Tuning Parameters in the Genetic Algorithm Optimization of Electrostatic Electron Lenses

Hesam Mahmoudi Nezhad, Neda; Ghaffarian Niasar, Mohamad; Hagen, Cornelis W.; Kruit, Pieter

DOI
10.1109/NEMO56117.2023.10202293

Publication date
2023

Document Version
Final published version

Published in
Proceedings of the 2023 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)

Citation (APA)
Hesam Mahmoudi Nezhad, N., Ghaffarian Niasar, M., Hagen, C. W., & Kruit, P. (2023). Tuning Parameters in the Genetic Algorithm Optimization of Electrostatic Electron Lenses. In *Proceedings of the 2023 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO)* (pp. 170-173). (2023 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization, NEMO 2023). IEEE.
https://doi.org/10.1109/NEMO56117.2023.10202293

2023 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO) | 979-8-3503-4740-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/NEMO56117.2023.10202293

Tuning Parameters in the Genetic Algorithm Optimization of Electrostatic Electron Lenses

Neda Hesam Mahmoudi Nezhad[1], Mohamad Ghaffarian Niasar[2] , Cornelis W. Hagen[1] and Pieter Kruit[1]

[1]Faculty of Applied Physics, Department of Imaging Physics, Delft University of Technology, The Netherlands

[2]Faculty of Electrical Engineering, High Voltage Technologies, Delft University of Technology, The Netherlands

***Abstract* — The design of electrostatic electron lenses involves the choice of many geometrical parameters for the lens electrodes as well as the choice of voltages applied to the electrodes. The purpose of the design is to focus the electrons at a specific point and to minimize the aberrations of the lens. In a previous study, genetic algorithm optimization was introduced to aid the designer. For speeding up the electrostatic field calculations, new methods for analytical approximations of the field near the optical axis were introduced. In this paper, the influence of the main tuning parameters of the Genetic Algorithms is analyzed. The analysis is performed on a typical electrostatic lens systems including 6 electrodes. Different combinations of population sizes and number of generations are taken and the quality of the optimized lens system is compared. It is seen that within a constant computational effort (time or total number of system evaluations), a lower population size with a larger number of generations can achieve better results compared to having larger population size and fewer generations. The combination of Crossover Heuristic with Mutation Gaussian showed significantly better results compared to all other combinations of Mutations and Crossovers. Crossover Fraction is also evaluated to find the most suited values of this parameter.**

***Index Terms* — Genetic Algorithms, Tuning Parameters, Electrostatic Lens, Lens Design Optimization.**

I. INTRODUCTION

Optimization routines such as Genetic Algorithms (GAs), though very effective for finding the optima in complex functions, have not been extensively used for the optimization of electrostatic electron lenses. In such lenses, the objective function is a combination of obtaining the correct focus position and the minimization of lens aberrations. For calculating these, one needs to find the electric field of the lens which is generally calculated by accurate methods such as the Finite Element Method (FEM). To perform the optimization while all geometries and voltages of the lens electrodes may vary, thousands of systems need to be evaluated. Using the accurate field calculation methods such as FEM (60 seconds per system evaluation on a modern PC), the optimization takes a very long time, up to several days [1- 2]. In 2018, we presented an optimization technique based on a fast but approximate model to calculate the fields around the optical axis [1, 3, 4] (0.4 second per system evaluation on the same PC). We combined this with an accurate method to calculate the field in a second optimization step. In a further search to reduce the computation time, we now analyze the influence of the tuning parameters of the Genetic Algorithm. We shall vary the population size, the number of generations, mutation method and cross-over type. In this study there is no need of fine tuning the electron lens by an accurate field calculation, so only our approximate method (the second order electrode method, SOEM) is implemented to calculate the electric field. The study is performed on a lens system having 6 electrodes.

II. OPTIMIZATION PROBLEM

An example of an electrostatic lens with 6 lens electrodes is selected as the case-study to perform the optimization. A cross-section of the round lens is shown in Fig 1. The free variables for the optimization are the thicknesses (T_i), Radii (R_i) and voltages (V_i) of each electrode, and the gaps between the electrodes (G_i). In total, there are 23 free variables.

