Multi-electrode lens optimization using genetic algorithms

Hesam Mahmoudi Nezhad, N.; Ghaffarian Niasar, M.; Mohammadi Gheidari, A.; Hagen, C. W.; Kruit, P.

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

World Scientific
www.worldscientific.com

Multi-electrode lens optimization using genetic algorithms

N. Hesam Mahmoudi Nezhad*,‡, M. Ghaffarian Niasar†,
A. Mohammadi Gheidari*, C. W. Hagen* and P. Kruit*

*Imaging Physics, Charged Particle Optics Group,
Delft University of Technology, Lorentzweg 1,
2628 CJ Delft, The Netherlands
†Faculty of Electrical Engineering, DC Systems,
Energy Conversion and Storage Mekelweg 4,
2628 CD Delft, The Netherland
‡n.hesammahmoudinezhad@tudelft.nl

In electrostatic charged particle lens design, optimization of a multi-electrode lens with many free optimization parameters is still quite a challenge. A fully automated optimization routine is not yet available, mainly because the lens potential calculations are often done with very time-consuming methods that require meshing of the lens space. A new method is proposed that improves on the low speed of the potential calculation while keeping the high accuracy of the mesh-based calculation methods. This is done by first using a fast potential calculation based on the so-called Second-Order Electrode Method (SOEM), at the cost of losing some accuracy, and then using a Genetic Algorithm (GA) for the optimization. Then, by using the parameters of the approximate systems found from this optimization based on SOEM, an accurate GA optimization routine is performed based on potential calculation with the commercial finite element package COMSOL. A six-electrode electrostatic lens was optimized accurately within a few hours, using all lens dimensions and electrode voltages as free parameters and the focus position and maximum allowable electric fields between electrodes as constraints.

Keywords: Multi-electrode lens design; optimization; second-order electrode method (SOEM); genetic algorithms (GAs).

PACS numbers: 07.78.+s, 41.85.Ne, 41.85.Gy, 02.60.Pn, 41.20.Cv, 41.75.−i

1. Introduction

Although optimization routines are very powerful techniques for finding optima of complex functions, they have not been used extensively in the field of charged particle optics. To find the optimum design of electrostatic lenses has always been quite challenging, especially when multiple electrodes are involved. Mostly, this is done manually in a trial and error fashion. Given a certain lens geometry and voltages on the electrodes, the electrostatic field is usually calculated numerically, meshing either the entire volume of the lens (finite element method) or the electrode surfaces (charge density method). From the axial potential and its derivatives (at least up to second-order) the first-order optical properties and the aberration coefficients of the lens can then be calculated.[1] To optimize the design, an objective function is then defined which needs to be minimized by changing the lens geometry and/or electrode potentials. For each trial-design this sequence of steps, of which the field calculation is most time consuming, needs to be done. Some attempts toward automated optimization were made in the past by Szilagy *et al.*,[2,3] Adriaanse *et al.*[4] and Barth *et al.*[5–8] They avoided the time-consuming field calculation and calculated the axial potential using the Second-Order Electrode Method (SOEM).[4,5] However, at that time, computers were still slow and the results were not very accurate. Now that more powerful computers are available it is worthwhile to revisit the problem. A new approach is proposed in which a Genetic Algorithm (GA) is used as the optimization technique. The lens systems considered are rotationally symmetric electrostatic lens systems with multiple electrodes. To achieve a quick evaluation of the objective function, SOEM is used. Although this comes at the cost of losing some accuracy, it allows for rapid optimization using the GA. Using the parameters from this optimization, a more accurate GA optimization is subsequently performed, in which the objective function is calculated using the electrostatic field simulated with COMSOL[9] (based on the finite element method). As a demonstration, a six-electrode lens system was optimized for the smallest focused probe size within a few hours, with all geometries and electrode voltages as free parameters, and the focus position and maximum breakdown fields between electrodes as constraints. To extend the method to more complex systems and designs is rather straightforward. This paper is organized as follows. In Sec. 2, a brief summary of SOEM will be presented. The geometry of the example system used in this study is described in Sec. 3. The fitness function, or objective function, needed for the optimization is discussed in Sec. 4. Section 5 addresses the accuracy of SOEM compared to more accurate methods such as finite element methods. The GA is introduced in Sec. 6 and the parameters relevant in the optimization process are discussed. Results of the GA optimization using various methods to calculate the objective function are presented in Secs. 7 and 8, and conclusions are drawn in Sec. 9.

2. SOEM

In a rotationally symmetric system, the potential in space can be expressed in terms of the axial potential and its derivatives with respect to the axial coordinate z:

$$\phi(r, z) = \phi(0, z) - r^2/4\phi^{(2)}(0, z) + r^4/64\phi^{(4)}(0, z) - \cdots . \tag{1}$$

Here, $\phi(0, z)$ is the axial potential, $\phi^{(n)}(0, z)$ is the nth derivative with respect to z and r is the radial coordinate. To allow for a fast calculation of the axial potential, this equation is solved by fitting the axial potential with a cubic spline[10] (i.e. a twice differentiable function built piece wise from polynomials of second-order). This approximation requires to omit terms higher than second-order in Eq. (1), which is then rewritten for each spline interval as

$$\phi''(r, z) = 4[\phi_i(0, z) - \phi_i(r, z)]/r^2, \tag{2}$$

where i is the index of a discrete axial point. A cubic spline on an arbitrary mesh of N can be written as[4]

$$\phi(1/\Delta_i + 1/\Delta_{i-1}) = \phi_{i-1}/\Delta_{i-1} + \phi_{i+1}/\Delta_i - (\phi''_{i-1}\Delta_i + 2\phi''i(\Delta_i + \Delta_{i-1})$$
$$+ \phi''_{i+1}\Delta_i)/6, \quad i = 2, \ldots, N - 1. \tag{3}$$

