

APPLICATION OF A MICRO-GENETIC ALGORITHM IN OPTIMAL DESIGN OF A DIFFRACTIVE OPTICAL ELEMENT *

Svetlana Rudnaya

Avant! Corporation,

48671 Bayside Parkway, Fremont, CA 94598, USA

Fadil Santosa

School of Mathematics,

University of Minnesota, Vincent Hall,

206 Church St SE, Minneapolis, MN 55455, USA

santosa@math.umn.edu

Abstract This study is motivated by a need to design a diffractive optical element arising in an application. Under realistic manufacturing constraints, it can be shown that the design problem is an optimization calculation with integer variables. We consider an optimization strategy based on Genetic Algorithms. We show that for a particular variant, called a Micro-Genetic Algorithm, the algorithm converges in a probabilistic sense to the global optimum. We demonstrate the use of the algorithm in the design of a diffractive optical element.

Keywords: Diffractive Optics, Genetic Algorithms, Optimization

1. INTRODUCTION

We study the problem of creating a desired light intensity pattern on a screen. A light source is given. What is required is a thin-film diffractive optical element, a lens, that alters the incoming light to produce the desired pattern. A sketch of the setup is shown in Figure 1. Under the thin-lens approximation, the desired unknown is a thickness distribution over the film. This is a form of inverse problem.

*The authors acknowledge support from the 3M Corporation, and from AFOSR through a MURI grant to the University of Delaware.

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: [10.1007/978-0-387-35514-6_15](https://doi.org/10.1007/978-0-387-35514-6_15)

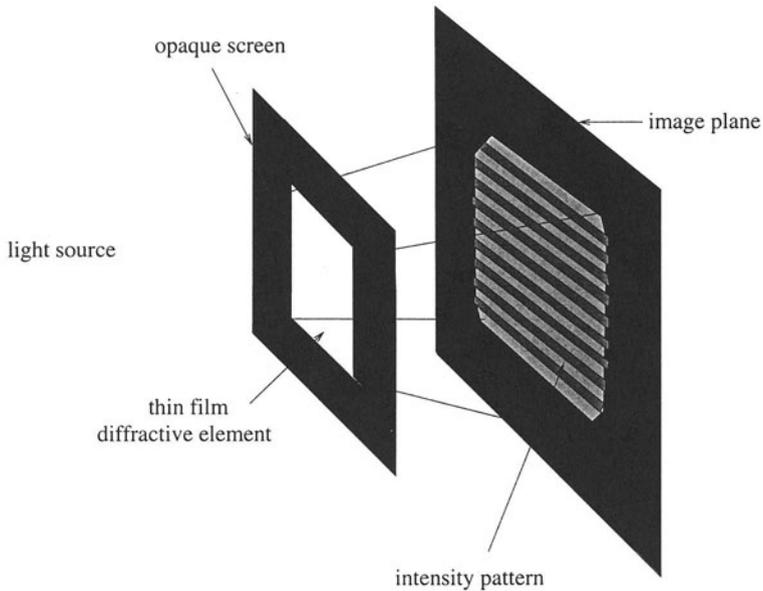


Figure 1 A sketch of the optimal design problem. Light enters the aperture where a thin film element has been placed. The incoming light is altered as it exits the film. As the light strikes the screen, it produces an intensity pattern on the screen. The inverse problem is to design a diffractive optical element that produces a desired intensity pattern.

The so-called ‘direct problem’ involves finding the intensity pattern given a light source, a diffractive optical element, and the geometry of the problem. We will show that this process is akin to function evaluation, and, under the Kirchoff approximation, amounts to a quadrature.

The inverse problem then is to find a diffractive optical element which produces an intensity pattern that is as close as possible to the desired pattern. Such a problem is very naturally cast as an optimization calculation. What is somewhat unusual about this work is that we are, in addition, given information about the manufacturing process for the diffractive optical element. The process poses additional constraints, which can be modeled as integer variables with simple bounds. However, the size of the problem, usually involving over 20000 variables in practical applications, makes it extremely difficult to solve using standard optimization methods.

We propose the use of a variant of the Genetic Algorithm [4, 7, 12]. Genetic Algorithms have found success in several areas of optimal design in optics [3, 9], as well as in other areas. We opted to use the Micro-Genetic Algorithm [11]. We study its convergence properties us-

ing Markov Chain analysis [15, 16]. The use of the algorithm applied to the optical problem is demonstrated in a numerical example. We end the paper with a discussion of some of the issues that occur in the optimization calculation.

