Discrete PSO with Simulated Annealing applied to improve multilevel phase holograms design

Ivan Cruz Aceves¹, Manuel Ornelas Rodríguez¹, and Arturo Hernández Aguirre²

¹ Instituto Tecnológico de León, León, Guanajuato, México.
² Centro de Investigaciones en Matemáticas, Guanajuato, Guanajuato, México.
{ivan.cruzac@gmail.com, mornelas67@yahoo.com.mx, artha@cimat.mx}

Abstract. This paper presents an efficient hybrid algorithm based on Particle Swarm Optimization (PSO) with discrete values using Simulated Annealing (SA), denominated (DPSOSA), for the optimum design of multilevel phase computer generated holograms (CGH’s). This algorithm deals with discrete values using the periodicity of the Discrete Fourier Transform to improve the diffraction efficiency of CGH’s. The experimental results indicate that DPSOSA can enhance the performance obtained with Genetic Algorithm (GA), Discrete Particle Swarm Optimization (DPSO) and Iterative Fourier Transform Algorithm (IFTA). The behaviour of DPSOSA is discussed.

Key words: Combinatorial Optimization, Computer Generated Holograms, Particle Swarm Optimization, Evolutionary Computation, Genetic Algorithms, Hybrid Algorithms, Simulated Annealing

1 Introduction

Computer generated holograms (CGH’s) are attractive for a variety of applications, when reduced dimensions, light weight, and low replication cost are required. Some applications include optical testing, image displays, optical security and metrology, to mention but a few. A number of algorithms have been proposed for designing computer generated holograms [1][2]. The Iterative Fourier Transform Algorithm (IFTA) is one of the most popular. This algorithm was developed by Gerchberg and Saxton in 1972 [3]. The IFTA is efficient and can handle large amounts of data, however, the algorithm is sensitive to initial parameters and it is prone to stagnate in local minima. This study proposes the use of global optimization algorithms that do not depend on the starting parameters of IFTA, and that are effective in avoiding local minima. The paper is organized as follows; section 2 deals with the CGHs design using the IFTA, whose drawback is the inability to escape from local optima. Section 3 describes evolutionary and bio-inspired algorithms like Genetic Algorithm and Particle Swarm Optimization in discrete domain. In section 4 it is explained the proposed hybrid algorithm DPSOSA. Section 5 is concerned with the experimental setup, the results, their analysis and discussion. Finally section 6 presents some conclusions and future work.
2 CGH’s design by using IFTA

The algorithm for hologram calculation with a standard version of IFTA [4], is shown in figure 1. The holograms were designed to work for transmission by modulating only the phase of the incident light and reconstructing the image of the desired object at the far field. Phase holograms are more attractive than amplitude holograms, because the former provide higher diffraction efficiency since ideally no energy is absorbed.

The first step is to calculate the transmission function of the hologram \((G_j(u))\) from the original object \(g_0(x)\) by means of the Discrete Fourier Transform (DFT). The operators \(U\) and \(X\) are the rules of the Fourier plane \((u)\) and the object plane \((x)\), respectively. The \(U\) operator replaces the module of \(G_j(u)\) by the unit value in the Fourier plane and retains its phase. The \(X\) operator maintains the phase of \(g_j(x)\) and replaces its amplitude by the original signal \(g_0(x)\). The common stop criterion of IFTA is by the approximation error between the original and the reconstructed signals, or by prefixed number of iterations. Due to limitations in the manufacturing methods of diffractive elements, the next step is to discretize the continuous transmission function obtained by IFTA in some number of phase levels \((N)\). Then, the discrete phase function is encoded as an array of pixels in gray scale, thereby generating the hologram mask. IFTA’s problem resides in the fact that it cannot achieve the established theoretical diffraction efficiency given by equation 1 [5], and shown in table 1. The efficiency problem is caused by stagnation at local minima.

\[
\eta = \left[ \left( \frac{N}{\pi} \right) \sin \left( \frac{\pi}{N} \right) \right]^2
\] (1)
Table 1. Theoretical diffraction efficiencies as a function of the number of phase levels.

