A million times speed up in parameters retrieval with deep learning

Muhammad Kasim[1], J. Topp-Mugglestone[1], P. Hatfield[1], D. H. Froula[2], G. Gregori[1], M. Jarvis[1], E. Viezzer[3] Sam Vinko[1]
[1]Department of Physics, University of Oxford, [2]Laboratory for Laser Energetics, University of Rochester
[3]Department of Atomic, Molecular and Nuclear Physics, University of Seville
[1]`firstname.lastname@physics.ox.ac.uk`
[2]`dfroula@lle.rochester.edu`, [3]`eviezzer@us.es`

Abstract

Retrieving parameters by matching simulations to experimental or observation data is a common procedure in many areas in physical sciences. However, as the procedure requires multiple trial-and-error simulation runs, it could take hours to weeks to get meaningful results. The slow process of parameters retrieval is hindering large-scale data processing, real-time diagnostics for better experimental control, and sensitivity assessment over large parameters space. Here we show that the process can be accelerated by a factor of up to one million using deep neural networks with optimization algorithms. The method is shown to be robust and quick in retrieving parameters for diagnostics and observations in various research fields: high energy density physics, inertial confinement fusion, magnetic confinement fusion, and astrophysics. The generality of the presented method allows it to be adapted to other parameters retrieval processes in other fields.

1 Introduction

Our ability in modelling processes has allowed us to measure properties that cannot be observed directly. The properties are often retrieved indirectly from some observable signals with the help of a predictive model. Often, only the predictive forward model is available and the inverse model, which relates the observable signals back to the parameters, is unavailable. In this case, the parameters are retrieved by trying multiple sets of the parameters in the forward model until it reaches an agreement with the observed signals.

Parameters retrieval processes has been playing important roles in building our understanding on physical systems. For example, retrieving solar properties from observations by standard solar model [1, 2], obtaining cosmological parameters from the observed cosmic microwave background [3–5], temperature and density measurements of a dense plasma from scattered x-ray spectrum [6–8], and geophysical exploration by reflection seismology [9].

Automated approaches in parameters retrieval are typically done by utilizing optimization algorithms [10] to minimize a loss function between the simulated signals and the actual observed signals. Although the parameters retrieval can be done automatically, the process still relies on evaluating the simulations thousand times which can be very slow.

As the speed of parameters retrieval is limited by the simulations, speeding up the simulations would speed up the parameters retrieval process. One way to speed up a simulation is by emulating it with a deep neural network that takes the same input parameters and produces the same outputs with low latency. Integrating the emulating deep neural networks with an optimization algorithm lets us obtain a set of parameters that best fits observed signals almost instantaneously instead of spending hours or

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

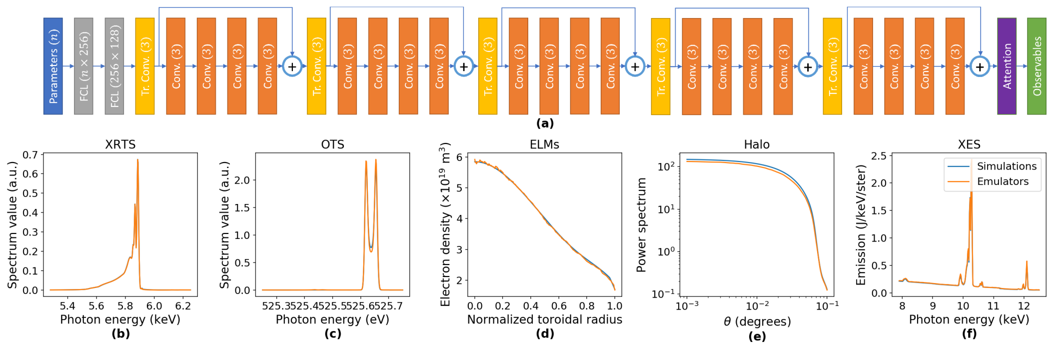

Figure 1: (a) The architecture of the Deep Emulator Network (DEN) which consists of fully connected layers (FCL), transposed convolutional layers, convolutional layers, and an attention layer. (b-f) Comparison results between the simulated output signal produced by the actual simulations (blue lines) from the test dataset and the emulators (orange lines) for (b) x-ray Thomson scattering, (c) optical Thomson scattering, (d) edge localized modes diagnostics, (e) galaxy halo observation, and (f) x-ray emission spectroscopy.

days to get the results. The speed up also enables sensitivity analyses which could discover sensitive and insensitive regions of a diagnostic.