2. OPTIMAL DESIGN OF DIFFRACTIVE OPTICAL ELEMENTS

Consider the 2-dimensional problem that is indicated in Figure 2. A lens has been placed in an opening of width $2w$ in an opaque screen. An incident light propagates from the left and is diffracted by the lens-screen setup. The diffracted light is gathered on the image plane $z = d$.

Under the scalar model of light, the scalar field, which could represent the strength of the electric or magnetic field, satisfies the Helmholtz equation. Specifically, the scalar field $u(x, z)$ has the property

$$\Delta u + k^2 n(x, z)^2 u = 0,$$

where $n(x, z)$, the index of refraction, is normalized to 1 in the air, and is equal to n_0 in the lens. The variable k represents the normalized wavelength in air. To complete the description of the diffraction problem, we would specify boundary conditions on the opaque screen along $\{z = 0, |x| > w\}$, and the Sommerfeld radiation condition for $x^2 + z^2 \rightarrow \infty$. By specifying an incident wave, in this case the wave generated by the source, one can determine the scattered wave by solving the partial differential equation with the boundary conditions. Then, after calculating $u(x, z)$, one can easily find the intensity at the image plane $z = d$, because it has the value $|u(x, d)|^2$.

When the features in the diffractive optical element are many times larger than the wavelength of light, we can approximate the field diffracted by the lens using Kirchhoff approximation [2, 5] (see also [14]). Under this approximation, the field at a point (x_0, z_0) is related to its value at the aperture by

$$u(x_0, z_0) = \frac{ik}{2} \int_{-w}^w u(x, 0) \frac{z_0}{r} H_1^{(1)}(kr) dx, \quad (1)$$

where $r = \sqrt{(x - x_0)^2 + z_0^2}$, and $H_1^{(1)}(\cdot)$ is the Hankel function of the first kind of order 1. This formula allows us to find the field at (x_0, z_0) if we know the field at the points where the light leaves the lens.

Another approximation, based on the fact that the lens is thin, is applied. Under this approximation, the lens changes only the phase of the incoming light. Specifically, if the incident light at $z = 0$ has

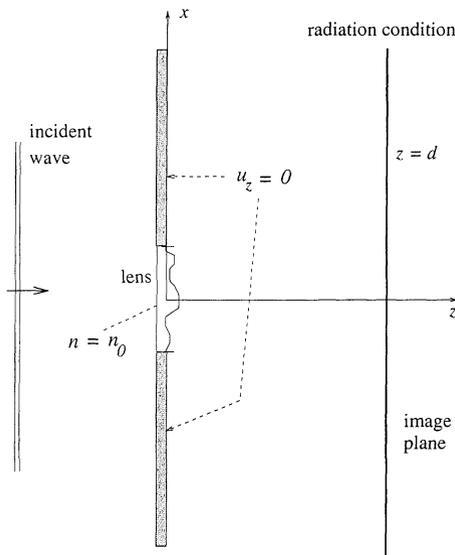


Figure 2 In the scalar model of light, the field $u(x, z)$ satisfies the Helmholtz equation. The lens has an index of refraction different from that of air. There are boundary conditions on the opaque parts of the screen, and where $x^2 + z^2 \rightarrow \infty$. The intensity at the image plane is $|u(x, d)|^2$.

amplitude $u(x, 0)$, the lens alters the amplitude to

$$u(x, 0) \exp i\varphi(x).$$

Here φ is the function

$$\varphi(x) = \frac{2\pi(n_0 - 1)}{\lambda} d(x), \quad -w \leq x \leq w,$$

where $d(x)$ is the thickness of film at x , n_0 is the index of refraction of the lens, and λ is the wavelength of light in air.

The diffractive optical element is produced by a material removal process. The resulting thickness profile $d(x)$ is piecewise constant over regular subintervals of width $\Delta x = 2w/n_x$, where n_x is a prescribed positive integer. Each value of $d(x)$ is an integer multiple of some fixed thickness t . This means that the corresponding phase $\varphi(x)$ is also piecewise constant. Assuming that the material removal rate is constant for each removal process, we choose t , depending on n_0 and λ , so that the phase $\varphi(x)$ is of the form

$$\begin{aligned} \varphi(x) &= f_j \frac{2\pi}{\ell}, & x_j \leq x < x_{j+1}, & \quad j = 0, 1, \dots, n_x - 1, & \quad (2) \\ f_j &\in A := \{0, 1, 2, \dots, \ell - 1\}, \end{aligned}$$

where $x_k = -w + k\Delta x$, $k = 0, 1, \dots, n_x$. The variable ℓ corresponds to the number of steps in the material removal process. Thus the unknowns of the problem are the components of a vector f of length \mathbb{R}^{n_x} , representing the thickness profile.

