$3.(0) \cdot 10^{-5}$
e	10006	5	64	0	$(6) \cdot 10^{-3}$	$(2) \cdot 10^{-3}$	$(7) \cdot 10^{-5}$
f	10006	10	64	0	$(2) \cdot 10^{-3}$	$1.(9) \cdot 10^{-3}$	$(6) \cdot 10^{-5}$
g	10006	2	64	0.2	$-(8) \cdot 10^{-2}$	$(2) \cdot 10^{-3}$	$(7) \cdot 10^{-5}$
h	10006	2	64	0.8	$-3.(9) \cdot 10^{-1}$	$1.(2) \cdot 10^{-3}$	$4.(1) \cdot 10^{-5}$
i	11886	2	64	0	$(3) \cdot 10^{-4}$	$(7) \cdot 10^{-4}$	$(8) \cdot 10^{-5}$

In Figure 3, the predictions of one of the trained networks for configuration **a** are shown.

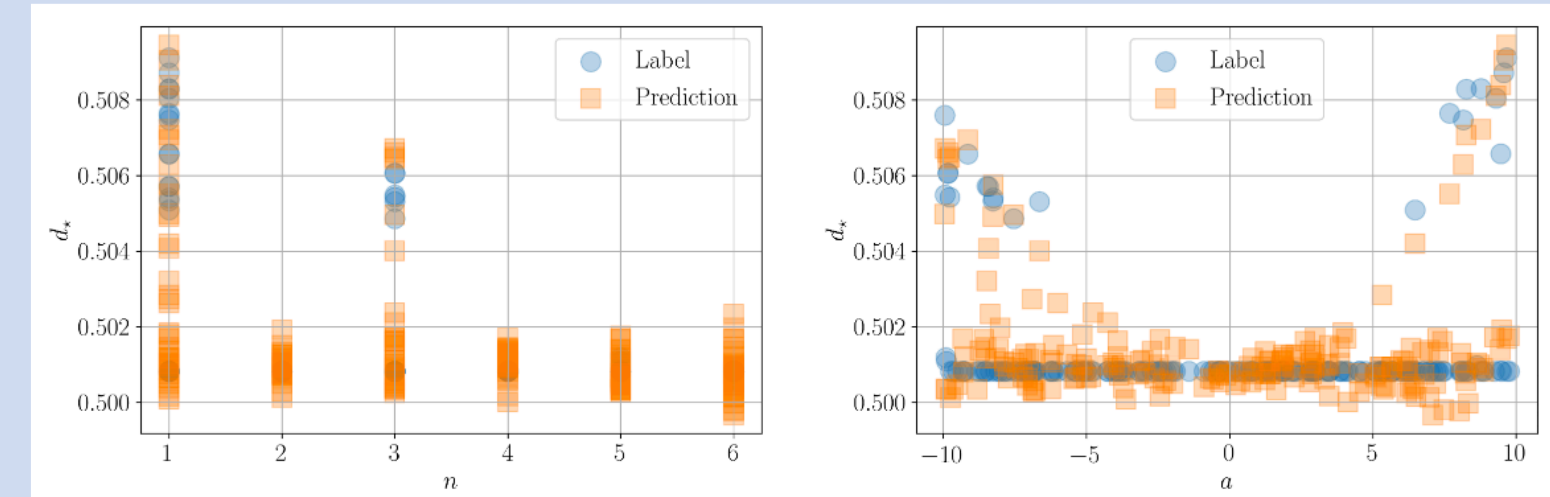

Figure 3

A key question to investigate is whether the latent representations in the trained network are able to highlight the principal features of the map between levee geometry and water height, and might potentially be used to model this relationship in an accurate, explainable, and generalizable way. In particular, it is interesting to observe the role of the parameter β, as this is supposed to influence the distribution of simulations in the latent space directly. In Figure 3, I plot the values of the network's latent neurons, r_1 and r_2, for all the simulations in two representative datasets: one from configuration **a** and one from configuration **h**.

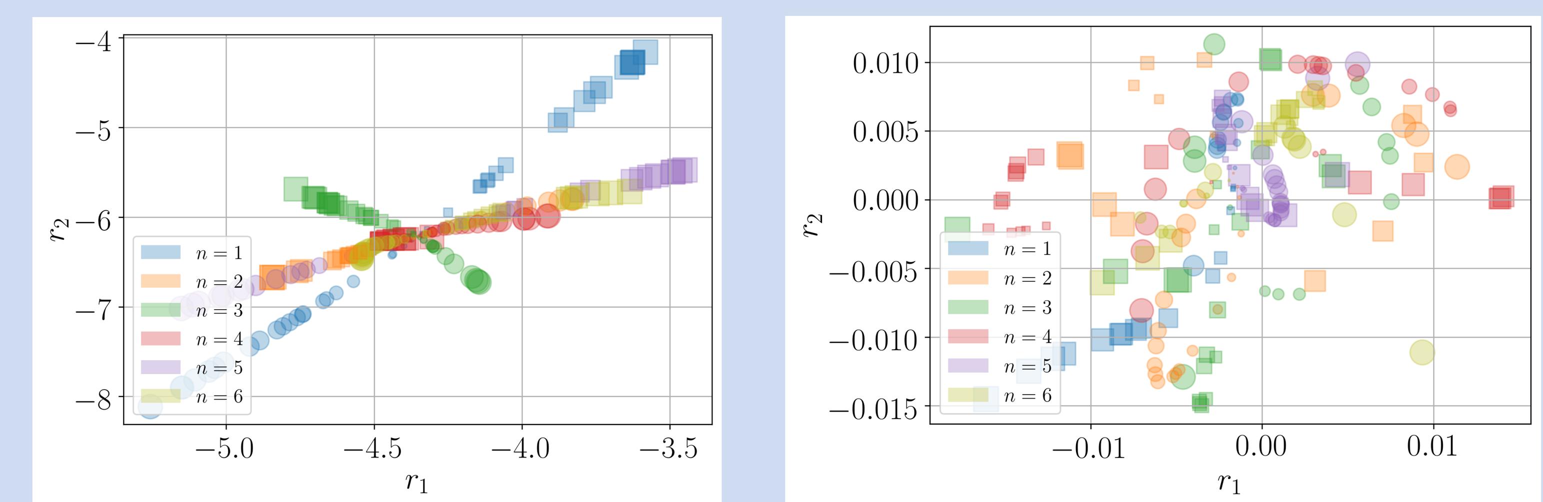

Figure 4

Conclusions

Depending on the value of β, more or less interpretable representations of the model parameters can be obtained. Therefore, using the decoder part of the trained networks as a simulation emulator appears feasible, so long as clear relationships between configuration parameters and latent neurons can be established, as in the left panel of Figure 3.

Acknowledgements

This work was supported by the STFC Hartree Centre's Innovation Return on Research programme, funded by the Department for Business, Energy & Industrial Strategy. The author also acknowledges a UKRI Future Leaders Fellowship for support through the grant MR/T041862/1.

References

Raban Iten et al. (2020), *Discovering Physical Concepts with Neural Networks, Phys. Rev. Lett., 124(1):010508*

S. Brandt, https://bitbucket.org/stevenrbrandt/cajunwave *(Simulation code)*

E. Bentivegna, https://zenodo.org/record/4728023 *(Simulation dataset)*