ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-TUPAF19

pyaopt OPTIMIZATION SUITE AND ITS APPLICATIONS TO AN SRF CAVITY DESIGN FOR UEMS*

Ao Liu†, Roman Kostin, Euclid Techlabs, Bolingbrook, IL, USA
C. Jing, Pavel Avrakhov, Euclid Beamlabs, Bolingbrook, IL, USA

Abstract

Designing and commissioning particle accelerators need great optimization efforts. This is particularly true for large accelerators with complex components that provide stable beam such as light sources and colliders, where nonlinearities of the beam play an important role. Currently, many design optimizations are provided by individual user-created automated problem-finding and solution-proposing algorithms, which requires an extensive amount of computing resources. Heuristic algorithms such as Genetic Algorithms (GA) and Simulated Annealing (SA) are commonly implemented. They are either created for individual tasks, or are implemented directly in simulation codes, such as OPAL or IMPACT3D. An optimization suite that is independent of the accelerator codes is needed for general application studies. Meanwhile, researchers now have access to the HPC resources, which can be utilized for parallelization of codes. We propose a Python-based optimization suite for general applications. In this paper, we introduce the pyaopt suite by giving some details of its applications, including a design of an SRF photogun for UEMs.

INTRODUCTION

Recently, there has been multiple new applications of heuristic algorithms in the particle accelerator community. The fields include secondary particle collection [1], DA optimizations [2, 3], and space charge calculations [4]. In most of these cases, algorithms were customized for specific physics problems, or built in a specific simulation program. In fact, the number of programs that include the Genetic Algorithm (GA) as the multi-objective optimizer is rapidly increasing [5, 6]. However, for many accelerator physicists and engineers, these algorithms are still inaccessible to some extent: there is no easy way to use them in a "plug and play" fashion.

The design of Python advanced optimization pyaopt suite aims at delivering a package that has an API for users to conveniently describe the optimization problem, select the optimization algorithm and start the job. It not only includes widely-accepted algorithms such as the GA, Simulated Annealing (SA) and the Particle Swarm Algorithm (PSA), but also gradient-based (deterministic) algorithms, such as the Gauss-Newton method, etc. The goal of the Python-based package is to let users run optimization jobs in any environments, including a personal computer, a small-scale cluster, or a HPC supercomputer. Users may select the mode such

* SRF cavity design work supported by DOE under contract DE-SC0018621
† a.liu@euclidtechlabs.com

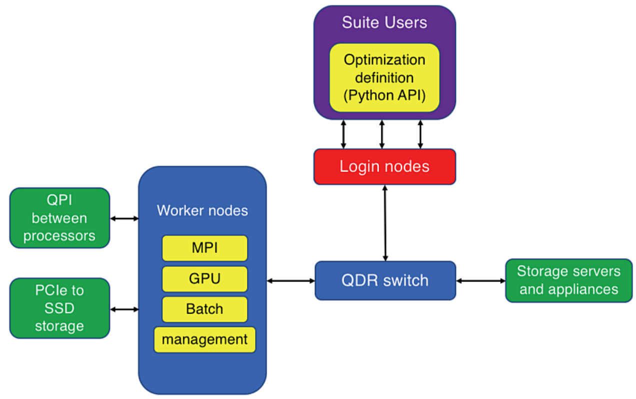

Figure 1: The flowchart of running optimization jobs on an HPC machine.

that the pyaopt job manager can handle the job submission, monitoring, and logging. The idea of running the jobs on an HPC is illustrated in Figure 1.

pyaopt includes a few customized metaheuristic algorithms and some deterministic algorithms. We introduce a selection of the metaheuristics below:

- pyaopt-GA, which is based on the NSGA-II [7] and can do both single-objective GA (SOGA) and multi-objective GA (MOGA). The customization is on the crowding distance (CD) of individuals, which represents the similarities of them, and the "rescue method": judgment day, which is used only when the whole population seizes improving prematurely. The algorithm is enhanced by MPI [8], such that calculations of dominance and CD are distributed on different ranks.

- pyaopt-SA, which is based on the standard annealing formula $P = e^{(f(x)-f(x')/T)}$ for $f(x) < f(x')$, where $f(x)$ is the fitness value for solution x, and T is the current temperature. The customization is on the adaptive cooling schedule, ΔT per iteration, and on the cooling range assignment for MPI implementation. Although it is already a common practice to normalize the fitness value to an expected one and implemented by many SA users, using an adaptive cooling schedule further helps to prevent the system to converge prematurely. As for the search range where multiple workers are present, users can choose how many slices each variable's range needs to be divided, based on the number of workers available. Then pyaopt allocates each combination of range slices evenly to the workers.