<table>
<thead>
<tr>
<th>2 phase levels</th>
<th>3 phase levels</th>
<th>4 phase levels</th>
<th>5 phase levels</th>
<th>6 phase levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>40.5%</td>
<td>68.3%</td>
<td>81%</td>
<td>87.5%</td>
<td>91.1%</td>
</tr>
</tbody>
</table>

The CGH’s design is a common NP-hard combinatorial optimization problem. The objective is, applying IFTA, to obtain output holograms which will initialize the evolutive algorithms. These algorithms will search the optimal combination of discretized pixel values that maximizes the diffraction efficiency, given by equation 2. This ratio is the fitness function for the evolutive algorithms.

\[
\text{Diffraction Efficiency} = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \text{ImgRecons}(x, y)}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \text{Img}(x, y)} \times 100
\]  

(2)

Where \(\text{ImgRecons}(x, y)\) denotes reconstructed image and \(\text{Img}(x, y)\) is the original one.

3 Optimization Techniques

Evolutionary algorithms (EA’s) [6] are bio-inspired metaheuristics that have proved very robust to solve hard or complex problems. These are population based algorithms, where every individual of the population encodes a candidate solution.

The mechanic of the algorithm is to evolve and improve the quality of the population. The repeated application of selection, reproduction and mutation operators produces variations with improved quality. A fitness function measures the goodness or quality of the solution carried by every individual.

3.1 Genetic Algorithms

Genetic algorithms (GA) have had a great deal of success in solving search and optimization problems [7][8][9]. The first step of a GA is to create the initial population, and to measure the quality of every individual with the fitness function. A selection operator, such as binary tournament or proportion, is applied and a temporal population with above average fitness is stored. Then, a reproduction operator, such as one point or two-point crossover, is applied to the pool of selected individuals and new individuals are spawned. After the application of the mutation operator, the new individuals populate the new generation.
3.2 Discrete Particle Swarm Optimization

Particle Swarm Optimization algorithm introduced by Kennedy and Eberhart [10], is applied in continuous spaces optimization.

Since its publication, many related works have appeared to adapt this technique for solving problems involving discrete spaces [11]. Equations 3 and 4 are preserved in modified algorithms in order to work in a discrete manner.

\[ V_i(t + 1) = V_i(t) + c_1 \omega * (Y_i(t) - X_i(t)) + c_2 \omega * (G - X_i(t)) \] (3)

This is the velocity equation where \( c_1 \) and \( c_2 \) are the inertia constants, \( \omega \) is learning factor, \( G \) is gbest i.e. the particle with the best fitness, \( Y \) is the best position of the present particle \( X \). Once the velocity term has been computed the new particle position is calculated via equation 4.

\[ X_i(t + 1) = X_i(t) + V_i(t + 1) \] (4)

The adaptation consists in the use of these canonical PSO equations, which are applied to the original population. Then the modified population is checked against the original one. Wherever a value is not repeated in corresponding individuals, they are marked as feasible for mutation. Finally, these individuals are replaced by feasible solutions by using a uniform random number generator. This is called random mutation.

3.3 Simulated Annealing

Simulated Annealing (SA) is a stochastic optimization algorithm that leads to a stationary state which is described by the Boltzmann distribution [12].

In essence this procedure resembles cooling a crystal down slowly such that defects and other lattice impurities can heal out and a pure crystalline structure (global minimum) is achieved at low temperature [13].

Its equilibrium properties at a temperature \( Temp \) are determined by the Boltzmann distribution with equation 5, and \( \Delta \) is the difference between the current and the new solution.

\[ Boltz = e^{-\Delta/Temp} \] (5)

4 Hybrid Discrete Particle Swarm Optimization with Simulated Annealing Algorithm Proposed

A promising strategy has been the combination of different metaheuristics in order to exploit their strengths and thus obtaining better results for difficult problems, where no
good approximations are achieved by using a single technique as shown in [14][15].