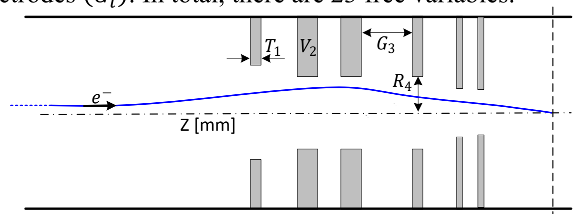


Fig. 1. A 2D illustration of a typical multi-electrode lens systems with 6 electrodes.

Any imaging system such as the electrostatic lens system suffers from aberrations. The smaller the aberrations, the higher the resolution of the image and therefore the higher the quality of the lens system. The aberrations can be calculated by aberration integrals, using the electric field on axis and a first order (aberration-free) trajectory. These aberrations can be combined into a contribution to the spot size when the lens is used to image an electron source on a sample as is done in a scanning electron microscope. The objective function for the optimization problem is hence the spot size at the image side. In our case-study it is presumed that the lens only suffers from spherical and chromatic aberrations. The spot size (presented by D_s in eq1.) can be calculated using the equation below [5].

$$D_s^2 = (0.18\, C_s\, \alpha^3)^2 + (0.6\, C_c\, \alpha\, \Delta U/U)^2 \tag{1}$$

Where C_c and C_s stand for chromatic and spherical aberration coefficients, respectively. α (the half opening angle of the beam) is taken as 10 milliradian. U and ΔU (the acceleration energy and the energy spread of the electron source) are chosen here to be 1 kV and 1 eV, respectively. The constraint of this optimization problem is to have the image at a fixed positon X_C (at 15mm). X_C is also a function of the electric field and can be calculated using ray-tracing. In our case-study MATLAB is used as the programing language. To calculate the objective function and image position (i.e. D_s and X_C), the field calculation is performed by SOEM and our ray-tracing codes use the paraxial approximation. The computational work related to this study is performed on a PC with an Intel (R) Xeon (R) W-2123 CPU @3.60 GHz and 32 GB of RAM.

III. GA Tuning Parameters

A. Population size and number of Generation

The population size (P) is determined by the number of members (here electrostatic lens systems) in each generation. The number of generations in an optimization execution is called G. To perform the assessment, different combinations of population sizes and number of generations are taken with the values 20, 50,100, 200 and 500, so 25 combinations in total. Since in GA each run with identical parameters can yield a different optimization result, the optimization is run 10 times to give statistically reliable results. The results are represented in Fig. 2. The average, maximum and minimum of the Objective Function (OF) values are given in blue, gray and green bars, respectively. The black thin bars inside the blue bars illustrate the standard deviation in the averaged OF values in 10 runs. The corresponding times are given in Fig 2.b. Note that in our optimization problem the constraint function is added to the objective function. By giving the lens systems which did not satisfy the constraint a very high contribution to the objective function, the lens systems which do not satisfy the constraints are automatically thrown out of the solution pools. To evaluate the GA performance, OF values and the execution time are the two factors which should be evaluated together. It can be seen in Fig. 2.a. that, as expected, increasing the population size and the number of generations, the average value of the objective function decreases (visualized by the dashed yellow lines). From Fig. 2.b it can be seen that at very high population sizes and number of generations, the execution time increases dramatically while the OF values shows no marked improvement. Hence, it can be concluded that if GA is run for a shorter time-frame, a larger improvement can be recognized in that short period rather than longer. This conclusion is in line with the investigations performed on GA in other optimization problems [6].

