12th Int. Particle Acc. Conf. IPAC2021, Campinas, SP, Brazil JACoW Publishing
ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-TUPAB216

MODELING PARTICLE STABILITY PLOTS FOR ACCELERATOR OPTIMIZATION USING ADAPTIVE SAMPLING

M. Schenk*[1], L. Coyle[1], M. Giovannozzi, A. Mereghetti[2], T. Pieloni[1]
CERN Beams Department, Geneva, Switzerland
E. Krymova, G. Obozinski
Swiss Data Science Center, ETH Zürich and EPFL, Lausanne, Switzerland
[1]also at École Polytechnique Fédérale de Lausanne, Switzerland
[2]presently at Centro Nazionale di Adroterapia Oncologica, Pavia, Italy

Abstract

One key aspect of accelerator optimization is to maximize the dynamic aperture (DA) of a ring. Given the number of adjustable parameters and the compute-intensity of DA simulations, this task can benefit significantly from efficient search algorithms of the available parameter space. We propose to gradually train and improve a surrogate model of the DA from SixTrack simulations while exploring the parameter space with adaptive sampling methods. Here we report on a first model of the particle stability plots using convolutional generative adversarial networks (GAN) trained on a subset of SixTrack numerical simulations for different ring configurations of the Large Hadron Collider at CERN.

INTRODUCTION

The performance of a storage ring is characterized, among others, by the extent of the volume in phase space where the charged particles feature bounded, i.e. stable, dynamics as simulated by long-term tracking. This is best described by the stability plot shown in Fig. 1, obtained with single-particle tracking codes such as SixTrack [1]. The plot illustrates how many revolutions particles starting from different initial horizontal and vertical amplitudes perform before they are lost from the beam. Every dot represents one particle at its initial horizontal and vertical amplitudes. Typically, particles at small amplitudes are stable for millions of revolutions, while those with higher initial amplitudes may be lost on much shorter timescales as they are more susceptible to the field errors of the imperfect magnetic multipoles and to the effects of the non-linear magnets (sextupoles, octupoles, etc.) required to suppress collective beam phenomena through chromaticity and Landau damping, for example [2, 3]. The boundary between the stable and the chaotic regimes is called Dynamic Aperture (DA) (see, e.g. [4, 5]). To maximize the DA, one has to properly adjust a large number of machine parameters, which requires efficient search algorithms in the available parameter space. The search for optimal parameter settings has been mostly a manual task where beam physicists scan promising areas of the entire parameter space with particle tracking simulations. This is usually complemented by employing analytical concepts like resonance driving terms (RDTs) [6]. Typically, lower priority parameters, although still relevant for machine performance, have often not been optimized for due to the lack of better search algorithms. For a machine as complex as the Future Circular Collider (FCC) [7], an extensive search of the available parameter space with a manual approach is rather impractical and would likely not reveal the best working point.

Surrogate modeling techniques combined with adaptive sampling have been used in other engineering domains where function evaluations, such as running compute-intensive simulations or experimental data acquisitions, are expensive (see e.g. [8, 9] and references therein). The process of finding optimal settings for a complex machine like the FCC requires among others thousands of computationally expensive tracking simulations and will hence benefit from combining such techniques with existing beam physics codes like SixTrack.

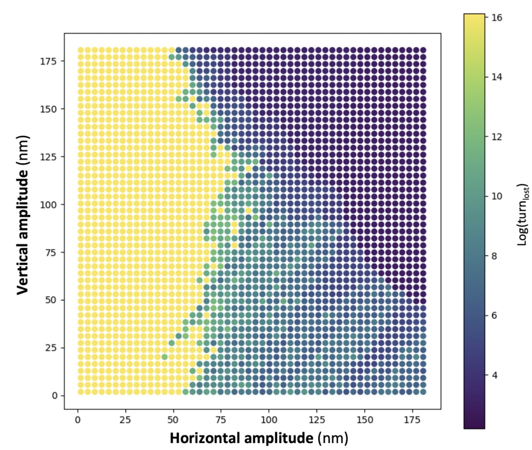

Figure 1: Example stability plot obtained with SixTrack.