Here, Δ_i is the interval between point i and $i + 1$. In our lens model, the points placed inside each electrode are assumed to be equidistant and the gaps between electrodes are taken as one single spline interval. Equation (2) can be substituted in Eq. (3) to get

$$\phi_{i-1}\left(\frac{2\Delta_i}{3r_{i-1}^2} - \frac{1}{\Delta_{i-1}}\right) + \phi_i\left(\frac{1}{\Delta_{i-1}} + \frac{1}{\Delta_i} + \frac{4}{3}\frac{\Delta_{i-1} + \Delta_i}{r_i^2}\right) + \phi_{i+1}\left(\frac{2\Delta_i}{3r_{i+1}^2} - \frac{1}{\Delta_i}\right)$$
$$= \frac{2V_{i-1}}{3r_{i-1}^2}\Delta_{i-1} + \frac{4V_i}{3r_i^2}(\Delta_{i-1} + \Delta_i) + \frac{2}{3}\frac{V_{i+1}}{r_{i+1}^2}\Delta_{i+1}, \quad i = 2, \ldots, N - 1, \tag{4}$$

where r_i is the radius of the electrode at position i and V_i is the electrode potential at point i.

In Eq. (4), renaming the coefficients of ϕ_{i-1}, ϕ_i and ϕ_{i+1} by a_i, b_i and c_i, respectively, and the right-hand side of the equation by d_i, it follows that

$$a_i = \frac{2\Delta_i}{3r_{i-1}^2} - \frac{1}{\Delta_{i-1}}, \quad b_i = \frac{1}{\Delta_{i-1}} + \frac{1}{\Delta_i} + \frac{4}{3}\frac{\Delta_{i-1} + \Delta_i}{r_i^2},$$

$$c_i = \frac{2\Delta_i}{3r_{i+1}^2} - \frac{1}{\Delta_i}, \quad d_i = \frac{2V_{i-1}}{3r_{i-1}^2}\Delta_{i-1} + \frac{4V_i}{3r_i^2}(\Delta_{i-1} + \Delta_i) + \frac{2V_{i+1}}{3r_i^2 + 1}\Delta_{i+1}, \tag{5}$$

$$a_i\phi_{i-1} + b_i\phi_i + c_i\phi_{i+1} = d_i.$$

Equation (5) correlates the axial potential at each point to its value at the neighboring points, and can be represented as a tri-diagonal matrix

$$
\begin{bmatrix}
b_1 & c_1 & 0 & 0 & 0 & 0 & 0 \\
a_2 & b_2 & c_2 & 0 & 0 & 0 & 0 \\
0 & a_3 & b_3 & c_3 & 0 & 0 & 0 \\
0 & 0 & \cdot & \cdot & \cdot & 0 & 0 \\
0 & 0 & 0 & \cdot & \cdot & \cdot & 0 \\
0 & 0 & 0 & 0 & a_{N-1} & b_{N-1} & c_{N-1} \\
0 & 0 & 0 & 0 & 0 & a_N & b_N
\end{bmatrix}
\begin{bmatrix}
\phi_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \phi_N
\end{bmatrix}
=
\begin{bmatrix}
d_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ d_N
\end{bmatrix}
\tag{6}
$$

where the matrix elements (a_i, b_i, c_i and d_i, $i = 2 : N - 1$) are known parameters related to the geometry of the lenses and electrode voltages. Applying boundary conditions (zero field and derivative at the first and last electrode), the tridiagonal matrix can be solved very fast by numerical methods (for which MATLAB[11] is used in this work) to derive the axial potential at all points i. Accordingly, first and second derivatives can also be numerically calculated.

3. Lens Geometry

The example lens system considered here is a six-electrode rotationally symmetric electrostatic lens as schematically shown in Fig. 1. The object is assumed to be far left from the lens system (10 times the length of the system) and the image plane is fixed at $z = 3.5$ mm from the entrance of the lens. The initial electron energy is eV_1, where V_1 is the voltage of the first electrode and the landing energy of the electrons in the image plane is fixed at 1 keV.

The lens parameters such as thicknesses of the electrodes, radii of the holes, gaps between electrodes and the voltages of each electrode, are the variable parameters of the optimization problem. The ranges of these free parameter are from 0.05 to 0.3 mm, 0.01 to 0.3 mm and 0.1 to 0.6 mm, for thicknesses, radii and gaps, respectively. The voltages at each electrode (V_i) are free to change in the range of 500 V–5 kV. However, there are some constraints. First of all, the maximum field between two sequential electrodes has to be less than 10 kV/mm to avoid voltage
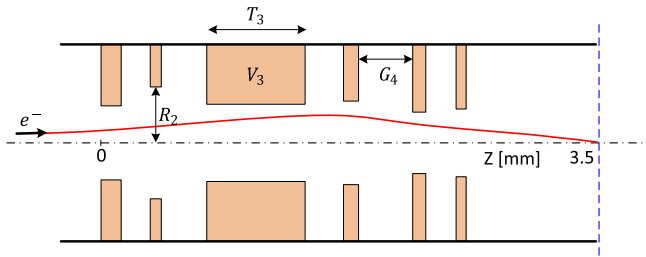

Fig. 1.   Schematic of a six-electrode lens (T_i: thicknesses, R_i: radii, G_i: gaps between electrodes, V_i: voltages of the electrodes).

breakdown, i.e.

$$(V_{i+1} - V_i)/(T_{i+1} - T_i) < 10 \text{ kV/mm} \quad \text{(no tolerance)}, \tag{7}$$

where V_i and T_i are the voltage and the thickness of the ith electrode, respectively. The second constraint is that the image plane is fixed at a position X_c

$$X_c = \text{const.} \quad \text{(with 2\% tolerance)}. \tag{8}$$

4. The Objective Function

In a rotationally symmetric probe-forming electron-optical system, the image of a point source is a blurred spot due to geometrical and chromatic aberrations of the imaging system. Geometrical aberrations strongly depend on the geometry of the lens as well as the voltages on the electrodes. The optimization routine can be used to find the optimum lens design by tuning the geometry of the lens and/or the voltages of the electrode to minimize the aberration contributions, thereby obtaining the smallest spot size. Since the aim is to minimize the spot size at the image side, it is taken as the objective function (or the so-called "fitness function" in the context of GA).