- pyaopt-ANN, which is based on multiple artificial neural network (ANN) algorithms. The parallelization is

ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-TUPAF19

done through both the forward and backward propagations of data, in simply a batch fashion.

pyaopt can be installed on multi-platforms, thanks to the installation capabilities brought by the setuptools [9] package. Our idea is to make this process painless, such that modules targeted for different computing accelerator architecture that are available on the machine can be automatically detected by the setup code, or specified by the users (when the installation frontend cannot see the heterogeneity from the worker's point of view.

The variable ranges are specified in a JSON [10] file (we are also investigating the HDF5 [11] format to contain the metadata of all the input, output and log files). The API provides the proposed combinations of variables (each combination of variable values is hereafter referred as "individual"), and users specify the functions to be called by pyaopt for evaluation of individuals.

EXAMPLES OF APPLICATIONS

In this section we use some applications of pyaopt to demonstrate its capability to be used on different areas of accelerator physics.

nuSTORM Magnetic Horn

The neutrinos from STORed Muons (nuSTORM) uses a magnetic horn to capture the secondary pions generated from bombarding a long target rod with high energy protons. Because of the finite length of the target rod, the original point-to-parallel principle of a simple horn design with double parabolic surface is no longer optimal. The horn has to be re-designed for each target that has different materials, lengths and diameters, even when the primary proton beam parameters are fixed. In this example, we used a 46 cm Inconel target, with a radius of 3 mm. We used the MOGA for this purpose, where one of the two objectives is to maximize the number of pions within the transverse phase space, and other is to maximize that in a the momentum acceptance described by a derived formula. Figure 2 shows the variation of fitness values for the dominant elite candidate in each generation. This treatment of converting a single-objective, time-consuming multiparticle tracking-based optimization was efficient in dramatically saving the optimization time and increase the acceptable pions at the end of the pion beamline by 13%.

nuSTORM Muon Storage Ring

The physics objective is to store as many muons in the 2 mm·rad full transverse admittance and 3.8 ± 10% GeV/c. Since this is an extra large beam, multiple nonlinearity terms of the beam optics become critical as stop bands for beam circulation in the ring. Sextupoles are introduced in the lattice, in both standalone sextupole magnets and also as combined-function Therefore, instead of isolating the nonlinearity terms one by one and evaluate the importance of each, we chose to directly rely on mult-particle tracking result as the single optimization objective. Since 90% of the muons will decay to neutrinos in approximately 100 turns, the percentage of survived muons after 100 turns was used as the evaluation function. We then chose SA for this task. The variation of fitness value per generation is shown in Figure 3. The momentum acceptance is compared for before and after the sextupole correction with the optimized setting in Figure 4.

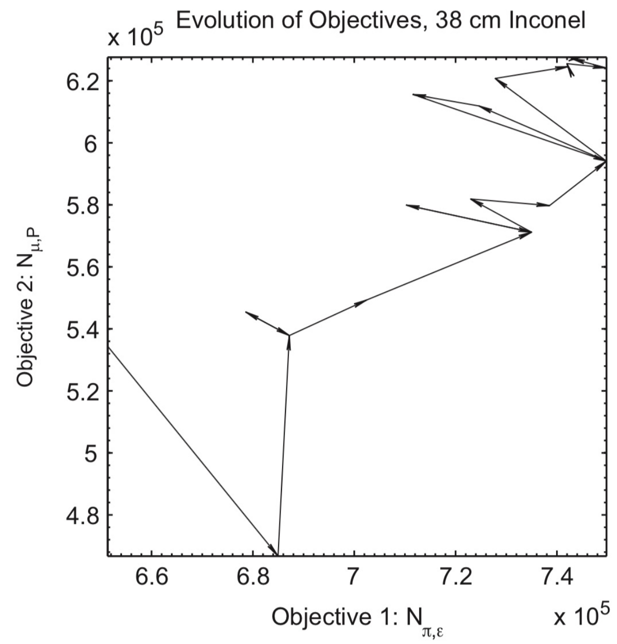

Figure 2: The variation of fitness values of the dominant elite candidate in each generation.

