ISBN: 978-3-95450-210-3 doi:10.18429/JACoW-FEL2019-THP065

# MULTI-OBJECTIVE FEL DESIGN OPTIMISATION USING GENETIC ALGORITHMS

D. J. Dunning*, J. K. Jones, H. M. Castañeda Cortés, N. R. Thompson
STFC Daresbury Laboratory and Cockcroft Institute, Daresbury, UK

*Abstract*

Simulation studies were carried out to optimise the performance of various FEL designs, with examples including longitudinal current profile shaping for a seeded FEL, and selection of the chicane delays for the High-Brightness SASE technique. In these examples multi-objective genetic algorithms were applied to a single section of the overall facility simulation, i.e. the undulator, as is the common approach. Further studies are also reported in which a full start-to-end simulation chain was optimised, with the aim of delivering a more holistic facility design optimisation.

## INTRODUCTION

Simulations are a key component in the design and operation of modern FEL facilities. The full machine is often modelled with a 'start-to-end' (S2E) chain of different simulation codes, each developed for specific sections of the facility (e.g. gun, accelerator, FEL, photon beamline), shown schematically in Fig. 1 (a). Optimisation within this simulation chain is often segmented in the same way, e.g. the accelerator section is often optimised to meet a set of target electron beam properties, which themselves are specified to allow the FEL section to deliver the target photon output, see Fig. 1 (b).

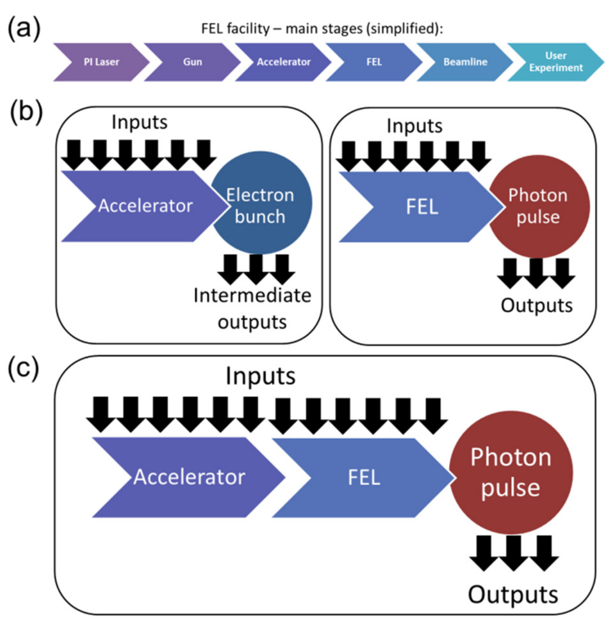


Figure 1: (a) Example of the main stages in a FEL facility, (b) a segmented optimisation approach requires specification of intermediate parameters, (c) joining up stages allows optimisation on the final outputs.

* david.dunning@stfc.ac.uk

A problem with this method is that mapping from electron bunch properties to photon output is a complex nonlinear process. Intermediate parameters such as emittance and bunch length are often overly reductive as a predictor of FEL performance when applied to realistic – sometimes highly non-Gaussian – distributions. One solution is to develop more prescriptive specifications of the intermediate parameters; however this can become increasingly elaborate, as illustrated in the first section of this paper.

An alternative approach, considered in the second section, is to combine multiple stages of the simulation chain into a single optimisation problem, Fig. 1 (c). This allows, e.g. the accelerator parameters to be directly optimised on the FEL output, without the need to specify intermediate targets. Previous studies have combined accelerator simulation codes with FEL analytical models, however not all effects are included [1]. Here we present a combined accelerator + FEL simulation framework with examples.

In the final section a more advanced FEL scheme is considered. In each case multi-objective genetic algorithms (MOGA) are used as the optimisation technique.

## SINGLE-STAGE OPTIMISATION

This section considers an example of optimising a single stage of the S2E chain, as shown in Fig 1 (b).

At the interface between the accelerator and FEL sections a common starting point is to specify the longitudinal profile in terms of peak current and electron bunch length. For non-Gaussian distributions these measures can be poor predictors of FEL performance, such that more refined targets (e.g. bunch shape) are required. A simple case was set up to optimise the current profile for a seeded FEL, using parameters of the CLARA project [2,3]. 3D FEL simulations were carried out using the Genesis 1.3 FEL code [4] (version 2, utilising the OCELOT framework [5]), in which the current profile was described by a 36-element array, entered via the 'beamfile' method (with all other properties kept constant). Optimisation was carried out using the NSGA-II [6] MOGA method, inside DEAP [7] and FEL performance was optimised on peak power and bandwidth at the end of the undulator (further refinement is discussed later).