The related works with the fact of combining Particle Swarm Optimization and Simulated Annealing have demonstrated to obtain acceptable results due to the fact that velocity and position of PSO equations can be adapted to work in discrete environments as shown in [16]. Below the proposed algorithm DPSOSA is presented.

**Algorithm 1 DPSOSA Algorithm**

1. Initialize the control parameters, $\omega, c_1, c_2, \gamma, D, \theta, iBest, t=0, Temp, \alpha, \psi$
2. Create and initialize the population $Pop[k, D]$ to apply Simulated Annealing,
3. SimulatedAnnealing($Temp, \alpha, \psi, Pop$)
   3.1 for each individual $\varphi$ do
   3.2 while($Temp > \psi$) do
   3.3 next = Permutation (current)
   3.4 $\Delta = f(\text{next}) - f(\text{current})$
   3.5 if($\Delta > 0$) Assign next to current
   3.6 else if($\varphi \text{ Mod } 2 == 1$) Apply Boltzmann Operator Decrement $Temp$ with $\alpha$
   3.7 end
   3.8 end
   3.9 return Pop // to Initialization of DDE
4. while stopping condition not true do
5. for each individual $X_{t}(i) \epsilon Pop[t,D]$ do
6. DiscreteMutation($X_{t}, \theta$)
7. Evaluate the fitness $f(X_{t})$
8. if $f(X_{t})$ is better than $f(X_{p_{best}})$ Add $X_{t}$ to $X_{p_{best}}$
9. if $f(X_{t})$ is better than $f(Pop(iBest))$ Add $X_{t}$ to $Pop(iBest)$
10. Update Velocity ($X_{t}, \omega, c_1, c_2$)
11. Update Position ($X_{t}$)
12. end
13. end
14. return Pop($iBest$)/as the solution

Where $c_1 \gamma c_2$ are inertia weights, $\gamma$ individuals in population, $\omega$ is learning factor, $\theta$ is mutation percentage, $iBest$ index best solution and $Temp, \alpha, \psi$ are control parameters of Simulated Annealing.

The Simulated Annealing algorithm allows to initialize the population of DPSO, yielding greater variability to the information that the individual gets directly from IFTA.
5 Experiments

In this section, it is compared the DPSOSA model with the previously described algorithms. All the models have been tested with the same image and conditions; the image size is 128 x 128 pixels. It is analyzed the performance to compare these models in 30 executions.

The features of algorithms are described as follows:

- Genetic Algorithm with Proportional Selection (GAP):
  - 20 individuals
  - 100 generations
  - crossover = 80%
  - mutation = 5%
- Genetic Algorithm with Tournament Selection (GAT): same conditions GAP.
- Genetic Algorithm with Boltzmann Selection (GAB): same conditions GAP.
- Genetic Algorithm with Sigma Scaling Selection (GAS): same conditions GAP.
- Discrete Particle Swarm Optimization (DPSO):
  - 20 particles
  - 100 iterations
  - inertia = 0.7
  - learning factor = 2
  - mutation = 50%
- Discrete Particle Swarm Optimization with Simulated Annealing (DPSOSA):
  - same conditions DPSO
  - Temp = 10
  - alpha =0.999
  - epsilon=0.01

In table 2 and 3, it is showed the maximum and average diffraction efficiencies reported by the algorithms.

Table 2. Maximum diffraction efficiencies obtained by the implemented algorithms.