The optimization problem can be posed as follows. Assume the incident amplitude is constant $u(x, 0) = C$. We are given the required intensity $I_{\text{target}}(x)$ on the image plane $z = d$. We wish to solve

$$\min_{f \in \mathbb{R}^{n_x}} \|I - I_{\text{target}}\|^2, \quad (3)$$

$$I(x) = |u(x, d)|^2, \quad (4)$$

$$u(x, d) = \frac{ik}{2} C \int_{-w}^w \exp i\varphi(\xi) \frac{d}{r} H_1^{(1)}(kr) d\xi, \quad (5)$$

where $r = \sqrt{(x - \xi)^2 + d^2}$. The integer vector $f \in \mathbb{R}^{n_x}$ is related to $\varphi(x)$ through (2). Conservation of energy allows C to be determined from I_{target} .

The difficulties in this problem stem from the facts that (i) the required components of f are integer variables, (ii) the number of unknowns n_x is large in typical design problems, (iii) the functional in (3) is nonlinear. While there has been much progress in solving nonlinear integer programming problems, we were drawn to use Genetic Algorithms due to their simplicity. We will demonstrate their ability to “solve” this problem in a numerical example, and we will discuss their convergence properties.

3. REVIEW OF GENETIC ALGORITHMS

Genetic Algorithms are “search algorithms based on the mechanism of natural selection and natural genetics” [4]. Principles of evolution and heredity are used for the optimization of an objective function.

Genetic Algorithms are iterative and operate on a set of potential solutions, called population, and utilize concepts such as selection (survival of the fittest), crossover (information exchange), and mutation (adding the genetic diversity) [12]. Genetic Algorithms use only objective function information (no derivatives) and probabilistic transition rules. The origins of Genetic Algorithms go back to the work of Holland [7] and others in the 1970s. In recent years the algorithms have attracted a lot of interest, and have been used successfully for optimization in different areas, in particular, in diffractive optics design problems [3, 9, 10].

Let us consider the most basic version, called the Canonical Genetic Algorithm. Suppose we want to solve

$$\max_{\mathbf{x} \in X} f(\mathbf{x}), \quad (6)$$

where X is some feasible set, and $f(\mathbf{x}) > 0$ for all $\mathbf{x} \in X$. Genetic Algorithms use a *binary representation* of the variable \mathbf{x} . If \mathbf{x} is a continuous variable, it is given finite precision, and can thus be expressed in binary form. In our problem, each member of X has integer components, which can be represented easily by a vector of binaries. Therefore we can henceforth assume that $\mathbf{x} = (x^1, x^2, \dots, x^\nu)$, $x^j \in \{0, 1\}$. Every such string is called a *chromosome* or an *individual*. Binary entries x^j of the string \mathbf{x} are called *genes*. The cost function $f(\mathbf{x})$ is referred to as the *fitness*.

A Genetic Algorithm works on a set of N chromosomes (individuals), $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$, called a *population*. The initial population is formed at random. Each iteration updates a population to the next generation. The basic principle of the update is to increase the average fitness of the population.

At the beginning of each iteration, all individuals of the population are evaluated for fitness, that is, we compute $f(\mathbf{x}_i)$, $i = 1, 2, \dots, N$. Next the *probability for reproduction* is assigned to each chromosome, according to its relative fitness

$$p_i = \frac{f(\mathbf{x}_i)}{\sum_{j=1}^N f(\mathbf{x}_j)}$$

In this way, the fitter members of the population have a better chance of being chosen, with some fit members having the possibility of being chosen more than once. We pick N chromosomes in this fashion, and they form the *mating pool*.

The mating pool is subdivided at random into pairs. Each pair of parents, with a probability p_c say, performs a *crossover* to produce 2 offspring. A crossover is akin to exchanging genetic information. For example, in a 2-point crossover

$$\begin{array}{l} \mathbf{x}_1 = 1 \ 0 \ | \ 0 \ 1 \ 1 \\ \mathbf{x}_2 = 0 \ 1 \ | \ 1 \ 0 \ 1 \end{array} \quad \text{becomes} \quad \begin{array}{l} \mathbf{x}'_1 = 1 \ 0 \ | \ 1 \ 0 \ 1 \\ \mathbf{x}'_2 = 0 \ 1 \ | \ 0 \ 1 \ 1 \end{array}$$

The crossover point, which could be chosen at random, is indicated above by a vertical line. If a crossover is performed, 2 *offspring* replace their 2 parents, but otherwise both parents proceed to the next generation unchanged.