In the first instance the system started from random current distributions. The mutation effect was to randomly modify a randomly selected subset of the current array elements. The crossover effect was to interleave elements from two existing current profiles. In both cases a further step to maintain constant charge was applied. Figure 2 shows the results of the optimisation. Some effects were expected – concentrating charge into a high current region in order to increase pulse energy: indeed surpassing the pulse energy of a Gaussian but with larger bandwidth.

ISBN: 978-3-95450-210-3 doi:10.18429/JACoW-FEL2019-THP065



However an unexpected alternative solution concentrated charge into a series of high current regions. This is probably not desirable for users but it is an interesting finding. Individual Genesis runs took ~1 minute with 24 processors, Figure 2 took ~2 days to complete 38 generations.

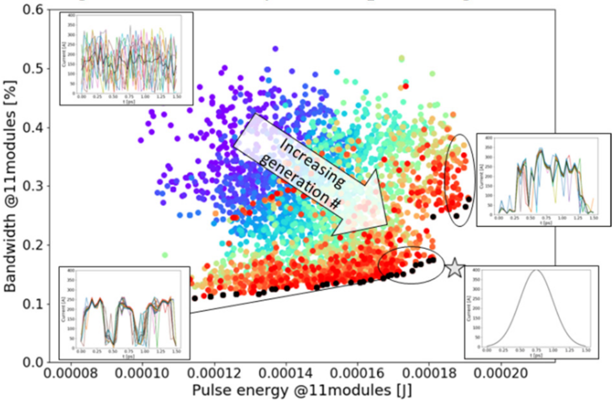


Figure 2: MOGA optimisation of FEL current profile starting from noise (top left) to an approximately flat-top solution (right) or a series of peaks (bottom left). The black points are the Pareto front after 38 generations. The result for a Gaussian profile (star, bottom right) is shown.

Further cases were performed to investigate current profile evolution beyond a Gaussian or a flat-top, as shown in Fig. 3. Some findings are obvious: a long, smooth bunch gives narrow bandwidth/low power; and vice versa. Some are less so, e.g. spiking at the bunch tail.

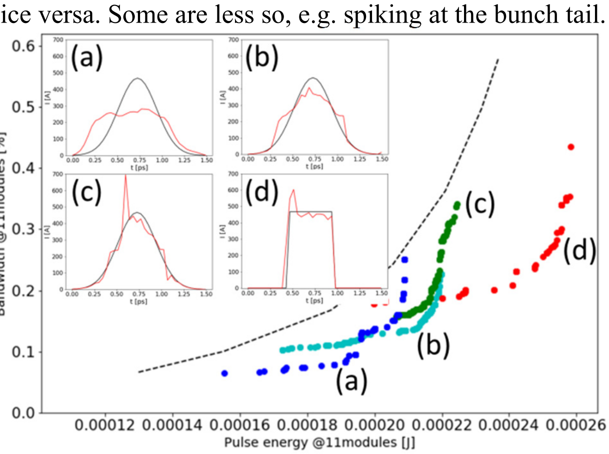


Figure 3: Pareto fronts and example profiles (inset – red) vs starting profiles (inset – black) for four different MOGA optimisations of FEL current profile. The black dashed line shows results for a range of Gaussian profiles.

The benefit of the approach taken in this section is that it helps to understand one property of the bunch in isolation. However the degree of complexity possible in specifying a single property also illustrates the difficulty in specifying a set of intermediate parameters.

## INTEGRATED S2E OPTIMISATION

This section describes combining multiple S2E stages into a single optimisation problem, as shown in Fig. 1 (c), thereby avoiding the issue of intermediate parameters.

The optimisation framework comprises ASTeC's in-house S2E framework (SimFrame) [8] in combination with OCELOT and DEAP as shown in Fig. 4.