Assuming that the lens only suffers from axial spherical and chromatic aberrations, the spot size can be expressed as[12]

$$D_s{}^2 = (0.18 C_s \alpha^3)^2 + \left(0.6 C_c \alpha \frac{\Delta U}{U}\right)^2, \tag{9}$$

where, if D_s is calculated at the image side (as is the case here), all parameters in the equation are taken at the image side. The first term is the contribution from spherical aberration, with C_s the spherical aberration coefficient of the lens, the second term is the contribution from chromatic aberration, with C_c the chromatic aberration coefficient of the lens, α is the half opening angle of the beam (chosen to be 10 mrad in this case-study), ΔU is the energy spread of the electron source (taken as 1 eV) and U is the acceleration energy (equal to the potential at the image plane = 1 kV). Note that the source image contribution is neglected in Eq. (9).

The aberration coefficients are first calculated at the object side. Then using the magnification, they are converted to the image side. The spherical and chromatic aberration coefficients can be calculated from the axial potential, its derivatives and a principle imaging ray $r_\alpha(z)$, starting on the optical axis in the object with angle 1, using the following integrals:[13]

$$C_s = \frac{1}{16\phi_0^{\frac{1}{2}}} \int_{z_o}^{z_i} \phi^{1/2} \left(\frac{5}{4}\left(\frac{\phi''}{\phi}\right)^2 + \frac{5}{24}\left(\frac{\phi'}{\phi}\right)^4 r_\alpha^4\right.$$
$$\left. + \frac{14}{3}\left(\frac{\phi'}{\phi}\right)^3 r_\alpha^3 r_\alpha' - \frac{3}{2}\left(\frac{\phi'}{\phi}\right)^2 r_\alpha^2 r_\alpha'{}^2\right) dz, \tag{10}$$

$$C_c = \phi_0^{\frac{1}{2}} \int_{z_o}^{z_i} \left(\frac{3}{8}\right) \frac{\phi'^2}{\phi^{\frac{5}{2}}} r_\alpha^2 dz, \tag{11}$$

whereas z_o and z_i denote the z-axis positions of the object and image, respectively. C_s and C_c are calculated at the object side, and ϕ_0 refers to the potential at the object side. The principle imaging ray $r_\alpha(z)$ is obtained by ray tracing. The relativistic equation of motion, in the paraxial approximation, takes the following form[13]:

$$\frac{1+\epsilon}{1+2\epsilon}\phi(z) \cdot r''(z) + \frac{1}{2}\phi'(z) \cdot r'(z) + \frac{1}{4}\phi''(z) \cdot r(z) = 0, \tag{12}$$

where $\epsilon(z) = e/(2m_0c^2)$ in which e, m_0 and c are the elementary charge, electron rest mass and velocity of light, respectively. The equation is solved numerically.

The magnification M is obtained by using another principle ray r_b, starting in the object plane at a height h and with angle 0, and calculating the height in the image plane, as determined by the axis crossing of r_a at the image side, and dividing by h. C_s and C_c at the image side are then obtained by

$$C_{\mathrm{si}} = M^4 \cdot \left(\frac{V_2}{V_1}\right)^{(3/2)} \cdot C_{\mathrm{so}}, \tag{13}$$

$$C_{\mathrm{ci}} = M^2 \cdot \left(\frac{V_2}{V_1}\right)^{(3/2)} \cdot C_{\mathrm{co}}. \tag{14}$$

Inserting C_{si} and C_{ci} into Eq. (9) the spot size D_s at the image side is calculated. When, in the remainder of the paper, the spot size is mentioned, the spot size in the image plane is meant.

5. Accuracy of SOEM

EOD[14] is a professional software package commonly used by electron optical lens designers, while COMSOL[9] is a multi-physics finite element-based solver software which is not very commonly used so far by electron optical designers. We wanted to use a well-known and reliable electron optical software package as the main reference (benchmark) for comparison. But we also wanted to make use of a program which can be integrated in MATLAB to accurately calculate the potential in an automated way, and could be used together with our other codes (ray-tracing and optimization). As EOD could not easily be automated and integrated into MATLAB, we have done that part using COMSOL. Although it is expected that COMSOL also produces accurate results, for more certainty, EOD is also taken as a benchmark initially to compare to results obtained from COMSOL. It has been shown that COMSOL gives results as accurate as EOD (Fig. 2), COMSOL then has been used as an accurate software program to calculate the potential, and it has been used as a reference for the rest of the study.

