Accelerating Dynamic Aperture Evaluation Using Deep Neural Networks

D. Di Croce[2], M. Giovannozzi[1], T. Pieloni[2], M. Seidel[2,3], F. F. Van der Veken[1]

[1]Beams Department - CERN, Geneva, Switzerland
[2]École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
[3]Paul Scherrer Institut, Villigen, Switzerland

E-mail: `davide.dicroce@epfl.ch`

Abstract. The Dynamic Aperture is an important concept for the study of non-linear beam dynamics in a circular accelerator. The DA is defined as the extent of the phase-space region in which the particle's motion remains bounded over a finite number of turns. Such a region is shaped by the imperfections in the magnetic fields, beam-beam effects, electron lens, electron clouds, and other non-linear effects. The study of the DA provides insight into the mechanisms driving the time evolution of beam losses, which is essential for the operation of existing circular accelerators, such as the CERN Large Hadron Collider, as well as for the design of future ones. The standard approach to numerical evaluation of the DA relies on the ability to accurately track initial conditions, distributed in phase space, on a realistic time scale, and this is computationally demanding. To accelerate the angular DA calculation, we propose the use of a Machine Learning technique for the angular DA regression based on simulated HL-LHC data. We demonstrate the implementation of a Deep Neural Network model by measuring the time and assessing the performance of the angular DA regressor, as well as carrying out studies with various hardware architectures including CPU, GPU, and TPU.

1. Introduction

The study of dynamic aperture (DA), defined as the extent of the connected phase-space region in which the single-particle dynamic is bounded, provides insight into the single-particle, non-linear beam dynamics and mechanisms driving the time evolution of beam losses (see, e.g. [1]), which is essential for the design and operation of existing (see, e.g. [2, 3]) and future circular accelerators (see, e.g. [4]).

The numerical calculation of the DA involves tracking a large number of initial conditions in phase space for many turns (see, e.g. [5, 6]). This method is computationally demanding, especially for large accelerators such as the CERN Large Hadron Collider (LHC) [2], and for this analytical scaling laws have been studied for several years [7, 6]. In general, in the accelerator community, there is growing interest in developing methods to accelerate the DA calculation while maintaining its accuracy.

In recent years, Machine Learning (ML) techniques have emerged as a promising approach to accelerate DA evaluation (see, e.g. [8, 9, 10, 11]). By training a model on a large data set of simulated initial conditions, an ML algorithm can learn the complex mapping between the initial conditions and the angular DA (see below) and provide a fast and accurate prediction of the angular DA for new sets of initial conditions and machine configurations. This approach has the potential to reduce the computational cost of DA evaluation and enable faster accelerator parameter optimisation.

Here, we propose to use machine learning techniques to speed up angular DA evaluation based on simulated data obtained using the High Luminosity LHC (HL-LHC) lattice [3]. We investigated the use of a Deep Neural Network (DNN) model to regress the angular DA as a function of the initial conditions. We study the performance of this ML model on various hardware architectures and compare it with the standard simulation method.

2. Simulated Samples

To train the ML model, we simulated several accelerator configurations using MAD-X [12] and the V1.0 HL-LHC lattice in the injection configuration at 450 GeV [13]. We varied six accelerator parameters, namely the betatron tunes Q_x, Q_y, chromaticities Q'_x, Q'_y, strength of the Landau octupoles (using the current, I_{MO}, powering the octupoles) and the realisations (also called seeds) of the magnetic field errors assigned to the various magnet families. Furthermore, both Beam 1 and Beam 2 have been considered in these studies. For this first study, we limited the parameters sampling to two Q_x, Q_y scans (8 Q_x values in $[0.255, 0.295]$ and 9 Q_y values in $[0.280, 0.325]$) and a Q', I_{MO} scan (15 Q' values in $[0, 15]$ and 17 I_{MO} values in $[-40, 40]$A) for Beam 1 and Beam 2 and the possible realisations of the magnetic errors. This resulted in a total of 29880 sets of accelerator parameters.