2 Deep Emulator Networks

We trained deep emulator networks (DENs) to emulate 5 diagnostics in plasma physics and astrophysics. They are x-ray Thomson scattering (XRTS) [6–8], x-ray emission spectroscopy (XES) [11, 12], optical Thomson scattering (OTS) [13], edge localized modes (ELMs) diagnostics [14], and galaxy halo observation (Halo) [15]. Each simulation takes an input of a vector with 3-14 elements and produces observable of 1-10 one-dimensional signals. The fastest simulation is Halo which runs in 3 seconds and the slowest ones are XES and ELMs which take about 10-20 minutes.

The architecture of the deep emulator network (DEN) is shown in Figure 1(a). It consists of fully connected layers at the beginning followed by a combination of transposed convolution and convolution layers. It ends with an attention layer to capture the global dependency [16, 17].

The input to the DEN is a vector of parameters and the output is 1-10 channels of one-dimensional signals with 250 points. The DEN is trained to minimize the mean squared error (MSE) between its outputs and the simulation outputs for parameters inside certain bounds. We generated 7,000 random sets of parameters and simulated signals for training, 3,000 for validation, and 4,000 for test.

After the DEN is trained, it can produce the simulation outputs accurately, as shown in Figure 1(b-f). DENs can produce outputs for 256 sets of input parameters in about 5 milliseconds using a Titan X GPU card.

3 Finding the best parameters

With the DENs that could emulate the simulations accurately in much shorter time, we can use them to retrieve parameters that best fit observed signals. Given an observed signal, we run an optimization algorithm to find the parameters that minimizes the error between the signal and the DEN's output.

The optimization algorithm we use for this purpose is the SNES [18] due to its proven successes [19] and simplicity. Although the gradient information can be calculated with DENs, we found that using evolutionary-based algorithms (e.g. SNES and CMA-ES [20]) can obtain the best fit parameters faster and more robust than using gradient-descent algorithms (e.g. L-BFGS [21]).

The parameters retrieval test was done by choosing a set of parameters and a simulated observed signal from the dataset and let the SNES retrieve the parameters using only the observable from the

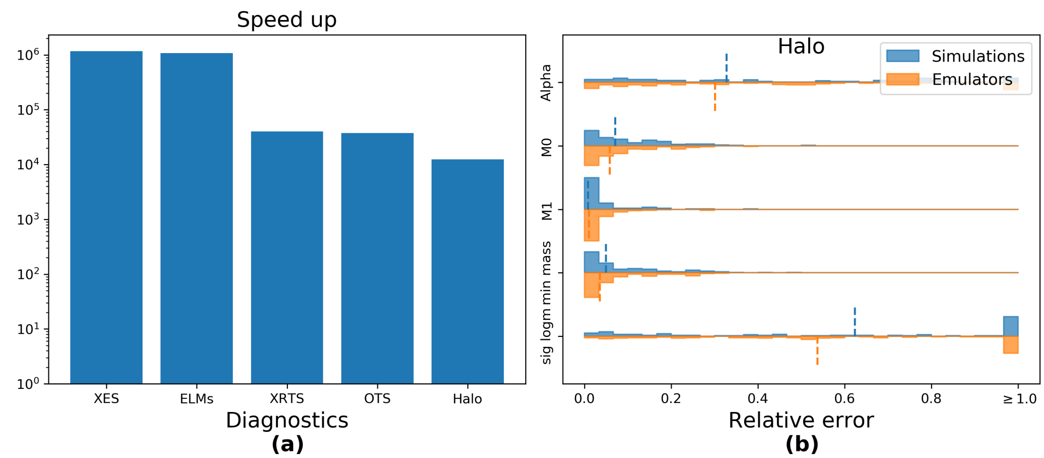

Figure 2: (a) The speed up achieved in obtaining the best parameter using the fast emulator compared to the one using the simulation. (b) Histograms of relative error obtained for the processes done by the simulations (blue) and the emulators (orange) from 100 data points in the dataset for galaxy halo observation. The dashed lines in (b) show the median of the relative error distributions.

dataset. For the comparison, the parameters retrievals were done twice for each data point, one is using SNES with the actual simulation and another one is using SNES with the DEN.

To evaluate the quality of the retrieved parameters, the relative error is calculated. The relative error is defined by the absolute deviation between the retrieved parameter and the actual parameter divided by the actual parameter. The relative errors of the parameters retrieved using the DENs and the actual simulations are compared in Figure 2(b) for galaxy halo observation.

From the Figure, we can see that the relative errors of the parameters retrieved with DENs is comparable to the parameters retrieved with actual simulations. The medians of the relative error distributions obtained by the DENs are close to the median values obtained by the actual simulations, except for the parameters to which the diagnostics are insensitive (e.g. *alpha* and *sig_logm*).