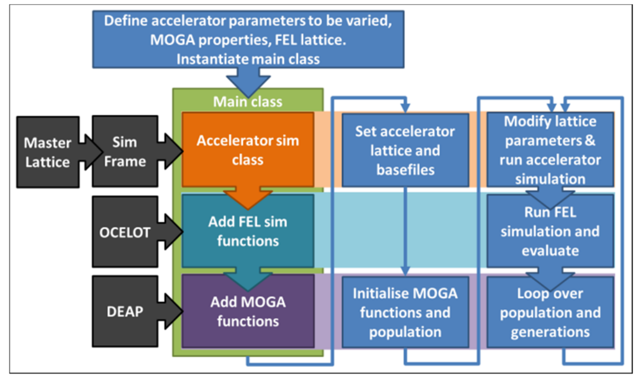


Figure 4: Schematic of S2E optimisation framework.

The Simulation Framework (SimFrame) is a python-based framework for performing accelerator simulations using a transparent interface to multiple tracking codes (ASTRA [9], Elegant [10], CSRTrack [11] and GPT [12]). It interfaces with the Master Lattice [13, 14] (written in YAML [15]), which is the central repository for machine and element information (mechanical and magnetic properties, errors, control system names etc.). SimFrame doesn't interface to a dedicated FEL code, relying instead on OCELOT's python interface to Genesis. Integration between the frameworks was relatively straightforward, allowing S2E from the cathode to the end of the FEL.

As shown in Fig. 4, the main class inherits from the accelerator simulation class (SimFrame functionality), OCELOT (FEL simulation functions) and DEAP (MOGA functions). A python script defines the MOGA parameters, FEL lattice and the accelerator parameters to be varied. Once the main class is instantiated, the accelerator lattice and a set of starting beam files is specified, allowing the simulation to start from a previously simulated location, thereby minimising duplication. DEAP functions are used to generate a population, which is evaluated and modified over a number of generations, with accelerator and FEL simulations performed for each individual.

The MOGA framework was used to optimise the performance of a S2E simulation of the proposed XARA [16] upgrade to CLARA. A key factor to achieving a successful outcome was establishing a simulation method that was suitably detailed while also suitably fast in order to perform the many hundreds of simulations required in a reasonable time. The 'distfile' method of input to Genesis was chosen since it retains the 6D information more completely than the 'beamfile' method, however it requires more macroparticles ($2^{15}$=32k). The baseline injector modelling (up to the exit of linac 1) was performed in ASTRA with $2^{18}$ (262k) particles and then sampled to create the 32k macro-particles used in Elegant. Individual runs took ~10 minutes, split 2/8 mins in Elegant/Genesis. The input parameters varied were: phase and amplitude of linacs 2-6 and the 4th harmonic cavity; bunch compressor angle; de-chirper gap; and laser heater interaction strength (modelled as momentum scattering). For simplicity the

ISBN: 978-3-95450-210-3 doi:10.18429/JACoW-FEL2019-THP065

FEL mode was SASE rather than seeded and the target FEL parameters were maximum pulse energy and minimum bandwidth, evaluated at the position of maximum brightness to allow for different saturation lengths. Fig. 5 shows the Pareto front in the pulse energy-bandwidth space, with a clear trade-off evident between the two. Fig. 6 shows two individuals at different parts of the front – examples of the range of different radiation pulse durations, and corresponding electron bunch properties.

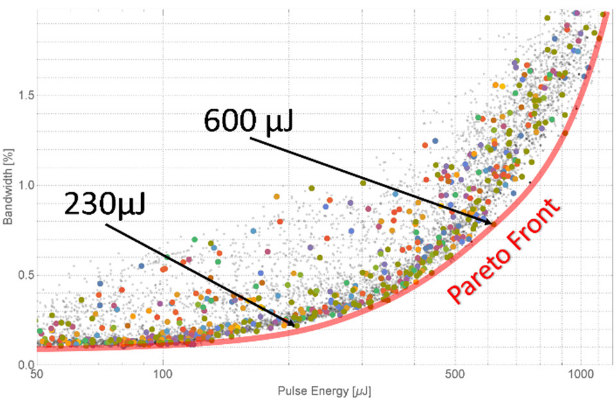


Figure 5: Final Pareto front and solutions for each generation of the S2E MOGA run. Individual points are shown in grey. The two example solutions are highlighted.