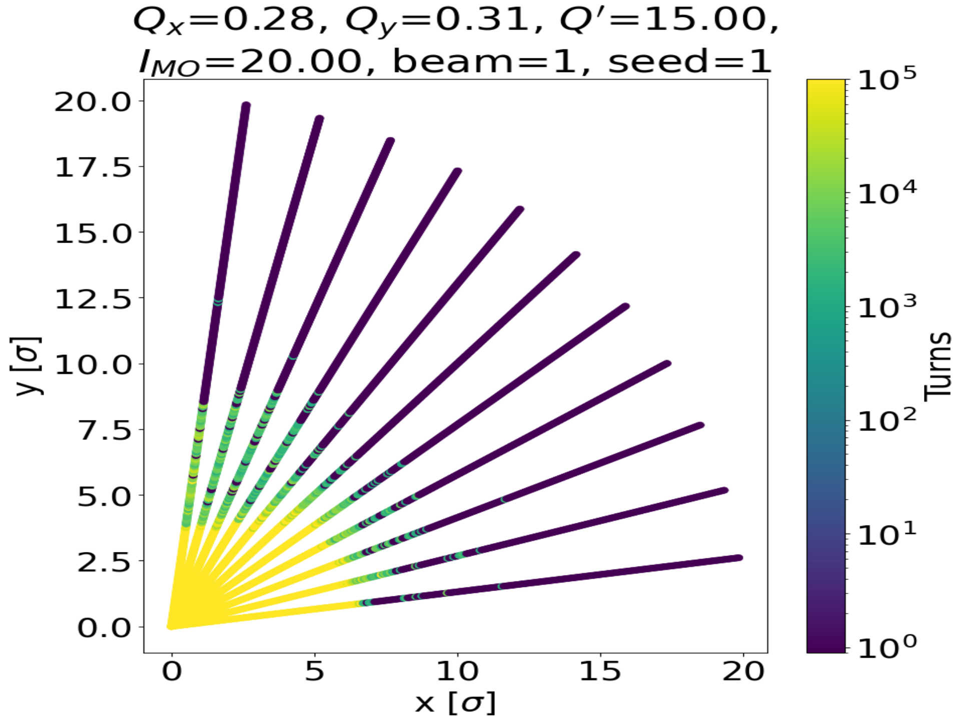

Figure 1. Stability distribution given in units of beam σ for a specific accelerator configuration.

The phase space was probed by tracking with SixTrack [14] for 10^5 turns a set of initial conditions selected along 11 polar angles, evenly distributed in $]0, \pi/2[$ and 290 radial amplitudes, evenly distributed in $[0.0\sigma, 20\sigma]$. An example of the results of these computations in the $x - y$ space is shown in Fig. 1 for a specific accelerator configuration, in which the stability time, i.e.

the time taken by the orbit to reach an amplitude corresponding to a numerical overflow, is provided for each initial condition.

The input for the surrogate model is given by the parameters describing the accelerator configuration and the polar angle, the regressor will learn for each accelerator configuration the value of the last stable amplitude for that angle, which we call angular DA. When considering the angle as an additional parameter, the number of samples is increased to 328680. From this number, 10% of the samples were used for validation during training, and 10% was used only to test the performance of the model, not for training.

To prevent extreme angular DA values from affecting the regressor [15], we cap values above 10 σ, as they are outliers in a distribution ranging from 0 σ to 20 σ. Additionally, to make the training data set more representative of any angular DA value, a weighting scheme based on the inverse angular DA distribution was used. This ensured that the less common angular DA values had higher weights, resulting in a more accurate regression model throughout the range of angular DA values.

3. Network Architecture and Training

As our data set is limited to about 328k samples, which is considered a small size to train a deep learning model [16], we decided to test a simple Deep Feed Forward Neural Network and train with the NADAM optimiser [17]. The network was developed using the TensorFlow library [18]. Architecture and hyperparameters were optimised by random search with the Keras Tuner framework [19]. The best model consists of four hidden layers with 1024, 512, 256 and 32 nodes, respectively. Batch normalisation and dropout (5%) were added between hidden layers to improve performance and avoid overfitting. The loss used for the regressor is Mean Absolute Error (MAE) function. The initial learning rate is 5×10^{-4} and is halved every 5 sequential epochs if the validation loss is not improved. We found training for 80 epochs to be sufficient for convergence.