Another study on population and generation is to evaluate the GA performance within a fixed time, that is with a specified number of system evaluations $N_{PG} = P.G$; the important question is then "for a fixed value of N_{PG}, which combination of P and G achieves the better result?". To study this, the cases

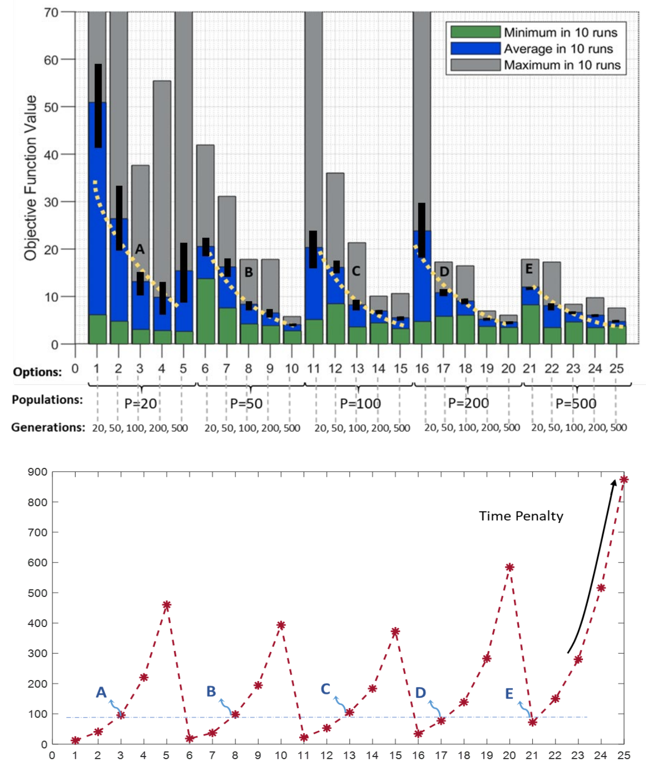


Fig. 2. a. GA performance for 25 different combinations of P and G (shown in the x-axis) b. The corresponding time for each of the 25 cases.

with the same N_{PG} having different combinations of P and G are picked from Fig.2.a. and shown together in Fig.3. The options with smaller G and larger P are shown in dark blue bars, and the options with larger P and smaller G are shown by light blue bars. The black thin bars inside the blue bars illustrate the standard deviation in the averaged OF values in 10 runs.

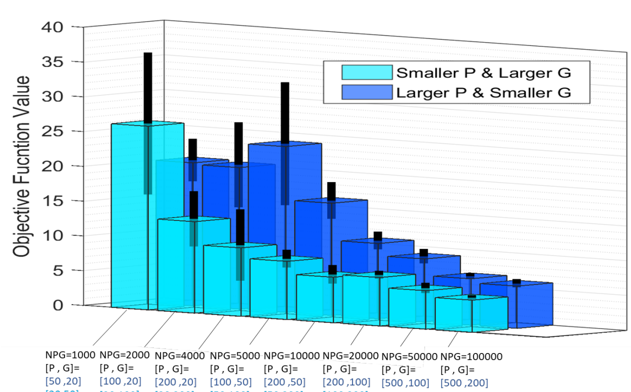


Fig. 3. Graphs of GA runs to illustrate comparison of similar values of N_{PG}, having different combinations of P and G.

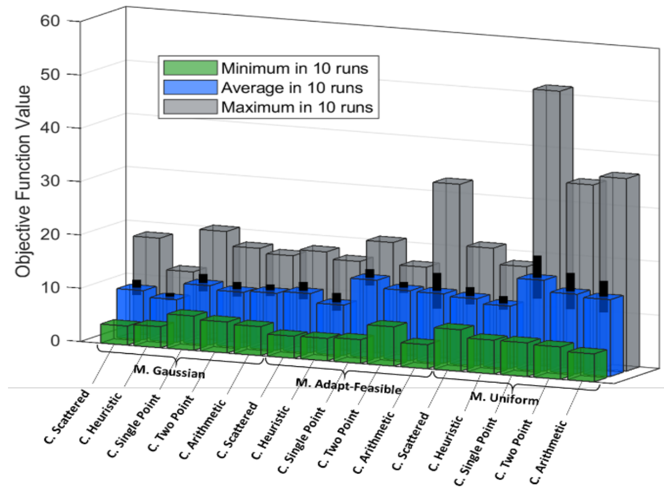
In all cases except the first one (i.e. N_{PG} =1000), the light blue bars are lower than the dark blue bars. Note however that the first case shows an unstable result as can be concluded from the large standard deviation. Clearly N_{PG} =1000 is too small for the optimization with so many free parameters. It is hence concluded that for the same amount of evaluations N_{PG} , a GA optimization with a higher number of generations and a smaller population size achieve a better result than a larger population size and a lower number of generations. This conclusion is in contrast with what has been reported in [7]. However, there were also other studies [8] in line with what we conclude here. In [8] it is shown that a large number of generations is better when the optimization problem has many basins of attractions, with multiple local minima in the objective function landscape. In such situations, having a large population size would not help GA to search the area more extensively, but it will degrade the GA performance since it will cause GA to be trapped in the wrong basin of attractions and stay in a local minimum. So probably our situation is like that.