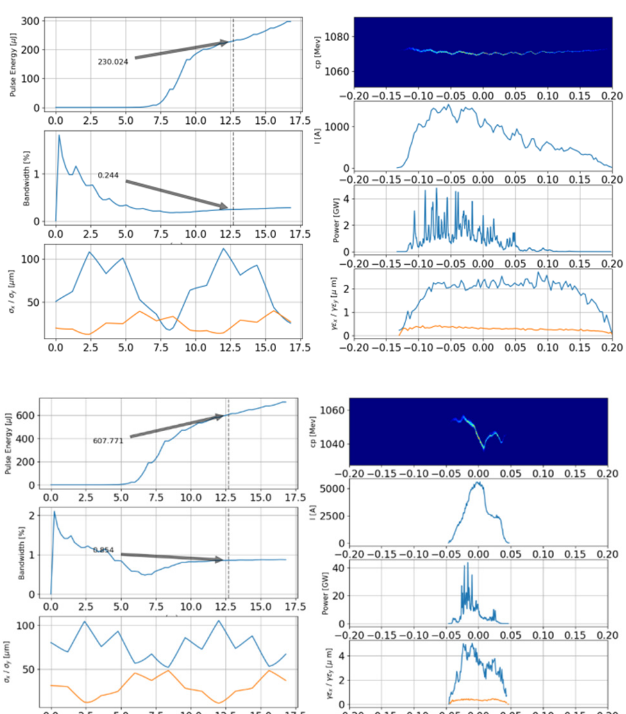

Figure 6: FEL output and bunch parameters (***left***: pulse energy, bandwidth and beam-size, ***right***: longitudinal phase space, current, power, emittance) for example solutions at 230 μJ (top) and 600 μJ (bottom).

## STABILISED HB-SASE

The MOGA framework was applied to the simultaneous optimisation of two FEL schemes which utilise the same configuration of delay chicanes inserted between undulator modules. In the first scheme, HB-SASE [17], the delay chicanes increase slippage and hence coherence length. Here we use dipole-only chicanes as these are more compact than isochronous chicanes. Studies using such chicanes in monotonically increasing or decreasing delay sequences show an increase in coherence length limited to a factor of 5-10. In this implementation the first fitness value (Figure 7, horizontal) is the FEL pulse energy multiplied by the coherence length, representing scaled brightness. We average over 9 shot noise seeds per data point. The second scheme aims to improve the shot-to-shot stability of the SASE FEL by varying the chicane dispersion to manipulate the electron bunching, and introduce a passive, negative feedback into the FEL mechanism [18]. The second fitness value (Figure 7, vertical) is the rms. of the scaled brightness over the shot noise seeds.

Figure 7 shows initial results, with the fitness values calculated at 13 m along the undulator, and tuned to 100 nm at 250 MeV/c. The points are colour coded from dark blue to light green as the generations increase. The 'Hall of Fame' (HOF) points are shown along with the Pareto front, indicated with the dotted red line. The equivalent SASE scaled brightness and rms. brightness variation are shown as green diamonds, for undulator lengths from 11–17 m. The delays corresponding to maximum brightness and minimum fluctuation are shown.

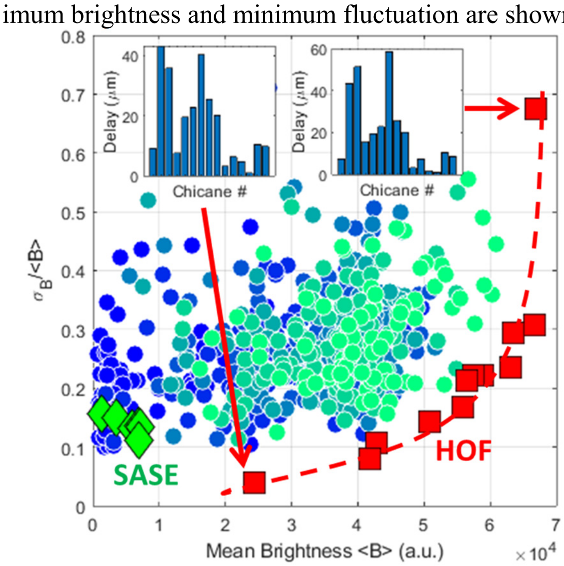


Figure 7: Initial results of optimisation to maximise $\langle B\rangle$, HB-SASE output brightness averaged over an ensemble of shot noise seeds, and minimise $\sigma_B/\langle B\rangle$, rms shot-to-shot brightness fluctuation within the ensemble.