The main objectives of this work are twofold: first, to identify and develop a suitable model for the stability plots as a function of the machine parameters, and second, to develop an adaptive sampling framework that provides a targeted, efficient, yet exhaustive search for optimal parameter settings. Initially, the model is trained on a readily available set of SixTrack simulation data. From this point onwards, it will be updated iteratively with new incoming simulation data obtained by actively and automatically exploring the available parameter space. The model should ideally provide confidence intervals on its predictions in order for the adaptive sampler to automatically pick the next sampling points that best improve the accuracy of the current model.

* michael.schenk@cern.ch

ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-TUPAB216

Here we report on first attempts to model the particle stability plots. After explaining the modeling technique and the data used, some first, preliminary results are discussed.

MODEL AND DATA

Generative Adversarial Network

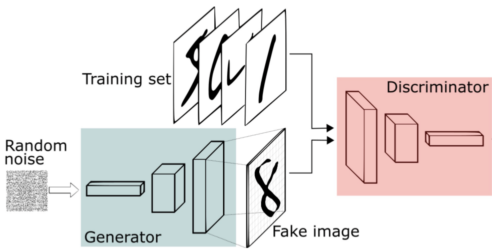

Figure 2: Schematic of a classical GAN [10].

Figure 2 shows a schematic of a generative adversarial network (GAN). It is composed of two neural networks, a *generator* and a *discriminator*, that constantly try to outperform each other as training progresses [11]. These two networks can be perceived as *actor* and *critic*, respectively. While the generator tries to produce fake images that look identical to those in the training set, the discriminator tries to distinguish real training images from fake ones. The two networks are trained simultaneously in the minimax game: as training progresses, the generator gets better at creating counterfeit images by minimizing the probability that the discriminator is correct. At the same time the discriminator gets better at distinguishing fake from real data. Eventually, an equilibrium is reached as we will see below. At this stage the GAN is considered trained.

Unfortunately, once trained, the generator of a classical GAN just outputs *random* images from the model distribution learnt from the training set without us having any control. For our purpose, however, we would like to force the generator to produce the particle stability plot of the specific parameter settings that we request. This can be achieved by employing a conditional GAN (c-GAN) [12]. C-GANs are trained much like classical GANs with the important difference that during training we also provide a label that characterizes the class that the image belongs to. In our case, the class label is given by the parameter settings associated with the stability plot.

The stability plots can be perceived as rasterized images where each initial condition represents one pixel. As a result, the c-GAN architecture employed here was chosen to be composed of 4 convolutional and 2 dense layers for both the generator and the discriminator networks. While the generator uses filter and up-scaling layers to get from 6×6 random noise input to 12×12, and eventually 24×24 pixel images, the discriminator uses filter and down-scaling convolutional layers to get from 24×24 to 12×12, and to 6×6 pixel images.

Training Data

For this initial study, the training data was produced using SixTrack, running a scan in both, first-order chromaticity Q' and Landau octupole current I_{oct} for the CERN Large Hadron Collider (LHC) lattice at injection energy 450 GeV. The stability plots for 1728 different (Q', I_{oct}) pairs were generated. The number of samples in the initial amplitudes in both transverse planes was set to 24, corresponding to stability plots with a relatively low resolution of 24×24 pixels. This will be increased in the future.

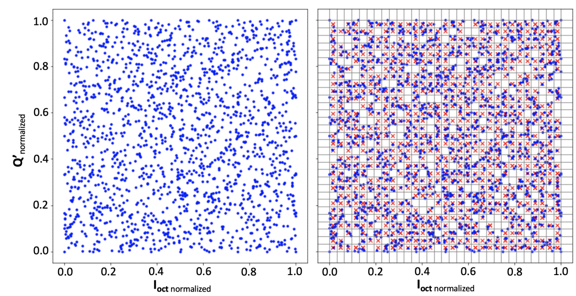

Figure 3: Distribution of training data (left) and discretization onto a regular grid (right) in parameter space.

The distribution of training examples in the 2D parameter space is shown in Fig. 3, left. The parameters have been mapped to the range $[0, 1]$, corresponding to $Q' \in [5, 25]$ and $I_{\mathrm{oct}} \in [-50, 50]$ A. To train the c-GAN one could in principle provide directly a tuple of the continuous parameter values (Q', I_{oct}) as input label to every image. However, to improve the stability of the c-GAN training, we create discrete labels instead as illustrated in Fig. 3, right. The chosen discretization is 31×31. It is likely and desirable that this restriction of discretizing the input parameters can be lifted in the future. One notices that in the discretized parameter space some settings of (Q', I_{oct}) do not have any corresponding training data. Assuming that the model generalizes well, it will interpolate for the labels with missing training data and produce reasonable results nevertheless. This will be demonstrated below.