4. Regressor for the Angular Dynamic Aperture

4.1. Results

The MAE of the angular DA regressor is 0.64(0.65) beam σ for the test (train) data set. This suggests that the regressor is making relatively accurate predictions, with errors that are generally small compared to the typical range of angular DA values. Although the Mean Absolute Percentage Error (MAPE) is 8.1(8.2)% for the test (train) data set and the Absolute Percentage Error (APE) distribution indicates that the angular DA regressor has the same accuracy throughout the range of angular DA values, rather than being biased towards specific values.

In addition, analysis of the angular DA scatter histogram in Fig. 2 reveals that the model performs well for most of the data points, with a tight cluster around the diagonal line, indicating accurate predictions. Furthermore, only 5.62% of the test data set has an APE greater than 10%. These higher errors are likely due to the limited size of the train data set and the number of accelerator parameters, causing a non-linear relationship between the input parameters and the angular DA values. To better understand and address these predictions, we plan to investigate the impact of new input parameters on the accuracy of the model, and to explore potential ways to improve the model's performance.

4.2. Timing Performance

Various factors, such as the number of samples and accelerator parameters, can significantly impact the training and evaluation time of a model, leading to an increase in computational costs and memory requirements.

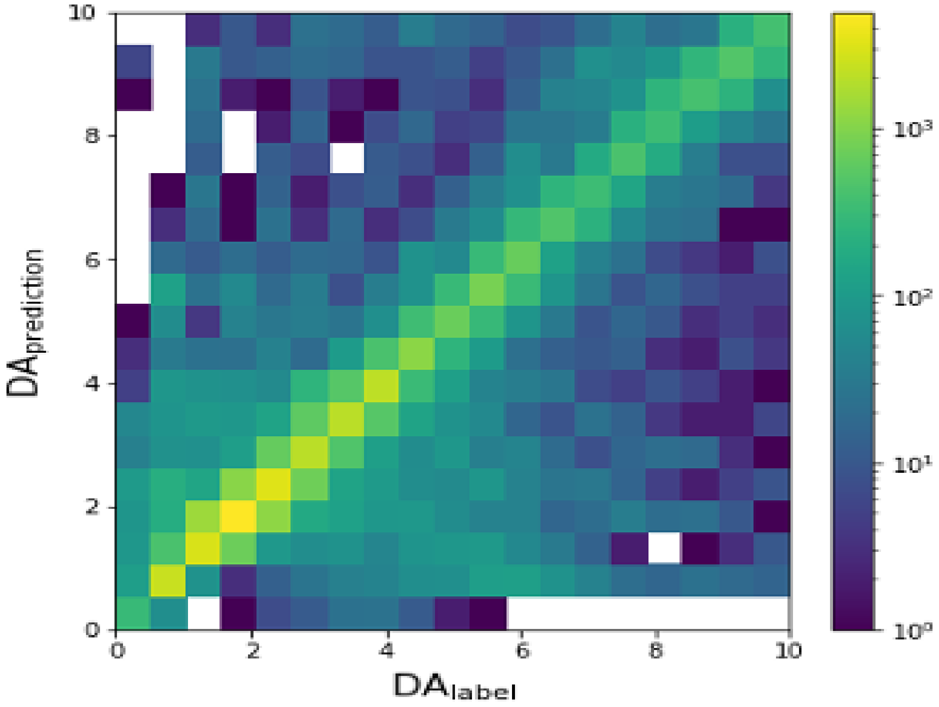

Figure 2. Angular DA predicted as a function of the expected angular DA values for the test data set.

To address this challenge, we compared timing performance using mixed precision operations and a fixed batch size of 128 samples for four hardware architectures with our angular DA regressor: an Apple M1 Pro [20], a NVIDIA Titan V [21], a NVIDIA A100 Tensor Core [22], and Google TPU v2-8 [23].

We tested the Apple M1 Pro on a MacBook Pro laptop. The Titan V was accessed through CERN workstations, utilising 48 AMD Ryzen Threadripper 2970WX CPU cores for data loading and pre-processing tasks. The A100 was accessed via the Google Colab platform with 12 available Intel Xeon 2.20 GHz CPU cores. Google TPU v2-8, a cloud-based accelerator optimised for deep learning workloads, was accessed through the Google Colab platform.