Finally, *mutation* is applied to each gene (binary element of the vector \mathbf{x}) of each offspring, with a probability p_m that is usually very small. The mutated gene is replaced by the opposite value ($0 \rightarrow 1$ or $1 \rightarrow 0$). Even when the probability of mutation is tiny, its role is important as an additional provider of diversity, so it is a source of valuable information that could be missing in the initial population.

This is the end of a single iteration. A new population is formed and the process is repeated until a convergence condition is satisfied or until the number of iterations reaches a prescribed limit. The process is summarized in the pseudo-code in Figure 3.

```

create initial population of  $N$  chromosomes
until convergence, do
    select parent pairs
    crossover
    mutate
    new population
end

```

Figure 3 The Canonical Genetic Algorithm

By convergence, we mean that the highest fitness value of the population has reached an acceptable value. Alternatively, we could stop the algorithm when the maximum of the population is sufficiently close to the average fitness of the population.

There are 3 parameters in the algorithm: (i) size of population N , (ii) crossover probability p_c , and (iii) mutation probability p_m . These parameters tend to be problem specific, and there are no general rules on how to set them [6]. An evolutionary strategy in which the parameters are allowed to evolve just like the population according to some rule has also been proposed [12].

An important variant to the Canonical Genetic Algorithm is one in which the fittest member of the population in each generation is saved. This guarantees that the maximum fitness value of the population is nondecreasing.

The Micro-Genetic Algorithm, first proposed in [10, 11], works on a small population of $m + 1$ chromosomes (for example, $m = 5$). There is an inner loop and an outer loop. Within the inner loop, *selection* and *crossover* take place, while the fittest member of the population is saved in each generation. After a convergence criterion is satisfied, we go out of the inner loop and regenerate a new population by mutating the fittest member m times. The new population then enters the inner loop. A feature suggested for this scheme is that the mutation probability p_m should be made smaller adaptively as we near convergence. The pseudo-code for the algorithm is given in Figure 4.

The crossover in Figure 4 is of a special type called ‘uniform crossover’. In this process, we take m members of the population and create m new ones by sequentially picking genes from the pool with uniform probability. That is, in creating a new \mathbf{x} of length ν , we pick the first gene (bit)

```

create initial population of  $m + 1$  chromosomes
for iteration = 1 to MAX
    until convergence, do
        save best
        generate  $m$  more chromosomes for the next
            population by performing uniform crossover
    end
    keep best chromosome
    mutate  $m$  times to create the other elements
        of the new population
end

```

Figure 4 The Micro-Genetic Algorithm.

from the first bits of each of the m members, with uniform probability. We pick the second bit from the second bits of the m members similarly, etc.

Two more features of the Micro-Genetic Algorithm are worth noting. Firstly, because we work with a small population, the inner loop converges fairly rapidly. Secondly, the purpose of the outer loop is to introduce diversity via *mutation*, in order to explore other areas of the search space.

One clear drawback of Genetic Algorithms is that they do not have a good stopping rule. Indeed, our numerical experience shows that improvements in the fitness function as we iterate tend to occur abruptly after a few steps that make little or no progress. However, we adopt the point of view that the problem is to get an acceptable design, i.e., a design whose residual in the fitness function is small relative to the norm of the target intensity. Then one can stop the iterations as soon as this threshold is crossed.

4. CONVERGENCE ANALYSIS OF MICRO-GENETIC ALGORITHMS

The application of Markov Chain analysis to Genetic Algorithms was first introduced by Rudolph [15], and developed in his book [16]. A more recent paper by Agapie [1] investigates the minimum criteria for convergence.

The main ideas are: (i) the population at any one point in the iteration can be viewed as a *state*, (ii) the finite binary representation of the variables makes the system a *finite* state machine, (iii) iterations are transitions from one state to another. Basic results from stochastic matrices [8, 13, 17] will be useful.

With this in mind, let \mathbf{x} be a binary vector of length ν , and let $m+1$ be the total number of members in a population. Each state is represented by a concatenation of $m + 1$ vectors $\mathbf{s}=[\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_m]$. We denote the state space by S . The total number of possible states is $|S| := L = 2^{(m+1)\nu}$. Now we index the states from 1 to L as s_1, s_2, \dots, s_L . Of all the possible states, there are $M = 2^\nu$ states in which the members $\mathbf{x}_i, i = 0, 1, \dots, m$, are *identical*. We define

$$U = \{s_i : \mathbf{x}_0 = \mathbf{x}_1 = \dots = \mathbf{x}_m\} = \{s_i : i = i_1, i_2, \dots, i_M\}, \quad (7)$$

say. Let $P\{\bullet\}$ denote the probability of the assertion \bullet . Consider the inner loop in the Micro-Genetic Algorithm, sketched in Figure 4, and let $s^{(k)}$ be the population after the k -th iteration.