RESULTS

Figure 4 shows the training evolution of the discriminator for real (blue) and fake (orange) input images, as well as for the generator (green). After training the GAN for 100 epochs corresponding to just over 5000 training steps, the GAN loss functions stabilize and reach an equilibrium [11]. The final loss values of the neural networks are consistent with expectations for this type of GAN.

Comparisons between SixTrack output and model predictions are shown in Fig. 5 for two different settings of (Q', I_{oct})[1]. The particle actions $(2J_x, 2J_y)$ are given in units of rms geometric emittance which was set to 7.3 nm in both transverse planes, corresponding to a normalized rms emittance of $\varepsilon_{\mathrm{rms}} = 3.5\,\mu$m. The particle loss vs turns is also

[1] More examples for different parameter settings are available here: https://cernbox.cern.ch/index.php/s/RQncUlluhuJARDx

ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-TUPAB216

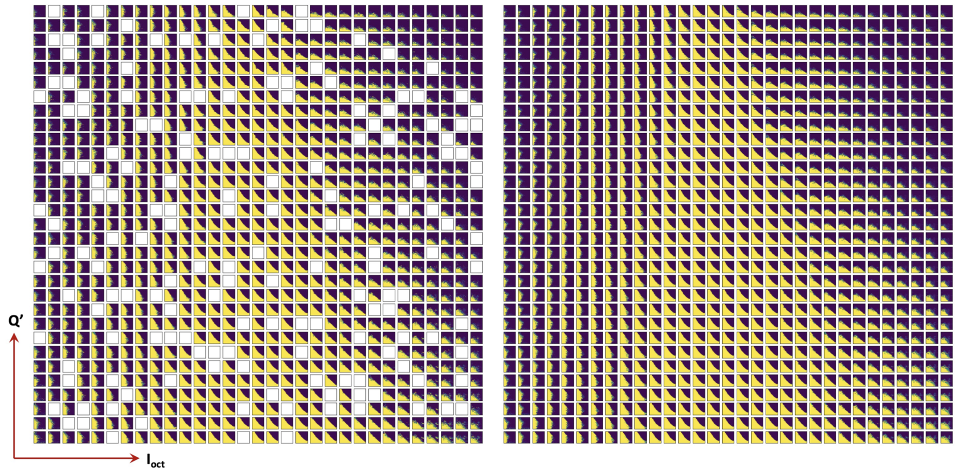

Figure 6: SixTrack output (left) and model predictions (right) over the entire Q' vs I_{oct} parameter space.

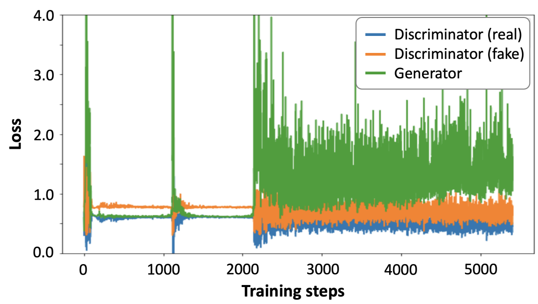

Figure 4: GAN training evolution of the loss functions of discriminator and generator, respectively.

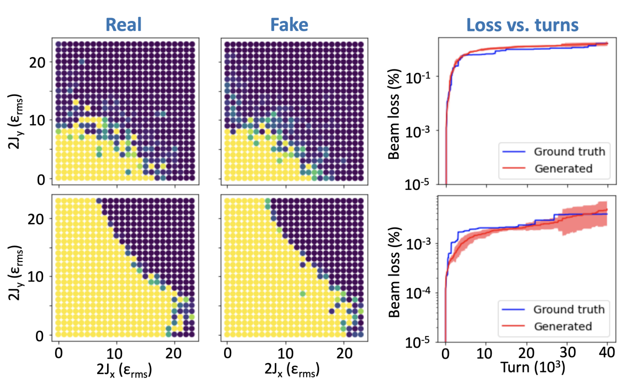

Figure 5: SixTrack output vs model predictions for two different (Q', I_{oct}) settings.