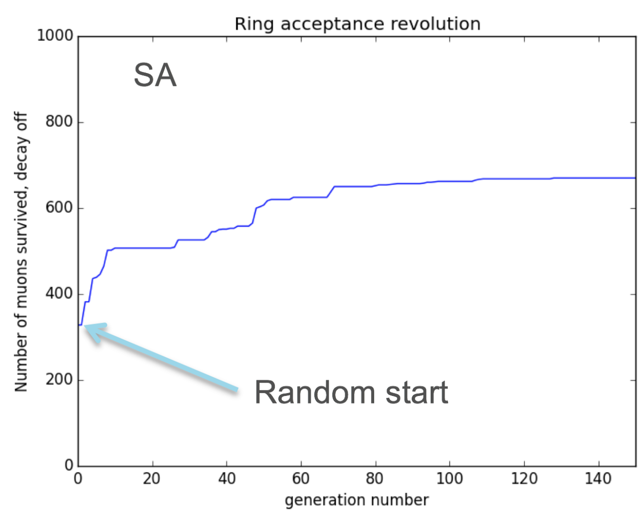

Figure 3: The variation of fitness value of the best solution in each generation.

The suite was also used on the optimization for a Step IV lattice of the international Muon Ionization Cooling Experiment (MICE) [12], 6D ionization cooling channel, etc. Because the suite was designed for general optimization problems, we foresee more areas of applications. In the next

ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-TUPAF19

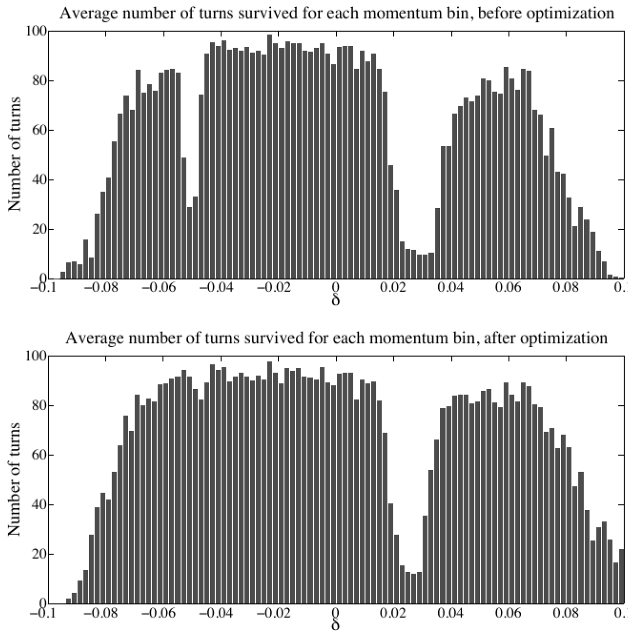

Figure 4: The improvement of momentum acceptance after the sextupole correction with the optimized setting.

section we discuss about its application on the design of an SRF photogun for UEMs.

SRF PHOTOGUN FOR UEMS

Benefiting from the rapid progress on RF photocathode gun technologies in the past two decades, the development of MeV-range ultrafast electron diffraction/microscopy (UED and UEM) has been identified as an enabling instrumentation, which may lead to breakthroughs in fundamental science and applied technologies [13–15]. In a UED/UEM, stable femtosecond (fs) electron bunches that are synchronized with fs laser pulses is required. Currently, there are room temperature RF photocathode electron guns for generating MeV electrons for UED/UEM. However, the shot-to-shot stability for those machines is still low to fully satisfy requirements from the UED/UEM community. Here we propose a 1.3 cell, 1.3 GHz SRF cavity as the UEM electron gun. The innovations of this structure include but are not limited by:

- It uses a Euclid-designed, ILC type SRF cavity cell with a novel detachable coupler, which was inherited from our previously completed DoE SBIR project (DE-SC0002479). The advantage of using this cell is that the manufacturing and operation time for the whole SRF cavity is dramatically reduced.
- It uses the backwall of the first 0.3 cell as the photocathode, where the quantum efficiency (QE) of the high RRR Niobium (Nb) is up to 10^{-5}.
- By using the novel technologies of conduction cooling and coating of Nb_3Sn, which are what Euclid and Fermilab are collaborating on now, the peak axial electric field (E_z) can reach 26 MV/m. Moreover, the conduction cooling allows one to use a cryocooler, without liquid helium, to cool down an SRF structure. See Figure 5 for a COMSOL simulation of the conduction cooling scheme. See also [16] for the published news on the scheme by Fermilab.