To evaluate the accuracy of the potential and its derivatives as calculated using SOEM, a comparison was made to results obtained with EOD, as well as with the finite element package COMSOL. The calculation was done for the lens system of Fig. 1. The axial potentials and its first and second derivatives, as calculated using SOEM, COMSOL and EOD for two different designs of the system of six-electrodes
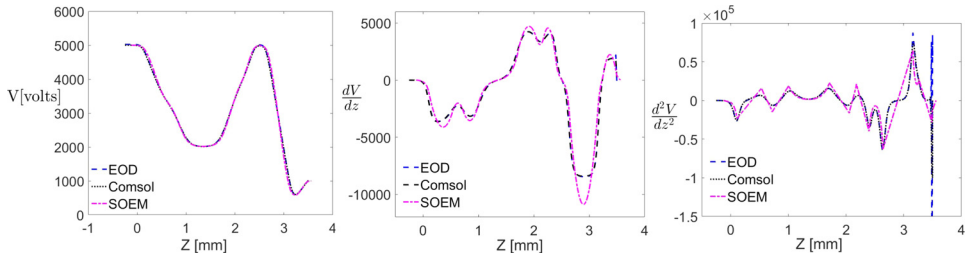
Fig. 2. Comparison of the axial potential and its first and second derivatives, calculated by SOEM, COMSOL and EOD, for a six-electrode lens of Fig. 1 ($T_i = 0.11, 0.18, 0.70, 0.12, 0.24, 0.10$] mm, $R_i = [0.10, 0.23, 0.21, 0.21, 0.09, 0.08]$ mm, $Gi = [0.43, 0.29, 0.35, 0.22, 0.52]$ mm, $V_i = [5000, 3300, 2000, 3850, 5000, 550]$ V. (Results: C_s COMSOL = 1.6 mm, C_c COMSOL = 0.59 mm, C_s SOEM = 1.3 mm, C_c SOEM = 0.52 mm; all values taken at the image side).
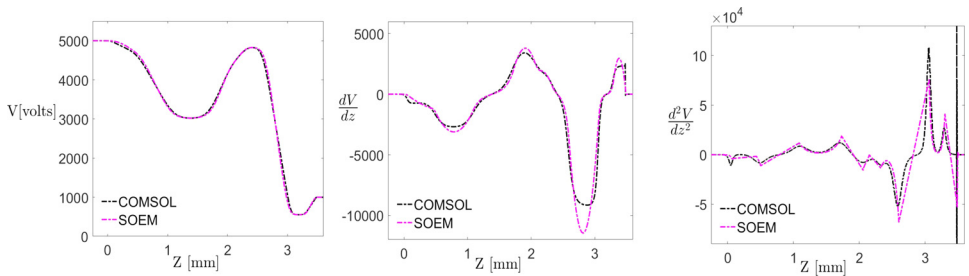

Fig. 3. Comparison of the axial potential and its first and second derivatives, calculated by SOEM and COMSOL, for a six-electrode lens of Fig. 1 ($Ti = 0.06, 0.06, 0.64, 0.09, 0.27, 0.24]$ mm, $Ri = [0.05, 0.19, 0.23, 0.24, 0.12, 0.06]$ mm, $G_i = [0.38, 0.59, 0.33, 0.18, 0.46]$ mm, $V_i = [5000, 4750, 3000, 4550, 4850, 550]$ V) (Results: C_s COMSOL = 3.13 mm, C_c COMSOL = 0.69, C_s SOEM = 9.88 mm, C_c SOEM = 1.41 mm; all values taken at the image side).

(Fig. 1), are shown in Figs. 2 and 3. Another example of a three-electrode lens system can be found in Ref. 15.

The results calculated with COMSOL and EOD agree very well with each other, even for the second derivatives. But the potential calculated by SOEM is slightly different, which becomes more visible in the derivatives. At first sight, the difference between the derivatives calculated by SOEM and COMSOL in both figures looks pretty much the same. But when calculating the aberration coefficients with both methods, almost the same C_s and C_c are found for the system of Fig. 2 but for the system of Fig. 3 COMSOL results in a three times lower C_s and a two times lower C_c. This means that the objective function is a quite sensitive function of the potential, and the approximate potential as determined by SOEM may lead to deviations of the objective function. Irrespective of that, the higher computation speed of SOEM, compared to the commercial programs, still makes it attractive to use in an optimization routine.

As many parameters are involved in optimizing the lens system considered here, it is expected that the objective function landscape may have many local minima.

Therefore, a global optimizer is needed. Due to its random nature, the GA is a good candidate to find such a global solution.[16,17] In Sec. 6 GA will be briefly introduced.

6. Genetic Algorithm Optimization Using SOEM

The GA is a heuristic search technique with a logic which mimics the natural evolution process.[18] The GA optimization procedure starts with a set of systems (each system representing a "chromosome" in the context of GA) called the initial population. The initial population can be a randomly generated state denoted by $P_1(x_1, \ldots, x_n)$, where the x_i are vectors representing the chromosomes of the individual systems i. For the lens system of Fig. 1, the initial population is the set of lens systems with each system characterized by a vector containing its design parameters, i.e. a vector of 23 parameters including six electrode thicknesses (T_i) (the third thickness, as a choice, is kept dependent to the other thicknesses and gaps, to keep the length of system constant), 6 radii (R_i), 5 gaps (G_i) and 6 electrode voltages (V_i).

This initial population gradually evolves toward a set of chromosomes (set of lens parameters) with better properties, to be judged by evaluating the objective function, a function of the chromosomes (lens design parameters). The evolution leads to a new generation, a new population of systems $P_{i+1}(x_1, \ldots, x_n)$ closer to the optimum of the objective function. The creation of a new population, in a so-called Simple Genetic Algorithm (SGA), is mainly done using operators such as selection, crossover and mutation. Systems are selected based on the value of the objective function. The crossover and mutation operators serve as a mechanism to produce the population for the next generation. Crossover is the operator which combines two individuals (parents) to create a new individual (offspring). The idea behind using this operator is to create new individuals with better characteristics than their parents by inheriting the best characteristics from each parent. Crossover is applied on the population during evolution based on a user definable crossover probability. Mutation is a procedure applied in the algorithm, analogous to biological mutation, to give genetic diversity by altering genes in a chromosome of a population from one generation to the next generation. This procedure is applied in the algorithm to give each individual a better opportunity to reach the full space of possible solutions by enhancing the diversity of the population. Elitism is an additional operator, which could be added to the process to enhance the result of the GA. This operator copies a certain number (i.e. the elite count) of best parents, unchanged, into the next generation. This prevents the GA from wasting time in rediscovering these best parents which could otherwise have been discarded while breeding new populations. A schematic of a simple GA is shown in Fig. 4.