<table>
<thead>
<tr>
<th>Method</th>
<th>Phase levels</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>IFTA</td>
<td></td>
<td>36.8832</td>
<td>61.3139</td>
<td>71.9660</td>
<td>76.9917</td>
<td>79.9846</td>
</tr>
<tr>
<td>GAP</td>
<td></td>
<td>38.2017</td>
<td>64.1379</td>
<td>75.0170</td>
<td>79.9976</td>
<td>82.6758</td>
</tr>
<tr>
<td>GAT</td>
<td></td>
<td>38.1199</td>
<td>63.9807</td>
<td>74.6681</td>
<td>79.7798</td>
<td>82.7991</td>
</tr>
<tr>
<td>GAB</td>
<td></td>
<td>38.0825</td>
<td>63.4223</td>
<td>74.1979</td>
<td>79.7394</td>
<td>82.7411</td>
</tr>
<tr>
<td>GAS</td>
<td></td>
<td>38.1085</td>
<td>63.3135</td>
<td>74.5224</td>
<td>79.8197</td>
<td>82.6829</td>
</tr>
<tr>
<td>DPSO</td>
<td></td>
<td>38.1853</td>
<td>63.3100</td>
<td>74.2949</td>
<td>79.2902</td>
<td>82.0156</td>
</tr>
<tr>
<td>DPSOSA</td>
<td></td>
<td>38.3106</td>
<td>63.4389</td>
<td>74.9432</td>
<td>80.1602</td>
<td>83.0577</td>
</tr>
</tbody>
</table>
Table 3. Average diffraction efficiencies reported by the algorithms.

<table>
<thead>
<tr>
<th>Method</th>
<th>Phase levels</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>IFTA</td>
<td>36.2821</td>
<td>60.7147</td>
<td>71.6853</td>
<td>76.7688</td>
<td>79.6112</td>
<td></td>
</tr>
<tr>
<td>GAP</td>
<td>37.8340</td>
<td>63.3993</td>
<td>74.4306</td>
<td>79.5346</td>
<td>82.5065</td>
<td></td>
</tr>
<tr>
<td>GAT</td>
<td>37.5643</td>
<td>63.2141</td>
<td>74.3967</td>
<td>79.4963</td>
<td>82.5386</td>
<td></td>
</tr>
<tr>
<td>GAB</td>
<td>37.5849</td>
<td>63.1036</td>
<td>74.3460</td>
<td>79.4334</td>
<td>82.4670</td>
<td></td>
</tr>
<tr>
<td>GAS</td>
<td>37.4874</td>
<td>63.0898</td>
<td>74.2884</td>
<td>79.4244</td>
<td>82.4182</td>
<td></td>
</tr>
<tr>
<td>DPSO</td>
<td>37.8008</td>
<td>62.8782</td>
<td>74.0260</td>
<td>79.0729</td>
<td>81.8170</td>
<td></td>
</tr>
<tr>
<td>DPSOSA</td>
<td>37.9287</td>
<td>63.3163</td>
<td>74.7011</td>
<td>79.9399</td>
<td>82.8017</td>
<td></td>
</tr>
</tbody>
</table>

Figure 2 shows the reconstructed images and their hologram of 256x256 pixels with 2, 3 and 4 phase levels. In figure 3 it is showed the experimental image of 128x128 pixels used in the implemented algorithms. Figures 4, 5 and 6 show the diffraction efficiency comparisons among the algorithms.

Fig. 2. Reconstructed image and their hologram with a) 2 phase levels, b) 3 phase levels, c) 4 phase levels.
Fig. 3. Experimental Image.

Fig. 4. Diffraction efficiency comparison between IFTA and theoretical values.

Fig. 5. Diffraction efficiency comparison between Genetic Algorithms and theoretical values.
6 Conclusions

In this work a hybrid algorithm design composed by robust global optimization techniques has been presented, demonstrating to obtain a better performance during the test. Discrete Particle Swarm Optimization with Simulated Annealing obtained the best solution for the CGH’s design. Ideas for future work involve extending the proposed strategies to solve problems including another heuristics. It would be also interesting to work the approach for including not uniform domain size for every variable.

Acknowledgments. Authors thank to CONCYTEG and CONACYT for financial support.

References