shown in the rightmost column for both the ground truth (blue) and the model prediction (red). This was obtained by assuming a 2D Gaussian particle distribution in the transverse planes and assigning corresponding weights to every particle in amplitude space. The particle loss vs turns predicted by the GAN also provides a confidence bound obtained by generating several fake images and computing their standard deviation. The images demonstrate that generally the model captures the main features of the stability plots. However, at the level of individual particles, there are some differences which have not been captured entirely by the GAN. With the low resolution of only 24 × 24 pixels, every particle has a relatively large associated weight. This means that in the stability plots, discrepancies between the ground truth and the model prediction at the level of individual particles can already have an important effect on the loss vs turns calculation. By increasing the number of samples in both amplitudes, this effect can be strongly reduced, thus improving the accuracy of the model. An overview of the SixTrack output (left) and model predictions (right) over the entire 2D parameter space is given in Fig. 6. The white squares correspond to parameter settings for which no training data was available. The plot illustrates that, at least visually, the GAN generalizes well on unseen (Q', I_{oct}) settings, which is an important requirement of the model. This is still to be evaluated quantitatively. At present the model relies on a smooth dependence of the survival plot on the input parameters which is not always a valid assumption, e.g. for the transverse tunes when crossing resonances. To include information on the resonance lines, one might incorporate resonance conditions analytically or use additional data from frequency map analysis (FMA) [13].

CONCLUSIONS

To determine the optimal parameter settings of a particle accelerator in an automated and efficient manner, we are investigating the potential of combining surrogate modeling with adaptive sampling techniques. As a first step we have studied the use of c-GANs to predict the stability plot for given machine settings. To that end, we have trained the GAN on a small subset of the available parameter space for the LHC using SixTrack simulation data. While these preliminary studies reveal promising results, the performance of GANs and the stability of their training remains to be explored for higher-dimensional parameter spaces and larger amounts of training data.

ACKNOWLEDGEMENTS

This work is partially funded by the SDSC project C18-07.

ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-TUPAB216

REFERENCES

[1] SixTrack, http://sixtrack.web.cern.ch/SixTrack/.

[2] F. Zimmermann, "Introduction to Collective Effects in Particle Accelerators", *ICFA Beam Dyn. News.*, vol. 69, pp. 8-17, 2016. doi:10.1103/PhysRevAccelBeams.24.063401

[3] E. Métral, G. Rumolo, and W. Herr, "Impedance and Collective Effects", in *Particle Physics Reference Library: Volume 3: Accelerators and Colliders*, Springer, 2020.

[4] W. Scandale, "Dynamic aperture", SL-Division, CERN, Geneva, Switzerland, Rep. CERN-SL-94-24-AP, 1994.

[5] E. Todesco and M. Giovannozzi, "Dynamic aperture estimates and phase-space distortions in nonlinear betatron motion", *Phys. Rev. E*, vol. 53, p. 4067, 1996. doi:10.1103/physreve.53.4067

[6] R. Bartolini and F. Schmidt, "Normal Form via Tracking or Beam Data", *Particle Accelerators Vol.*, vol. 59, pp. 93-106, 1998.

[7] A. Abada *et al.*, "FCC-hh: The Hadron Collider", *Eur. Phys. J. Spec. Top.*, vol. 228, pp. 755–1107, 2019. doi:10.1140/epjst/e2019-900087-0

[8] D. Gorissen *et al.*, "A Surrogate Modeling and Adaptive Sampling Toolbox for Computer Based Design", *Journal of Machine Learning Research*, vol. 11, pp. 2051-2055, 2010.

[9] D. Gorissen *et al.*, "Automatic Approximation of Expensive Functions with Active Learning", *Foundations of Computational Intelligence*, vol. 1, pp. 35-62, 2009. doi:10.1007/978-3-642-01082-8_2

[10] A Beginner's Guide to Generative Adversarial Networks (GANs), https://wiki.pathmind.com/generative-adversarial-network-gan

[11] I. Goodfellow *et al.*, "Generative Adversarial Networks", *Proc. International Conference on Neural Information Processing Systems*, pp. 2672–2680, 2014. doi:10.5555/2969033.2969125

[12] M. Mirza and S. Osindero, "Conditional Generative Adversarial Networks", 2014. arXiv:1411.1784

[13] J. Laskar, "Frequency Map Analysis and Particle Accelerators", in *Proc. 20th Particle Accelerator Conf. (PAC'03)*, Portland, OR, USA, May 2003, paper WOAB001.