These results show solutions with brightness increased by an order of magnitude but with increased fluctuations, and solutions with slightly less brightness increase but with fluctuations damped compared to SASE. It should be noted that in regular SASE the fluctuations should scale

ISBN: 978-3-95450-210-3 doi:10.18429/JACoW-FEL2019-THP065



approximately as $\sigma_B/\langle B\rangle \propto \langle B\rangle$ so the results favourably violate this scaling. The chicane settings for maximum brightness and minimum fluctuations appear qualitatively similar to each other, while very different to the monotonically varying sequences employed in previous studies. These are interesting findings for further investigation.

## REFERENCES


[1] R. Bartolini, M. Apollonio and I. P. S. Martin, "Multi-objective genetic algorithm optimization of the beam dynamics in linac drivers for free electron lasers", *Phys. Rev. ST Accel. Beams*, vol. 15, no. 3, p. 030701, 2012. doi:10.1103/PhysRevSTAB.15.030701

[2] J. A. Clarke *et al.*, "CLARA conceptual design report," *Journal of Instrumentation*, vol. 9, no. 05, p. T05001, 2014. doi:10.1088/1748-0221/9/05/T05001

[3] D. J. Dunning *et al.*, "Start-to-End Simulations of the CLARA FEL Test Facility", in *Proc. 9th Int. Particle Accelerator Conf. (IPAC'18)*, Vancouver, Canada, Apr.-May 2018, pp. 4430-4433. doi:10.18429/JACoW-IPAC2018-THPMK060

[4] S. Reiche, "GENESIS 1.3: a fully 3D time-dependent FEL simulation code," *Nucl. Instr. Meth. Phys. Res. Sect. A*, vol. 429, p. 243, 1999. doi:10.1016/S0168-9002(99)00114-X

[5] I. Agapov, G. Geloni, S. Tomin, and I. Zagorodnov, "OCELOT: A software framework for synchrotron light source and FEL studies," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 768, pp. 151-156, 2014. doi:10.1016/j.nima.2014.09.057

[6] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," in *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182-197, April 2002.

[7] Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau and Christian Gagné, "DEAP: Evolutionary Algorithms Made Easy", *Journal of Machine Learning Research*, pp. 2171-2175, no 13, July 2012.

[8] ASTeC, https://github.com/VELA-CLARA-software/OnlineModel/tree/master/SimulationFramework

[9] K. Flöttmann, "ASTRA - A Space Charge Tracking Algorithm," 2000. http://www.desy.de/~mpyflo

[10] M. Borland, "User's Manual for elegant," APS-ANL, Chicago, IL, 2017.

[11] M. Dohlus, "CSRtrack code", http://www.desy.de/fel-beam/csrtrack/

[12] Pulsar Physics, "General Particle Tracer.", http://www.pulsar.nl/gpt/

[13] T. J. Price *et al.*, "Virtual VELA-CLARA: The Development of a Virtual Accelerator", in *Proc. 9th Int. Particle Accelerator Conf. (IPAC'18)*, Vancouver, Canada, Apr.-May 2018, pp. 4773-4776. doi:10.18429/JACoW-IPAC2018-THPML060

[14] ASTeC, https://github.com/VELA-CLARA-software/OnlineModel/tree/master/MasterLattice

[15] K. Simonov, https://github.com/yaml/pyyaml

[16] D. J. Dunning, J. K. Jones, and L. S. Cowie, "XARA: X-Band Accelerator for Research and Applications", presented at the 39th Int. Free Electron Laser Conf. (FEL'19), Hamburg, Germany, Aug. 2019, paper THP066, this conference.

[17] B. W. J. McNeil, N. R. Thompson, and D. J. Dunning, "Transform-Limited X-Ray Pulse Generation from a High-Brightness Self-Amplified Spontaneous-Emission Free-Electron Laser", *Phys. Rev. Lett.*, vol. 110, p. 134802, 2013. doi:10.1103/PhysRevLett.110.134802

[18] N. Thompson, "Possible Method for the Control of SASE Fluctuations", in *Proc. 38th Int. Free Electron Laser Conf. (FEL'17)*, Santa Fe, NM, USA, Aug. 2017, pp. 129-131. doi:10.18429/JACoW-FEL2017-MOP039