The Titan V outperformed other devices in terms of I/O time (0.012 ms/batch) due to the higher number of available cores for data loading and pre-processing tasks. On the other hand, the Google TPU had the highest I/O time (0.092 ms/batch) due to the need to transfer data to and from the cloud, which introduced network latency. However, the TPU outperformed other devices in terms of training (4 ms/batch) and validation time (3 ms/batch), indicating that it is optimised for deep learning workloads and has a low latency I/O pipeline that does not compete for CPU resources.

Both Titan V and A100 had similar performance in training (7 ms/batch) and validation (2 ms/batch), outperforming M1 Pro (49 ms/batch for training and 9 ms/batch for validation) in terms of training and validation time, indicating that they are better suited for large-scale ML workloads.

Despite the relatively small data set used in this study (20 MB), future studies with larger data sets and more accelerator parameters should consider I/O time as bottlenecks can occur and affect training and validation times.

5. Discussion

Examples of angular DA reconstruction carried out by our regressor for four accelerator configurations (available only in the test data set) are shown in Fig. 3.

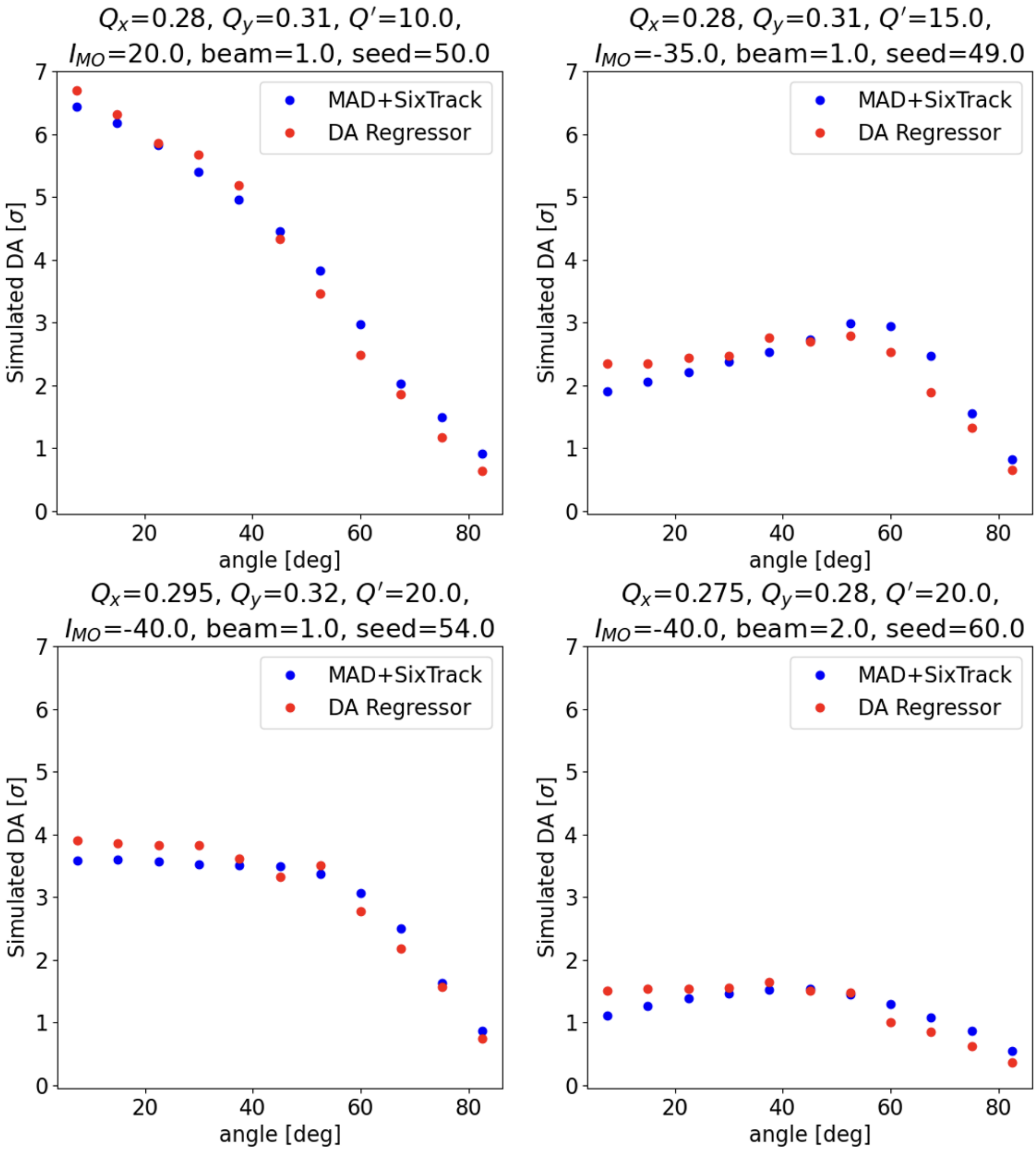

Figure 3. Angular DA regressor in the test data set.

As mentioned above, the data set used in this study is relatively small, with only 328680 instances. This may have limited the complexity of the surrogate model, as deep learning models typically require large amounts of data to achieve their full potential [24]. Therefore, it is likely that the predictive accuracy of the regressor will improve with larger data sets. Furthermore, the model accuracy could improve by including additional dynamical variables, such as the phase advance between Interaction Points, the amplitude detuning, the linear coupling, and others.

Note also that the average inference time of the angular DA regressor for a batch size of 128 samples is approximately 1.4 s using GPUs (approximately 0.12 s/machine configuration), which is significantly faster than the average 5 day run time of a full simulation submission of 4800 HL-LHC accelerator configurations using the standard approach based on MAD-X and SixTrack combined with the BOINC system [25] to submit multiple tracking jobs in parallel (approximately 90 s/machine configuration). Therefore, the angular DA regressor is about 750 times faster than the standard approach in terms of computing time per machine configuration. Therefore, the angular DA regressor is about 750 times faster than the standard approach in terms of computing time per machine configuration.