SOEM is used in the objective function evaluation part of the optimization. Generation is the number of iterations the GA algorithm executes before it terminates. This parameter can be set as the stopping criterion of the GA, as was
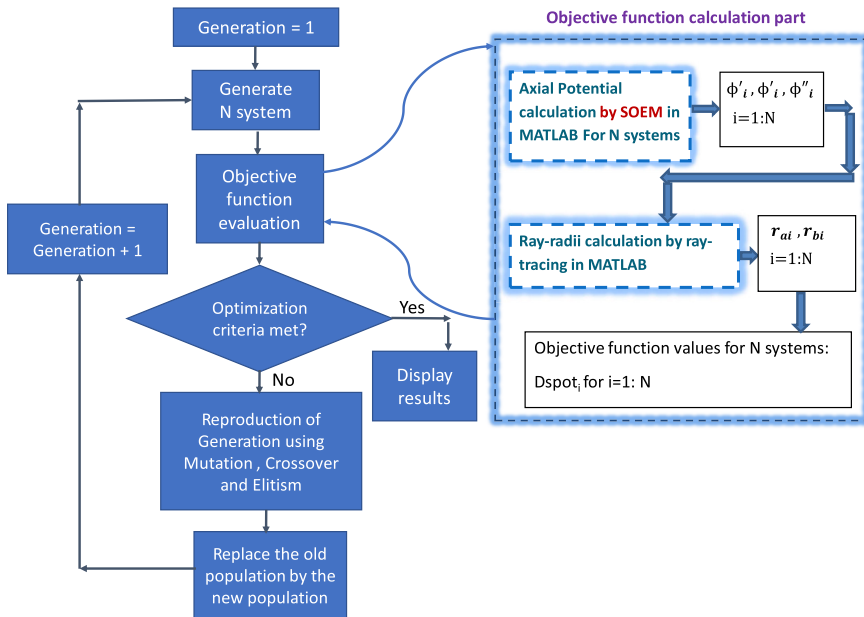
Fig. 4. A schematic of the GA, with SOEM used for the objective function evaluation.

done in this study. The population size is the number of individuals present in each generation.

The MATLAB GA module with constraints was used for the optimization. The potential calculation using SOEM and the ray-tracing codes to evaluate the objective function were coded in MATLAB as well. In Sec. 6.1 the choice of GA optimization parameters is discussed.

6.1. *GA parameters*

The GA has a variety of parameters such as initial population, population size, generation, crossover, mutation and elitism (elite count). These parameters should be tuned well based on the exact application, to get the best performance.

Increasing the population size, the GA will search the space more extensively. This increases the probability of reaching the global solution and reduces the chance of getting trapped in local optima. However, increasing the population size to large values will also significantly increase the execution time of the GA. Thus, it is a tradeoff between run time and the best result. As a good compromise a population size of 50 was chosen in this study.

The MATLAB GA routine contains different crossover and mutation types. For optimization problems with constraints, as is the case here, most of the available crossover and mutation types are not applicable. Among the ones applicable, the types Intermediate and Adapt-feasible for crossover and mutation, respectively, were found to be the most efficient ones.
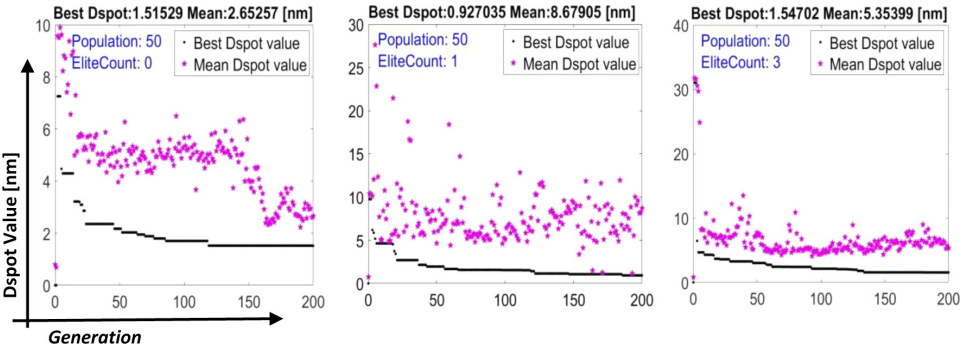
Fig. 5. Three different GA runs to assess the generation number and elite count. The elite count is indicated in the graphs, and the lowest Dspot value achieved in each graph is indicated above the graphs.

The effects of increasing the generation number and elite count on the performance of the GA was also assessed. As the generation number has a direct effect on the execution time of the algorithm, a tradeoff between result and execution time has to be made. Figure 5 shows a few examples of these assessments. The plots show the value of the objective function, or the spot size D_{spot}, as a function of the generation number, for three different values of the elite count: 0, 1 and 3. For each generation number the mean value of D_{spot} is plotted (purple pentagram, light pentagram in gray scale) as well as the smallest, or best, value of D_{spot} (black dots, dark dots in gray scale). It is seen, in all plots, that increasing the number of generations to more than 200 did not significantly lower the best value of D_{spot}, while the execution time would continue to increase. Therefore, the number of generations was limited to 200 in this study. Analyzing the effect of the elite count parameter on the result of the GA, it is observed that a value of one (the middle graph in Fig. 5) led to the smallest value of D_{spot} ($D_{\mathrm{spot}} = 0.927$). So, a value of one was chosen in this work.

Summarizing, the MATLAB GA parameters chosen for this study were: generation number 200, population number 50, elite count 1, selection type stochastic uniform, crossover type Intermediate, and mutation type Adapt-feasible.

7. Results of GA Optimization with SOEM

A GA run starts with the first generation of 50 systems, randomly created within the parameter ranges defined in Sec. 3, and then evolves for 200 generations. In a run 10000 systems in total are evaluated, of which around 1500 systems (15%) fulfilled the constraints and went through the selection, based on the value of the objective function as determined by SOEM. The spot sizes D_{spot} obtained for these 1500 systems are sorted in descending order and plotted in that order in Fig. 6 (the black ∗ symbols). Note that the horizontal axis represents the position of the systems in the sorted array according to their spot sizes and not the systems in
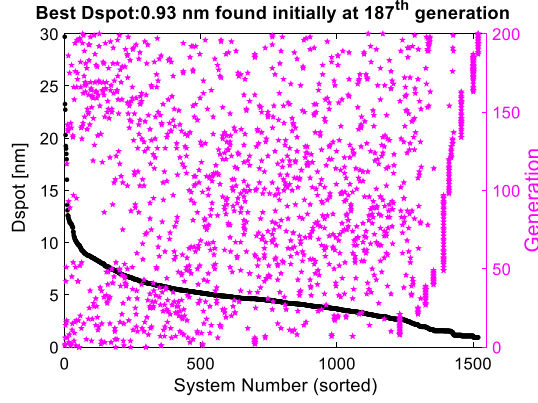
Best Dspot:0.93 nm found initially at 187th generation