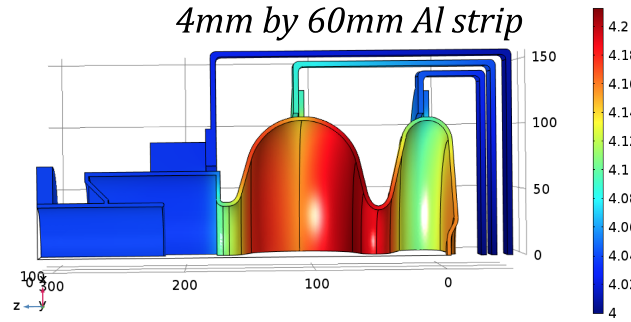

Figure 5: Simulation of the conduction cooling scheme in COMSOL. Figure courtesy of R. Kostin, Euclid Techlabs.

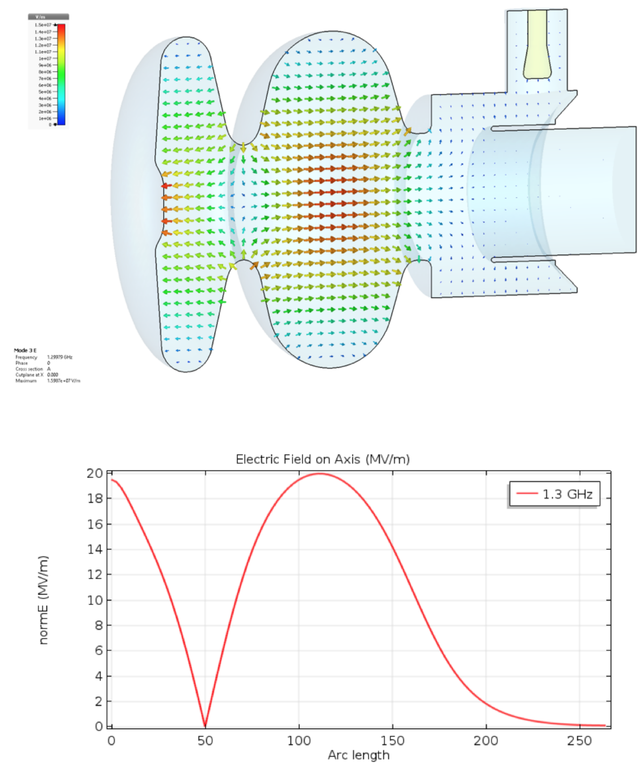

Figure 6: CST simulation of the 1.3 cell SRF cavity and the corresponding axial field, normalized to a maximum value of 20 MV/m.

Figure 6 shows the designed cavity in CST, and the corresponding field scaled to a conservative estimate of peak axial E_z of 20 MV/m.

The back wall geometry was preliminarily optimized to provide transverse RF focusing when the beam is generated at the cathode. It has a unique “step” design where the flat face is used for photocathode, and the curved geometry provide the transverse field needed. pyaopt is able to parallelize

ISBN: 978-3-95450-200-4 doi:10.18429/JACoW-ICAP2018-TUPAF19

the RF simulation in Superfish [17] on a Linux cluster or the OSX platform via using a WineHQ container. The resultant field is then used in Astra [18] for multi-particle tracking. In Table 1, beam parameters suitable for UED/UEM applications, simulated in Astra with space charge effect with the gun design shown in Figure 6, are listed and compared. Further more thorough optimizations will be done in the future studies.

Table 1: Beam parameters for UED/UEM Applications

Parameter name [unit]		Value	
Application		UED	UEM
Beam Energy	[MeV]	1.655	1.655
Charge per pulse	[fC]	5	500
Laser pulse length	[fs]	6.4	6.4
Laser spot size	[µm]	36	180
Bunch length	[fs]	167	741
Beam emittance	[nm]	6.6	39
Relative energy spread	[1]	1.3×10^{-5}	6.4×10^{-5}

FUTURE WORK

The Python wrapper, pyCUDA, will be implemented in pyaopt in the future to utilize users' NVIDIA GPU accelerators, or GPU on NERSC [19]. Furthermore, in order to implement algorithms that are more robust against noises, such that the suite can be deployed on experimental jobs, we will add the RCDS [20] algorithm to pyaopt soon in the future. More test cases, such as a collaboration with lattice design work at BNL [3] will also be considered.