Theorem 1 *The inner loop of the Micro-Genetic Algorithm in Figure 4 converges to a uniform population in the sense that*

$$\lim_{k \rightarrow \infty} P \{s^{(k)} \in U\} = 1.$$

Proof Let $\mathbf{P}_{\text{inner}} = \{p_{ij}\}$ be the transition matrix of the inner loop for each iteration (generation); i.e.,

$$p_{ij} = P \{s_i \rightarrow s_j\},$$

by the process of saving the best member and performing uniform crossover. Note that $\mathbf{P}_{\text{inner}}$ will be stochastic (rows summing to 1) and have nonnegative entries. If we order the states so that the first M states are in U , then the transition matrix $\mathbf{P}_{\text{inner}}$ has the form

$$\mathbf{P}_{\text{inner}} = \begin{bmatrix} I & 0 \\ A & B \end{bmatrix},$$

where I is the $M \times M$ identity matrix. This is because, if $s^{(k)} \in U$, then $s^{(k)}$ is not altered by the uniform crossover of the inner loop.

Next we study the properties of A and B . Note that the block A is of size $(L - M) \times M$, and corresponds to the transition from $s^{(k)} \notin U$ to $s^{(k+1)} \in U$. This means that it has only a single nonzero entry per row, corresponding to the element of U that is composed of $m + 1$ copies of the ‘best’ chromosome that was saved at the beginning of the inner iteration. The probability of reproducing m copies of the first member of $s^{(k)}$ is at least $(1/m)^{m\nu}$, which provides the crude lower bound

$$P \{s^{(k+1)} \in U\} \geq a := \left(\frac{1}{m}\right)^{m\nu}.$$

Therefore, if a_{ij} is the nonzero entry in row i of A , then $a_{ij} \geq a > 0$.

The submatrix B is of size $(L - M) \times (L - M)$. Since $\mathbf{P}_{\text{inner}}$ is stochastic

$$\sum_j b_{ij} \leq 1 - a < 1.$$

By direct calculation, we find that

$$\mathbf{P}_{\text{inner}}^k = \begin{bmatrix} I & 0 \\ A_k & B^k \end{bmatrix},$$

where $A_k = \left(\sum_{i=0}^{k-1} B^i\right) A = (I - B)^{-1}(I - B^k)A$. Now, if a matrix B has rows whose sums are less than one, then $B^k \rightarrow 0$ as $k \rightarrow \infty$. It follows that

$$\lim_{k \rightarrow \infty} \mathbf{P}_{\text{inner}}^k = \mathbf{P}_{\text{inner}}^\infty = \begin{bmatrix} I & 0 \\ A_\infty & 0 \end{bmatrix},$$

with $A_\infty = (I - B)^{-1}A$.

Therefore, starting with any distribution $p^{(0)}$, the final distribution

$$p^{(\infty)} = p^{(0)} \mathbf{P}_{\text{inner}}^\infty$$

has the property $p_j^{(\infty)} = 0$ for $j = M + 1, \dots, L$, thus proving the assertion of the theorem. ■

A convergence rate for the inner loop can be estimated. From the probability transition $p^{(k)} = p^{(0)} \mathbf{P}_{\text{inner}}^k$, we deduce

$$P \{s^{(k)} \in U\} = \sum_{\{i: s_i \in U\}} p_i^{(k)} = 1 - P \{s^{(k)} \notin U\} = 1 - \sum_{\{i: s_i \notin U\}} p_i^{(k)}.$$

Further, the nature of the transition matrix implies

$$p_i^{(k)} = \sum_{\{j: s_j \notin U\}} p_j^{(0)} b_{ji}^{(k)}, \quad \{i : s_i \notin U\},$$

where $B^k = [b_{ij}^{(k)}]$. By using $\sum_j b_{ij} \leq 1 - a$ for all i , and by summing both sides of the equation, we get

$$\begin{aligned} \sum_{\{i: s_i \notin U\}} p_i^{(k)} &= \sum_{\{i: s_i \notin U\}} \sum_{\{j: s_j \notin U\}} p_j^{(0)} b_{ji}^{(k)} \\ &= \sum_{\{j: s_j \notin U\}} p_j^{(0)} \sum_{\{i: s_i \notin U\}} b_{ji}^{(k)} \leq (1 - a)^k \sum_{\{j: s_j \notin U\}} p_j^{(0)} \leq (1 - a)^k. \end{aligned}$$

Therefore we have established

$$P \{s^{(k)} \in U\} \geq 1 - (1 - a)^k,$$

where $a = (1/m)^{m\nu}$.