With the similar errors of the retrieved parameters, the parameters retrieval using DENs can be done in 80-300 milliseconds using a Titan X GPU card. This is much quicker than parameters retrieval using the actual simulations which could take hours to days with 8 CPU cores. The average speed up factors of parameters retrieval using the DENs with a GPU card compared to the actual simulations with 8 CPU cores are shown in Figure 2(b). We can see that the parameters retrieval using the DENs can be a million times faster than the parameters retrieval using the actual simulations without compromising much of the retrieval quality.

4 Discovering insensitive regions

In Figure 2(b), there are some parameters that have large relative errors even though they are retrieved using the actual simulations. This is due to the presence of multiple regions in the parameters space that maps into a very similar observable signal. This effect can be seen on Figure 3 where we compare the observable from the dataset and the observable signals at the best fit parameters obtained by the emulators. In those cases, we can see that even though the relative error of the parameters can be up to 100% or more, they can produce very similar observable signals.

Given an observable signal from one point in the parameters space, one can perform Bayesian posterior sampling with an MCMC algorithm to discover the parameters span that produces similar observable signals. If the parameters span is large, then we know that the diagnostic is insensitive at that point in the parameters space. Finding insensitive regions of a diagnostic in the parameters space requires performing Bayesian posterior sampling for many points in the parameters space. Performing this with an actual simulation could take up to a year even if the simulation runs in a few seconds. With the fast emulator, the process can be done in a few hours.

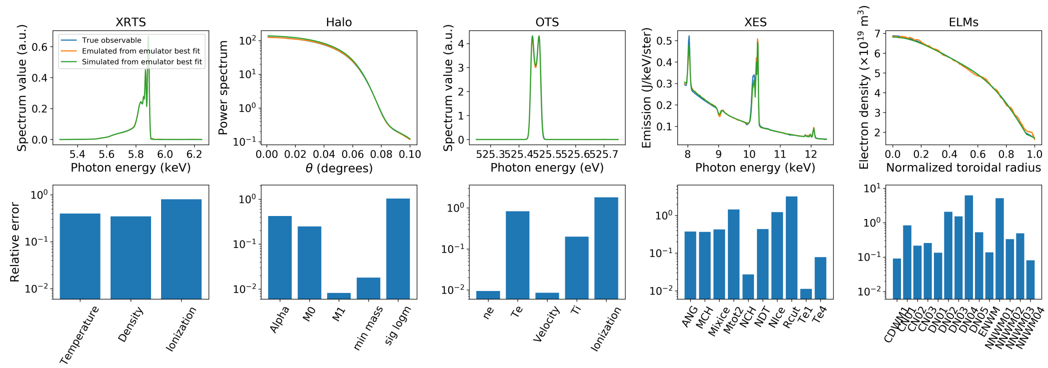

Figure 3: (Top rows) The comparison between the true observable where the parameters are to be retrieved from (blue) and the simulated observable signals produced by the fast emulator (orange) and the actual simulation (green) using the best fit parameters found by the fast emulator. (Bottom rows) The relative error of the best fit parameters retrieved by the fast emulator and the actual known parameters.

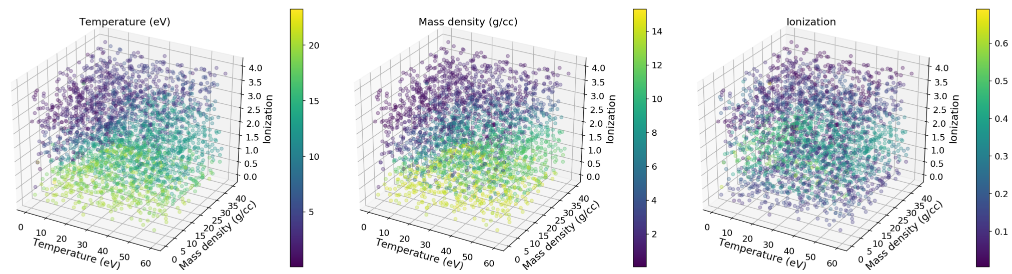

Figure 4: Global sensitivity analysis results for XRTS on specified bounds. The color of each point in the boxes represents the 1σ uncertainty (i.e. half of the 68% confidence interval's width) of each parameter distribution that produces spectra within the 3.5% bounds.

We searched for the sensitive and insensitive regions for the XRTS using the trained DEN by evaluating the sensitivity at 3,000 points in the parameters space. For each evaluated point, we run the ensemble MCMC algorithm [22] to get the parameter ranges that produce observable signals that lie within the 3.5% bound of the observable signal of the evaluated point. The parameter ranges that fit the criteria above at each point are shown in Figure 4.