As a MAPE of 8.1% could be considered high in the context of deep learning [26], it may not completely replace traditional simulations for predicting angular DA values due to its sub-optimal accuracy. As the physics of unstable chaotic motion is only accessible via tracking, traditional simulations are still essential in determining whether initial conditions are chaotic. For this reason, one could use the regressor in conjunction with tracking to reduce the number of simulations required and ensure reliable results.

6. Conclusions

Our study explored the potential of ML techniques, specifically a DNN model, for predicting the angular DA of circular accelerators. Although the results showed promising performance of the model, with relatively low errors in predicting angular DA values, more research is necessary to confirm its effectiveness in a larger and more diverse data set. Furthermore, our results indicate that deep learning can reduce the computational cost of angular DA evaluation. This approach could enable faster machine parameter optimisation and, in the future, perhaps even real-time monitoring and control of beam dynamics in circular accelerators. However, tracking is still essential to determine whether the dynamics of certain initial conditions is chaotic.

Acknowledgments

This work was carried out under the auspices and with the support of the Swiss Accelerator Research and Technology programme [27] and the Swiss Data Science Centre (SDSC).

References

[1] Giovannozzi M 2012 *Phys. Rev. ST Accel. Beams* **15** 024001

[2] Brüning O S, Collier P, Lebrun P, Myers S, Ostojic R, Poole J and Proudlock P 2004 *LHC Design Report* CERN Yellow Rep. Monogr. (Geneva: CERN)

[3] Apollinari G, Béjar Alonso I, Brüning O, Fessia P, Lamont M, Rossi L and Tavian L 2017 *High-Luminosity Large Hadron Collider (HL–LHC)* (*CERN Yellow Rep. Monogr.* vol 4) (Geneva: CERN)

[4] Benedikt M *et al.* 2019 *The European Physical Journal Special Topics* **228** 755–1107

[5] Todesco E and Giovannozzi M 1996 *Phys. Rev. E* **53**(4) 4067–4076 URL `https://link.aps.org/doi/10.1103/PhysRevE.53.4067`