Each single step of the outer loop of the Micro-Genetic Algorithm can be expressed using the transition matrix

$$\mathbf{P}_{\text{outer}} = \mathbf{M}\mathbf{P}_{\text{inner}}^{\infty}, \tag{8}$$

where \mathbf{M} represents the mutation process, and where the entire inner loop has been encapsulated in $\mathbf{P}_{\text{inner}}^{\infty}$. Now we can state the convergence result.

Theorem 2 *Assume that there is a unique global maximizer of the optimization problem (6). Let the probability of mutation in the outer loop be fixed with $p_m \in (0, 1)$. Then the iterations of the Micro-Genetic Algorithm of Figure 4 converge to a population consisting of $m + 1$ copies of the global maximizer of the cost function.*

Proof Recall that an unknown binary vector of length ν can take 2^ν values. When these 2^ν different vectors are arranged in descending order of fitness, we denote them by $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_{2^\nu}$. Moreover, we order the populations of size $(m + 1)$ by *class*. The first class contains populations whose first member is \mathbf{x}_1 , the second class contains populations whose first member is \mathbf{x}_2 , and so on, where the members of each population need not be in order of fitness. We have a total of 2^ν classes. In each class, we let the first population be the one that has $m + 1$ copies of its leading member. This process is nothing more than renumbering the states in a particular order.

Next, recall that the inner loop takes any population into one that has identical members. Consider the transition

$$p^{(k+1)} = p^{(k)}\mathbf{P}_{\text{inner}}^{\infty}.$$

The new vector $p^{(k+1)}$ has only 2^ν nonzeros, corresponding to the populations whose members are identical in the 2^ν classes. Note also that the populations can only go up in class, or remain in the same class, because we always save the fittest member of the population. Therefore we deduce that the matrix $\mathbf{P}_{\text{inner}}^{\infty}$ must be lower triangular, and of the

form

$$\mathbf{P}_{\text{inner}}^\infty = \begin{bmatrix} \boxed{v^{(1,1)}} & & & & \\ \boxed{v^{(2,1)}} & \boxed{v^{(2,2)}} & & & \\ & \vdots & \vdots & \ddots & \\ \boxed{v^{(2^\nu,1)}} & \boxed{v^{(2^\nu,2)}} & \dots & \boxed{v^{(2^\nu,2^\nu)}} & \end{bmatrix}. \tag{9}$$

The nonzero elements of the matrix are confined to the 2^ν columns indicated by thin rectangles and labelled $v^{(i,j)}$. From our indexing scheme, we know that

$$v^{(1,1)} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix},$$

because $\mathbf{P}_{\text{inner}}^\infty$ is stochastic and each row of the block corresponding to the first class has a single nonzero entry. Therefore the entry must be 1. Let us consider $v^{(i,1)}$. Each element of this vector corresponds to the probability of a transition to the first class from class i . We argue that at least one element of $v^{(i,1)}$ must be equal to 1. These elements correspond to the existence of populations in class i with probability 1 of exiting the inner loop in class 1. They have \mathbf{x}_1 as a member, and therefore would immediately be promoted to class 1 by the inner loop, and would stay in that class thereafter.

Now we look at the transition matrix \mathbf{M} corresponding to mutation. The population, upon exit from the inner loop, has $m + 1$ identical members. A total of m new members are created by sequentially ‘flipping’ bits of the chromosome with probability p_m . The probability that every bit is flipped is $p_m^{m\nu}$, whereas the probability that every bit remains the same is $(1 - p_m)^{m\nu}$. Therefore the nonzero part of the transition matrix \mathbf{M} satisfies

$$\min_{ij} [\mathbf{M}]_{ij} \geq a := \min\{p_m^{m\nu}, (1 - p_m)^{m\nu}\}.$$

We keep the original member before mutation and label it as the first member, thus defining the class of the population, so the mutation process cannot change the class. Therefore the transition matrix corre-

sponding to mutation has the block diagonal structure

$$\mathbf{M} = \begin{bmatrix} \mathbf{M}_1 & & & \\ & \mathbf{M}_2 & & \\ & & \ddots & \\ & & & \mathbf{M}_{2^\nu} \end{bmatrix},$$

where each block is a $2^{m\nu}$ by $2^{m\nu}$ stochastic matrix. Using (9) and the above, the product (8) takes the form