To continue the rest of analysis on other GA tuning parameters, a fixed value of P and G is taken at which the computation time was not very long, while the objective function was reasonably small. For this aim, option 8 (pointed out by 'B' in Fig.2.b.) with N_{PG}=5000 (P=50, G=100) and run time ~100 Sec, is selected.

B. Crossover and Mutation

This section is devoted to discovering the most suitable options of the two main tuning parameters of GA, Crossover and Mutation. The mutation type concerns the distribution of the random changes in the population within one generation. The crossover type determines how the offspring in a next generation is formed from parents in the earlier generation.

There are 3 different Mutation methods namely Gaussian, Adapt-Feasible and Uniform, and 5 different options for Crossover; Scattered, Heuristic, Single point, Two point and Arithmetic available in MATLAB (in total 15 different combinations). Fig. 4.a. gives the results of the comparison. The Y-axis shows the average value (blue bars), the maximum (gray bars) and minimum value (green bars) of the objective function value for 10 runs per combination. As can be seen from Fig.4.a., the best performance comes from the combination Crossover Heuristic and Mutation Gaussian. However, to be able to better compare the options, a 2D graph for averaged values of the objective functions, in a categorized manner (grouped by their different Mutation methods) is given in Fig.4.b. Looking at this figure, by comparing bars with the same colors, it is seen that all cases with different crossover methods have the smallest OF values when combined with Mutation Gaussian.



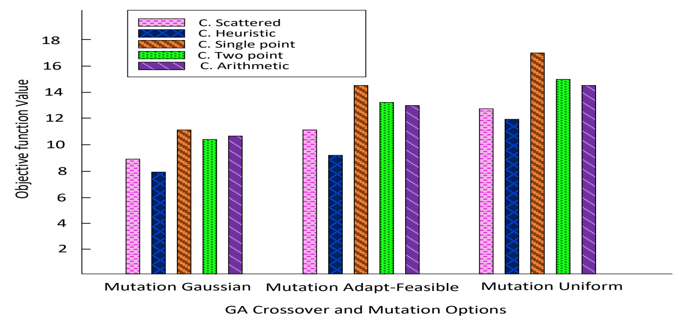


Fig. 4. a. 3-D Bar graph illustrating the objective function values averaged over 10 runs for 15 different combinations of Crossover and Mutation GA options. b. The data of graph a when categorized according to Mutation and Crossover type.

Among Crossover methods, it is seen that Crossover Heuristic achieves better results than other mutation types . Noticeable is that Mutation Uniform perform the worst. Crossover Single-point also achieves the worst result compared to other Crossover options.

C. Crossover Fraction

Another tuning parameter of GA which can influence the results is the Crossover Fraction, that is the fraction of individuals that is incorporated in the next generation through the crossover process. The study is performed on 9 different values of the Crossover Fractions varying from 0 to 1 with a step sizes of 0.1. The GA is run 10 times with a Population of 50 and Generation of 100. The Crossover and Mutation methods are taken as Crossover Heuristic and Mutation Gaussian. The results are given in Fig. 5. with the average values as bars in blue with the error bars as thin black bars. Crossover fractions of 0.5 or 0.6 give the smallest values of the objective function. The result can be understood by realizing that having an intermediate value of the Crossover Fraction allows enough diversity in the population occurs while the diversity is not too high to avoid the convergence of the GA.