Fig. 6. The spot size (D_{spot}, left-hand axis), taken from one optimization run using SOEM, sorted in decreasing order (purple (light in gray scale) pentagrams), and the generation in which the spot size occurred (black (dark in gray scale) dots, right-hand axis).

which the plotted spot sizes were found. The plot also contains information on the generation in which each spot size value was obtained. Corresponding to each point on the black curve, there is a purple pentagram (or pentagrams) in the graph. The pentagrams show the generation in which the system index has occurred (the pentagram lies on a line that passes through that black dot and is perpendicular to x-axis). It is seen that larger spot sizes can be found in all generations, basically but the smallest spot sizes (on the right-hand side of the graph) occur predominantly in higher generations. The smallest spot size of 0.93 nm was found first in the 187th generation.

In Sec. 5, it was seen that SOEM only provides an approximate potential, compared to the potential obtained from COMSOL, and it was seen for two different systems that the values of C_s and C_c are quite sensitive to the actual potential. Therefore, it was investigated how far off the SOEM optimization results are from COMSOL results.

The values of C_s, C_c, X_c (the image plane position), and D_{spot} (spot size) were obtained by an optimization with SOEM, and were then compared to the values as calculated using COMSOL for the systems found. As, by nature of the GA, the resulting data of a run, are jumping up and down to reach to a final good result (as is seen in Fig. 6), the data are again sorted by their objective function value in descending order, to more clearly see the trends. Because it is time consuming to do the field calculation with COMSOL, not all 1500 systems were calculated. Of 980 systems only one out of 20 systems were taken, resulting in still a representative number of 49 systems to compare. In Fig. 7, the values of C_s, C_c, X_c and D_{spot}, based on the SOEM optimization, are plotted in descending order (the black star symbols). For each system indicated with a black star the same parameters were then inserted in the field calculation with COMSOL. The resulting values of C_s, C_c, X_c and D_{spot} are shown as red stars in the plots.
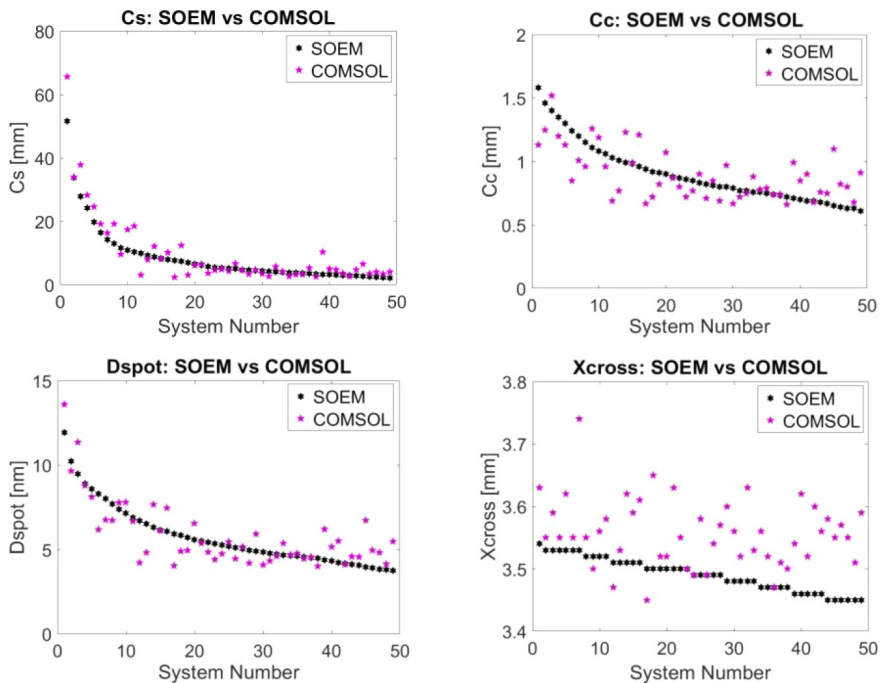
Fig. 7.   The sorted data of spherical aberration (C_s), chromatic aberration (C_c), spot size (D_{spot}) and X_c (image plane position) for 49 systems, taken from an optimization run based on SOEM and the comparison with accurate values calculated using COMSOL.

It is seen in all plots that the SOEM-based data and the COMSOL-based data show the same trend. The SOEM-based values are mostly close to the more realistic COMSOL-based ones, only in some cases deviations up factors of 2–3 can be seen, as was already seen for the two systems shown in Sec. 5. Please note that the smallest spot size in Fig. 7 is 3.8 nm, and not 0.927 as in Fig. 6, because only 980 systems are considered instead of 1500. A very critical parameter is the value of X_c, which is one of the constraints. In Fig. 7, it is seen that the SOEM-based values of X_c are often lower than the COMSOL-based ones, by up to 5%. Such deviations, however, are still considered to be acceptably small. As such, the SOEM-based optimization offers a very good approximate parameter estimation. But because of the observed deviations, SOEM-based GA is not advisable when accurate results are desired. Therefore, a more accurate method is needed, which is proposed in Sec. 8.