CONCLUSIONS

We are actively developing a Python-based optimization suite, pyaopt to let users conveniently describe and run optimization problems on personal computers, small-scale clusters or HPC supercomputers. pyaopt includes a selection of deterministic and metaheuristic algorithms and allow users to run them in parallel mode. We showed two test cases for the GA and SA of pyaopt on nuSTORM related studies. The algorithm was also tested by cases of optimizations for MICE and 6D ionization cooling channel designs. In all cases the suite works efficiently in reducing the computing time and finding optimal solutions. By using WineHQ, we are able to combine RF and tracking simulations by Superfish and Astra on OSX or Linux platforms. A preliminary optimization on the SRF photogun design shows promising beam qualities for it to be applied to UED/UEM. The SRF photogun has multiple advantages over the room-temperature photoguns, including its superstability, CW mode operation enabled by the conduction cooling, etc.

ACKNOWLEDGMENTS

The SRF photogun design work is supported by DoE SBIR grant DE-SC0018621, and the current optimization application study is supported by DoE SBIR grant DE-SC0013121. The authors gratefully thank Dr. A. Bross, D. Neuffer from Fermilab, and Dr. P. Snopok from IIT for their inputs on the muon accelerator-related optimizations.

REFERENCES

[1] A. Liu *et al.*, "Design and Simulation of the nuSTORM Pion Beamline", *Nuclear Inst. and Methods in Physics Research* A, 2015. doi:10.1016/j.nima.2015.08.020

[2] A. Liu *et al.*, "A FODO racetrack ring for nuSTORM: design and optimization", *JINST*, 2017. doi:10.1088/1748-0221/12/07/P07018

[3] Y. Li *et al.*, "Genetic Algorithm Enhanced by Machine Learning for Dynamic Aperture Optimization", presented at ICAP'18, Key West, FL, USA, Oct 2018, paper SAPAF01, this conference.

[4] J. Qiang, "Advances in Simulation of High Brightness/High Intensity Beams", presented at ICAP'18, Key West, FL, USA, Oct 2018, paper SAPLG01, this conference.

[5] OPAL http://gitlab.psi.ch/OPAL/src/wikis/home

[6] ELEGANT, https://www.aps.anl.gov/Accelerator-Operations-Physics/Software#elegant

[7] K. Deb *et al.*, "A fast and elitist multiobjective genetic algorithm: NSGA-II", IEEE Transactions on Evolutionary Computation, vol. 6, no. 2, pp. 182 - 197, Apr 2002. doi:10.1109/4235.996017

[8] Open MPI, https://www.open-mpi.org/

[9] setuptools, https://setuptools.readthedocs.io/en/latest/

[10] JSON, https://www.json.org/

[11] HDF5, https://www.hdfgroup.org/HDF5/

[12] A. Liu, on behalf of the MICE collaboration, "MICE Step IV Optics without the M1 Coil in SSD", in *Proc. IPAC'16*, Busan, Korea, May 2016. doi:10.18429/JACoW-IPAC2016-TUPMY006

[13] T. Chase *et al.*, "Ultrafast electron diffraction from non-equilibrium phonons in femtosecond laser heated Au films", *Applied Physics Letters*, vol. 108, no. 4, pp. 041909, 2016. doi:10.1063/1.4940981

[14] Ming-Fu Lin *et al.*, "Ultrafast non-radiative dynamics of atomically thin MoSe2" Nature Communications, vol. 8, pp. 1745, 2017. doi:10.1038/s41467-017-01844-2

[15] Xiaoxi Wu *et al.*, "Light-induced picosecond rotational disordering of the inorganic sublattice in hybrid perovskites", *Sci. Adv.*, vol. 3, no. 7, pp. e1602388, 2017. doi:10.1126/sciadv.1602388

[16] Fermilab news, https://science.energy.gov/hep/highlights/2018/hep-2018-10-b/

[17] Superfish, LANL, https://laacg.lanl.gov/laacg/

[18] Astra, DESY, http://www.desy.de/~mpyflo/

[19] NERSC computing center, http://www.nersc.gov

[20] X. Huang *et al.*, "An algorithm for online optimization of accelerators", *NIM* A, vol. 726, pp. 77-83, Oct 2013. doi:10.1016/j.nima.2013.05.046