[6] Bazzani A, Giovannozzi M, Maclean E, Montanari C E, Van der Veken F F and Van Goethem W 2019 *Phys. Rev. Accel. Beams* **22**(10) 104003 (*Preprint* `1909.09516`) URL `https://link.aps.org/doi/10.1103/PhysRevAccelBeams.22.104003`

[7] Giovannozzi M, Scandale W and Todesco E 1998 *Phys. Rev. E* **57**(3) 3432–3443 URL `https://link.aps.org/doi/10.1103/PhysRevE.57.3432`

[8] Giovannozzi M, Maclean E, Montanari C E, Valentino G and Van der Veken F F 2021 *Information* **12** ISSN 2078-2489 URL `https://www.mdpi.com/2078-2489/12/2/53`

[9] Van der Veken F F, Giovannozzi M, Maclean E H, Montanari C E and Valentino G 2021 Using Machine Learning to Improve Dynamic Aperture Estimates *Proc. IPAC'21* (JACoW Publishing, Geneva, Switzerland) pp 134–137 URL `https://jacow.org/ipac2021/papers/MOPAB028.pdf`

[10] Schenk M, Coyle L, Giovannozzi M, Krymova E, Mereghetti A, Obozinski G and Pieloni T 2021 Modeling Particle Stability Plots for Accelerator Optimization Using Adaptive Sampling *Proc. IPAC'21* (JACoW Publishing, Geneva, Switzerland) pp 1923–1926 URL `https://jacow.org/ipac2021/papers/TUPAB216.pdf`

[11] Van der Veken F F, Akbari R, Bogaert M, Fol E, Giovannozzi M, Lowyck A, Montanari C E and Van Goethem W 2022 Determination of the Phase-Space Stability Border with Machine Learning Techniques *Proc. IPAC'22* (*International Particle Accelerator Conference* no 13) (JACoW Publishing, Geneva, Switzerland) pp 183–186 ISBN 978-3-95-450227-1 ISSN 2673-5490

[12] MAD - Methodical Accelerator Design (`https://mad.web.cern.ch/mad/`)

[13] Maria R D, Fartoukh S D, Bogomyagkov A V and Korostelev M 2013 HLLHCV1.0: HL-LHC Layout and Optics Models for 150 mm Nb3Sn Triplets and Local Crab-cavities *Proc. IPAC'13* (JACoW Publishing, Geneva, Switzerland) pp 1358–1360 URL `https://jacow.org/IPAC2013/papers/TUPFI014.pdf`

[14] De Maria R *et al.* SixTrack – 6D Tracking Code (`http://sixtrack.web.cern.ch/SixTrack/`)

[15] Cook R D and Weisberg S 1982 *Journal of the American Statistical Association* **74** 169–174

[16] Hubara I, Soudry D and Meir R 2016 Do deep nets really need to be deep? *Advances in Neural Information Processing Systems* vol 29 pp 265–273

[17] Timothy D 2016 Incorporating Nesterov momentum into Adam *International Conference on Learning Representations*

[18] Abadi M *et al.* 2015 Tensorflow: Large-scale machine learning on heterogeneous systems URL `https://www.tensorflow.org/`

[19] Gomes A O *et al.* Keras tuner GitHub repository URL `https://github.com/keras-team/keras-tuner`

[20] Apple Inc 2021 Apple M1 Pro chip URL `https://www.apple.com/mac/m1/`

[21] NVIDIA Corporation 2017 NVIDIA TITAN V URL `https://www.nvidia.com/en-us/titan/titan-v/`

[22] NVIDIA Corporation 2020 NVIDIA A100 URL `https://www.nvidia.com/en-us/data-center/a100/`

[23] Google LLC 2022 Cloud TPU URL `https://cloud.google.com/tpu`

[24] LeCun Y, Bengio Y and Hinton G 2020 *Nature* **521** 436–444

[25] Barranco J *et al.* 2017 *Open Eng.* **7** 378–392

[26] Venkatesan K and Ramachandran R 2018 *International Journal of Advanced Computer Science and Applications* **9** 418–425

[27] Accelerator Research and Technology programme - CHART (`http://www.chart.ch`)