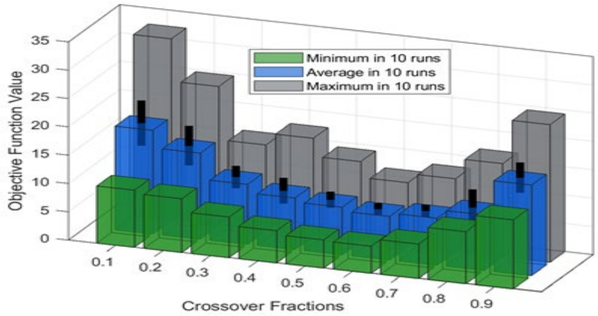


Fig. 5. 3-D Bar graph illustrating the objective function values averaged over 10 runs for 9 different Crossover Fraction values.

IV. Conclusion

Having implemented a Genetic Algorithm optimization for electrostatic electron lens design allowed us to perform a study on the influence of GA tuning parameters. The study is performed on a typical lens with six electrodes which has 23 free variables. The extension to more complex designs is straightforward. Our study illustrates that the GA has the robustness to be implemented as a global optimizer for electrostatic lens design. It also shows that there is not one optimized design because the value of the final objective function is different every time the GA is run. This implies that the fine tuning of the GA parameters is important for optimizing the performance of the GA. An analysis is performed of the impact of the values of population size and number of generations. As expected, the results improve by increasing both values. However, a population of 50 with 100 generations can provide reasonably good results. Increasing to higher values of population and generation will not significantly improve the results while the related computational time dramatically increases. It is also seen that within a constant computational effort (time or total number of system evaluations), having a lower population size than the number of generations can achieve better results than having a larger population size than the number generations. The Crossover and Mutation types as the main tuning parameters of GA are analyzed to find the most suitable options. It is found that irrespective of the type of Crossover, the Mutation Gaussian achieves the best result. Moreover, Crossover Heuristic shows the best performance among different crossover types. The combination of Crossover Heuristic with Mutation Gaussian shows significantly better results than all other combinations of Mutations and Crossovers. Crossover Fraction is also evaluated to find the most suitable values of this parameter. It is shown that a Crossover Fraction of 0.5 or 0.6 achieve the best results.

The guidelines provided here for tuning the GA parameters may be helpful not only for the optimization of electrostatic lens designs, but also for other GA optimization of functions with similar complexity.

References

[1] N. Hesam Mahmoudi Nezhad, M. Ghaffarian Niasar, A. Mohammadi Gheidari, C. W. Hagen, and P. Kruit, Int. J. Mod. Phys. A 34, 1942020 (2019).

[2] N. H. M. Nezhad, M. G. Niasar, C. W. Hagen, and P. Kruit, Proceedings of 2020 IEEE 6th International Conference on Optimization and Applications (ICOA), (IEEE, Piscataway, NJ, 2020), p. 9094475.

[3] N. Hesam Mahmoudi Nezhad, M. Ghaffarian Niasar, A. Mohammadi Gheidari, P. Kruit and C. W. Hagen, J. Vacuum Science & Technology B 39, 062605 (2021).

[4] J.P. Adriaanse, H.W.G. van der Steen, and J.E. Barth, “Practical optimization of electrostatic lenses”, J.Vac. Sci. Technol. B7, pp. 651-666, 1989.

[5] J.E. Barth, P. Kruit, “Addition of different contributions to the charged particle probe size”, Optik101, pp. 101-109, 1996.

[6] M. Sipper et al. Investigating the parameter space of evolutionary algorithms, Bio Data Mining 11:2, (2018)

[7] D. Vrajitoru, Soft Computing in Information Retrieval, F. Crestani et al. (eds.), Springer-Verlag Berlin Heidelberg 2000, Large Population or Many Generations for Genetic Algorithms? Implications in Information Retrieval, ISBN : 978-3-7908-2473-5.

[8] T. Chena, K. Tangb, G. Chenb and X. Yao, Theoretical Computer Science 436 (2012) 54–70.