8. GA Optimization with SOEM and COMSOL

To improve the optimization accuracy, the possibility to integrate COMSOL in the optimization routine was investigated. This provides a more accurate calculation of the potential and the objective function but the optimization will become more time consuming. As a compromise, a first rough optimization with SOEM is performed
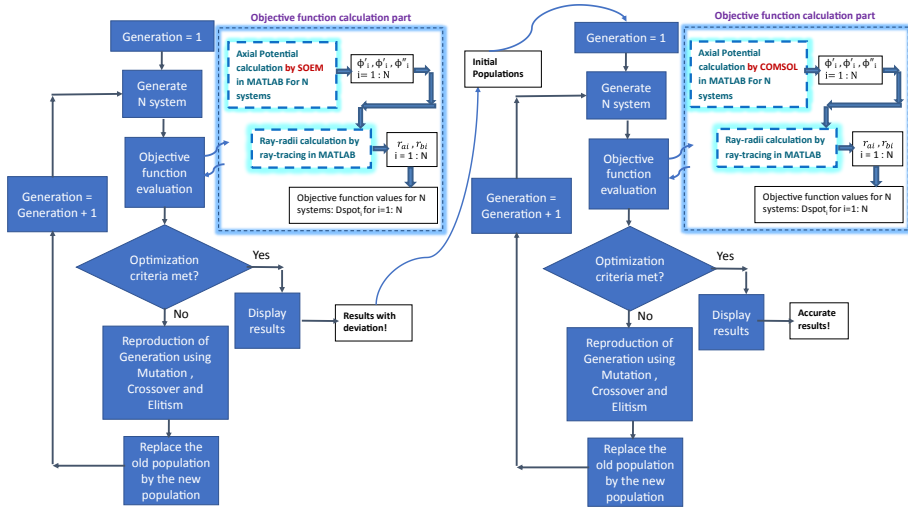
Fig. 8.   Schematic of the flowchart of the GA based on a combination of SOEM and COMSOL.

(GA-SOEM), resulting in a few approximately optimized systems. Then, using the parameters of the systems found, a new set of systems is defined and fed to the GA among the population of the first generation. Then a more accurate but somewhat slower, COMSOL-based optimization is done, starting from these almost good systems (GA-SOEM+COMSOL). A schematic of this optimization flowchart is shown in Fig. 8. This method prevents the GA from running for many more generations before reaching a good system. This method has been tested and succeeded in very accurately optimizing the example system of Sec. 3, within a few hours.

In Fig. 9(a), the resulting spot size as obtained from the GA-SOEM, with 50 populations and 200 generations, is shown as a function of the generation number. Again, as in Fig. 5, the minimum spot size and the mean spot size are plotted. This optimization run took about 30 min and the best spot size was 0.927 nm. For this best system the potential and its derivatives were subsequently calculated with COMSOL, resulting in a more accurate spot size for this system of 4 nm and a slightly different location of the focal plane.

In the next step, the 20 best systems found by GA-SOEM are fed as initial population into GA-SOEM+COMSOL. MATLAB GA automatically creates the additional 10 random systems to create a total of 30 initial populations. Afterward, GA-SOEM+COMSOL is run for only 20 generations to reach the best solution of 3.317 nm (see Fig. 9(b)), i.e. slightly smaller than the 4 nm. This took about 1 h. To continue for more generations did not lead to further improvement. Thus, running GA-SOEM followed by GA-SOEM+COMSOL resulted in a good and accurate system in about 1 h. To assess the speed of GA-SOEM+COMSOL, this method is compared with the situation where the GA is only based on COM-SOL (GA-COMSOL). In Fig. 9(c), the result of GA-COMSOL is presented, where
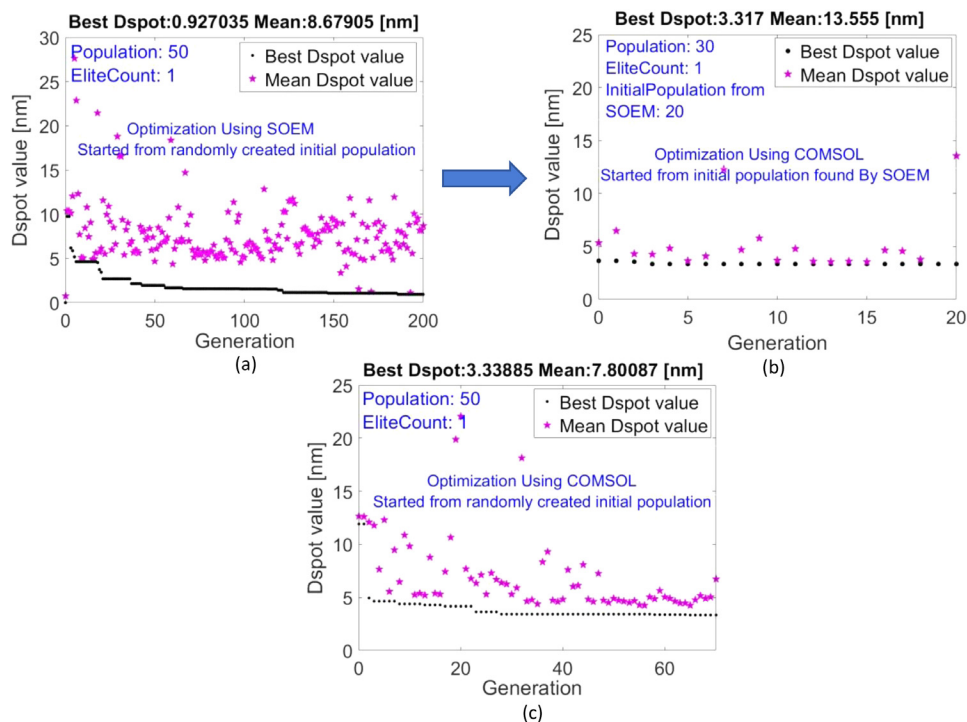
Fig. 9. (a) optimization using GA-SOEM, (b) optimization using GA-SOEM+COMSOL, starting from 20 initial populations resulting from GA-SOEM (plot a) and (c) optimization only using GA-COMSOL.

the GA started from randomly created systems and ran for 70 generations, with a population of 50. The execution time was about 5.5 h. The smallest spot size at the first generation was about 12 nm. As is shown, after 70 generations, it still has not arrived at the best solution obtained from GA-SOEM+COMSOL. It is thus concluded that it is worthwhile to first use GA-SOEM, although its results sometimes deviate a bit from accurate results obtained from GA-COMSOL, as a rough optimization and as preoptimized input for a short COMSOL-based GA optimization. A considerable time saving is achieved this way.