$$\mathbf{P}_{\text{outer}} = \begin{bmatrix} \boxed{w^{(1,1)}} & & & \\ \boxed{w^{(2,1)}} & \boxed{w^{(2,2)}} & & \\ & \vdots & \vdots & \ddots \\ \boxed{w^{(2^\nu,1)}} & \boxed{w^{(2^\nu,2)}} & \dots & \boxed{w^{(2^\nu,2^\nu)}} \end{bmatrix},$$

where $w^{(i,j)} = \mathbf{M}_i v^{(i,j)}$. The fact that \mathbf{M}_1 is stochastic implies

$$w^{(1,1)} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = v^{(1,1)}.$$

Observe also that

$$w_j^{(i,1)} = [\mathbf{M}_i v^{(i,1)}]_j = \sum_l [\mathbf{M}_i]_{jl} v_l^{(i,1)} \geq a \sum_l v_l^{(i,1)} \geq a, \quad (10)$$

the last inequality being valid because we have deduced that at least one element of $v^{(i,1)}$ is unity. Further, showing only the first row of $\mathbf{P}_{\text{outer}}$ explicitly, we can write

$$\mathbf{P}_{\text{outer}} = \left[\begin{array}{c|ccc} 1 & 0 & \dots & 0 \\ \hline \vdots & & & \\ & & Q & \\ \vdots & & & \end{array} \right],$$

which implies

$$\mathbf{P}_{\text{outer}}^k = \left[\begin{array}{c|ccc} 1 & 0 & \dots & 0 \\ \hline \vdots & & & \\ & & Q^k & \\ \vdots & & & \end{array} \right].$$

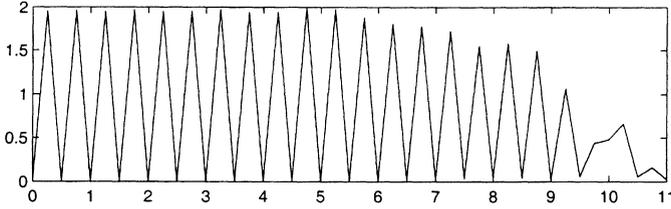


Figure 5 The target intensity pattern.

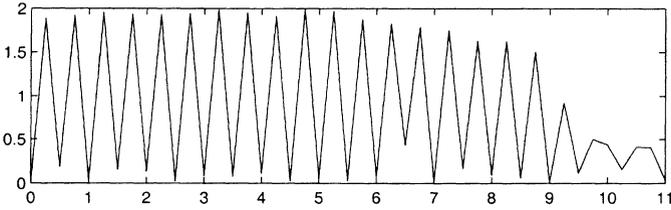


Figure 6 The intensity pattern of the optimized diffractive optical element.

The matrix Q has the property $\sum_j Q_{ij} \leq 1 - a < 1$ by (10), so $Q^k \rightarrow 0$ as $k \rightarrow \infty$. It follows from the stochastic property of $\mathbf{P}_{\text{outer}}$ that

$$\lim_{k \rightarrow \infty} \mathbf{P}_{\text{outer}}^k = \mathbf{P}_{\text{outer}}^\infty = \left[\begin{array}{c|cccc} 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & \cdots & 0 \end{array} \right].$$

Therefore, the probability that the population reaches a state in which all the members are \mathbf{x}_1 is 1 in the limit. ■

5. NUMERICAL EXPERIMENT

We applied a version of the Micro-Genetic Algorithm to the problem (3). In the example, we chose a symmetric profile with 1000 unknowns. The half-aperture is $x \in [0, 10]$, so the width of each ‘bump’ is 0.001. The diffractive optical element has 4 levels, so $A = \{0, 1, 2, 3\}$. The desired target image was derived from a rough sampling of the intensity at the image plane $d = 1$ over the interval $x \in [0, 11]$. It was obtained by solving the forward problem with a known diffractive optical element at wavenumber $k = 25$, and is shown in Figure 5. The image found by optimization is given below it in Figure 6.

We used a version of the optimization algorithm that changes the mutation probability p_m adaptively. The algorithm ran for a total 500

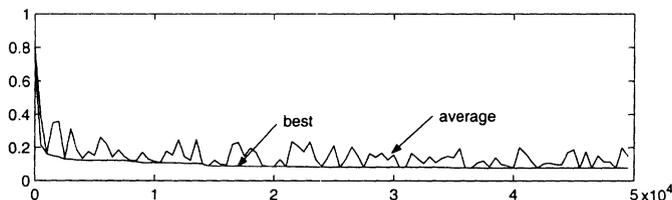


Figure 7 The progress of the optimization process. Since we fix the number of inner loop iterations to 100, the exiting population is not uniform. This graph shows the average fitness and the fitness of the best member upon exit from the inner loop.