From the figure, we can see the regions where the parameters have sensitive region and insensitive region. For example, the temperature and density measurements have relatively small uncertainties and high sensitivities when the ionization is high. We can also see that the ionization measurement have a sensitive region where the ionization is low (i.e. less than about 1) and where it is high (i.e. greater than about 3). This kind of analysis results would be very expensive to obtain with the actual simulations, even if the simulation runs in a few seconds.

5 Conclusions

We have presented a technique combining deep neural networks with optimization to speed up the process in retrieving parameters from observations using simulations. The parameters retrieval can be accelerated up to a million times faster without losing much of the accuracy. The accelerated parameters retrieval process opens up a possibility of performing various sensitivity analysis to discover sensitive and insensitive regions that would be prohibitively expensive without the emulator. We have shown that the method can be applied to diagnostics in various research fields in plasma physics and astrophysics. The presented method can be easily adapted to other diagnostics.

References

[1] John N Bahcall, Aldo M Serenelli, and Sarbani Basu. “10,000 standard solar models: a Monte Carlo simulation”. In: *The Astrophysical Journal Supplement Series* 165.1 (2006), p. 400.

[2] John N Bahcall et al. “Are standard solar models reliable?” In: *Physical Review Letters* 78.2 (1997), p. 171.

[3] Peter AR Ade et al. “Planck 2013 results. XVI. Cosmological parameters”. In: *Astronomy & Astrophysics* 571 (2014), A16.

[4] Peter AR Ade et al. “Planck 2015 results-xiii. cosmological parameters”. In: *Astronomy & Astrophysics* 594 (2016), A13.

[5] N Aghanim et al. “Planck 2018 results. VI. Cosmological parameters”. In: *arXiv preprint arXiv:1807.06209* (2018).

[6] HJ Lee et al. “X-ray Thomson-scattering measurements of density and temperature in shock-compressed beryllium”. In: *Physical review letters* 102.11 (2009), p. 115001.

[7] Andrea L Kritcher et al. “Ultrafast X-ray Thomson scattering of shock-compressed matter”. In: *Science* 322.5898 (2008), pp. 69–71.

[8] JC Valenzuela et al. “Measurement of temperature and density using non-collective X-ray Thomson scattering in pulsed power produced warm dense plasmas”. In: *Scientific reports* 8.1 (2018), p. 8432.

[9] J Korenaga and WW Sager. “Seismic tomography of Shatsky Rise by adaptive importance sampling”. In: *Journal of Geophysical Research: Solid Earth* 117.B8 (2012).

[10] MF Kasim et al. “Inverse Problem Instabilities in Large-Scale Plasma Modelling”. In: *arXiv preprint arXiv:1805.08301* (2018).

[11] SP Regan et al. “Hot-spot mix in ignition-scale inertial confinement fusion targets”. In: *Physical review letters* 111.4 (2013), p. 045001.

[12] Orlando Ciricosta et al. “Simultaneous diagnosis of radial profiles and mix in NIF ignition-scale implosions via X-ray spectroscopy”. In: *Physics of Plasmas* 24.11 (2017), p. 112703.

[13] P Tzeferacos et al. “Laboratory evidence of dynamo amplification of magnetic fields in a turbulent plasma”. In: *Nature communications* 9.1 (2018), p. 591.

[14] J Galdon-Quiroga et al. “Beam-ion acceleration during edge localized modes in the ASDEX Upgrade tokamak”. In: *Physical review letters* 121.2 (2018), p. 025002.

[15] PW Hatfield et al. “The galaxy–halo connection in the VIDEO survey at 0.5< z< 1.7”. In: *Monthly Notices of the Royal Astronomical Society* 459.3 (2016), pp. 2618–2631.

[16] Ashish Vaswani et al. “Attention is all you need”. In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[17] Xiaolong Wang et al. “Non-local neural networks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 7794–7803.

[18] Daan Wierstra et al. “Natural evolution strategies”. In: *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*. IEEE. 2008, pp. 3381–3387.

[19] Tim Salimans et al. “Evolution strategies as a scalable alternative to reinforcement learning”. In: *arXiv preprint arXiv:1703.03864* (2017).

[20] Nikolaus Hansen. “The CMA evolution strategy: A tutorial”. In: *arXiv preprint arXiv:1604.00772* (2016).

[21] Dong C Liu and Jorge Nocedal. “On the limited memory BFGS method for large scale optimization”. In: *Mathematical programming* 45.1-3 (1989), pp. 503–528.

[22] Jonathan Goodman and Jonathan Weare. “Ensemble samplers with affine invariance”. In: *Communications in applied mathematics and computational science* 5.1 (2010), pp. 65–80.