As a final example, to show the usefulness of an optimization routine, a manual design of a five-electrode lens system is shown in Fig. 10(a). The parameters to vary were the electrode diameters, their thicknesses and voltages. The values of C_s, C_c and D_{spot} obtained after many many hours of trial and error simulations, are shown in Fig. 10(a). For comparison, using the GA-SOEM+COMSOL method, a six-electrode system could be found with a smaller spot size, C_s and C_c (see Fig. 10(b)) in a reasonably fast time (̃few hours). A precise time is not given for the traditional methods (manual optimization by the designers), as it is very dependent on the expertise and knowledge of the designer to manually reach a satisfactory
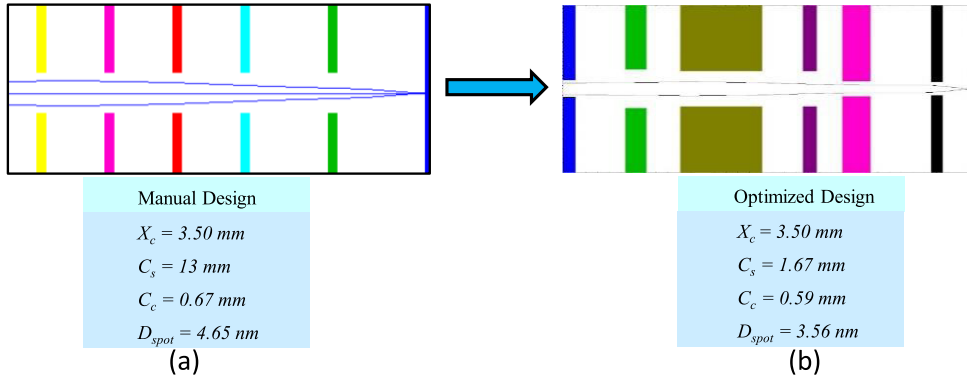
Fig. 10. An example of a manually optimized five-electrode system (a), compared to a six-electrode system optimized using the GA-SOEM+COMSOL method (b).

result. However, the consumed time would be far longer than the presented fully-automated optimization technique presented in this paper, which basically tests thousands of systems and reachs an optimized system in a few hours. It should also be mentioned that it is not only a matter of time comparison. To manually reach a satisfactory result for complex systems with all parameters as variables, with many objective functions to be minimized simultaneously, is a very cumbersome task and is not guaranteed in a reasonable time. This is indeed the advantage of having an automated optimization technique to ensure getting a satisfactory result within a short amount of time.

9. Conclusion

Fully automated optimization routines can relieve the laborious manual design of charged particle optics components. However, such routines suffer from low computational speed. The main reason for the low computation speed is the time-consuming process of the lens potential calculation, which is needed to calculate the objective function. This is often done with very time-consuming methods based on a meshing of the lens space. A new method is proposed that improves on the low speed of the potential calculation while keeping the high accuracy of the mesh-based calculation methods. In this method, the potential of the system is calculated using a fast potential calculation based on the so-called SOEM. A set of coarse optima for the design is then selected using a GA. Then by using the parameters of the approximate systems found from this optimization based on SOEM, an accurate GA optimization routine is performed based on a potential calculation with the commercial finite element package COMSOL. A six-electrode electrostatic lens was optimized accurately within a few hours, using all lens dimensions and electrode voltages as free parameters and the focus position and maximum allowable electric fields between electrodes as constraints.

References

1. V. Zworykin, *Electron Optics and the Electron Microscope* (J. Wiley & Sons, 1945).
2. M. Szilagyi and J. Szep, *J. Vacuum Sci. Technol. B*: *Microelectron. Process. Phenomena* **6**, 953 (1988).
3. M. N. Szilagyi, *Proc. IEEE* **73**, 412 (1985).
4. J. Adriaanse, H. van der Steen and J. Barth, *J. Vacuum Sci. Technol. B*: *Microelectron. Process. Phenomena* **7**, 651 (1989).
5. H. van der Steen, J. Barth and J. Adriaanse, *Nucl. Instrument. Methods Phys. Res. Sec. A*: *Accelerators, Spectrometers, Detectors and Associated Equipment* **298**, 377 (1990).
6. H. van der Steen and J. Barth, *J. Vacuum Sci. Technol. B*: *Microelectron. Process. Phenomena* **7**, 1886 (1989).
7. M. A. J. van der Stam, J. E. Barth and P. Kruit, Design of a multimode transport lens with optimization program SOEM, in *Charged-Particle Optics* (1993), pp. 45–57.
8. J. E. Barth, H. van der Steen and J. Chmelik, *Proc. SPIE* **2522**, 128 (1995).
9. Comsol multiphysics, version 5.3a (2017), https://www.comsol.com/.
10. G. D. Knott, *Interpolating Cubic Splines* (Birkhäuser, Boston, 1999).
11. Matlab, version r2016 b, https://www.mathworks.com/products/matlab.html.
12. J. Barth and P. Kruit, *Optik* **3**, 101 (1996).
13. M. Szilagyi, *Electron and Ion Optics* (Springer Science & Business Media, 1988).
14. Eod, version 4.008 (1999-2015 spoc), http://www.lencova.com/index.php/about-eod.
15. N. Hesam Mahmoudi Nezhad, M. Ghaffarian Niasar, A. Mohammadi Gheidari, T. Janssen, C. Hagen and P. Kruit, Optimization of electrostatic lens systems using genetic algorithms, in *Recent Trends in Charged Particle Optics and Surface Physics Instrumentation* (2018), pp. 24–26.
16. K. Höschel and V. Lakshminarayanan, *J. Opt.* 1 (2018).
17. N. Hesam Mahmoudi Nezhad, Optical system optimization using genetic algorithms, Master's thesis, Delft University of Technology (2014).
18. M. Mitchell, *An Introduction to Genetic Algorithms* (MIT Press, Cambridge, London, 1998).