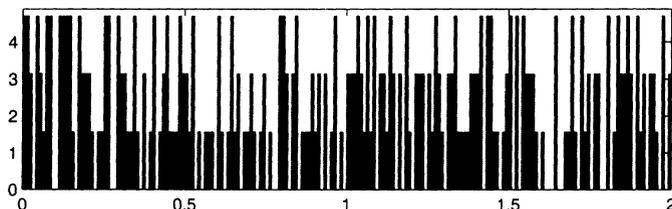


Figure 8 Part of the optimized diffractive optical element. The phase is shown as a function of position.

iterations of the outer loop, and each iteration ran for 100 generations. The total number of function evaluations with the population size $(m + 1) = 5$ is 2×10^5 . Note that the search space has over 10^{600} elements. The reduction in the residual at the end of each iteration in the outer loop is shown in Figure 7. Part of the calculated profile found is shown in Figure 8.

We were able to run larger examples, with 20000 unknowns, with some success. A disturbing feature of the Micro-Genetic Algorithm, however, is that many iterations may fail to reduce the cost function, which makes it difficult to decide when to stop.

We devised a simple way to obtain crude bounds on the value of the cost function at the optimum. To obtain a lower bound, we relax the search space to continuous variables. This is an unconstrained optimization problem and can be solved by standard methods. In order to get an answer $\varphi(x)$ that is between 0 and 2π , we simply add or subtract integer multiples of 2π if necessary. To obtain a crude upper bound, we take the values of $\varphi(x)$ calculated in the manner described, and round them to the nearest integer multiples of the phase increment. While these bounds are crude, they do provide some guidance for stopping the iterations.

6. DISCUSSION

We have described a problem of designing diffractive optical elements using an optimization strategy. Because of the integer constraints imposed by the manufacturing process, and the size of the problem, we chose a rather unconventional method for solving the optimization calculation. The method is a Micro-Genetic Algorithm. We have analyzed its convergence properties, finding that, while there is convergence, there is no easy way to estimate the convergence rate. Numerical experiments with the method yield satisfactory results. A good match of the target intensity profile is usually achieved by this method.

Acknowledgments

The authors are grateful to Drs. David Misemer and Alessandra Chiareli of 3M for valuable discussions and support of this work.

References

- [1] A. Agapie (1998), Genetic Algorithms: Minimal conditions for convergence, *Lecture Notes in Computer Science*, 1363, pp. 183–193.
- [2] M. Born and E. Wolf (1964), *Principles of Optics*, Pergamon Press.
- [3] D. Brown and A. Kathman (1995), Multi-element diffractive optical designs using evolutionary programming, *SPIE*, 2404, pp. 17–27.
- [4] D.E. Goldberg (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company.
- [5] J.W. Goodman (1968), *Introduction to Fourier Optics*, McGraw-Hill, New York.
- [6] J.J. Grefenstette (1986), Optimization of control parameters for Genetic Algorithms, *IEEE Transactions on Systems, Man, and Cybernetics*, 16(1), pp. 122–128.
- [7] J.H. Holland (1975), *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor.
- [8] M. Iosifescu (1980), *Finite Markov Processes and Their Applications*, Wiley, Chichester.
- [9] E.G. Johnson, A.D. Kathman, D.H. Hochmuth, A.L. Cook, D.R. Brown and W.F. Delaney (1993), Advantages of Genetic Algorithm optimization methods in diffractive optic design, *SPIE*, CR49, pp. 54–74.
- [10] E.G. Johnson and M.A.G. Abushagur (1995), Microgenetic-algorithm optimization methods applied to dielectric gratings, *JOSA A*, 12(5), pp. 1152–1160.

- [11] K. Krishnakumar (1989), Micro-Genetic algorithms for stationary and non-stationary function optimization, *SPIE*, 1196, pp. 289–296.
- [12] Z. Michalewicz (1992), *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, Berlin.
- [13] J.S. Rosenthal (1995), Convergence rates for Markov chains, *SIAM Review*, 37(3), pp. 387–405.
- [14] S. Rudnaya, F. Santosa and A.O. Chiareli (1998), Optimal design of a diffractive optical element, in *Proceedings of the 4th International Conference on Mathematical and Computational Aspects of Wave Propagation*, SIAM, Philadelphia.
- [15] G. Rudolph (1994), Convergence analysis of Canonical Genetic Algorithms, *IEEE Transactions on Neural Networks*, 5(1), pp. 96–101.
- [16] G. Rudolph (1997), *Convergence Properties of Evolutionary Algorithms*, Verlag Dr. Kovac, Hamburg.
- [17] E. Seneta (1981), *Non-negative Matrices and Markov Chains*, Springer-Verlag, New York.