Design of triangular FBG filter for sensor applications using composite differential evolution

XIANGTAO LI, MINGHAO YIN*
College of Computer Science, Northeast Normal University, Changchun, 130117, P.R. China

Fiber Bragg grating is one of the most important optical communication optimization problems of current interest. The Triangular FBG Filter (TFBG) can be used as a readout device in FBG-based sensor application. This article describes the application of a recently develop metaheuristic algorithm, known as the composite differential evolution (CoDE), to optimize a TFBG filter design problem for a given grating length. The CoDE has been used to solve a difficult instance of the design problem and the optimization goal in each example is easily achieved. The experimental results of the CoDE algorithm have been shown better than the recently published results obtained using CMAES algorithm is a statistically meaningful way.

(Received December 8, 2011; accepted February 20, 2012)

Keyword: Triangular FBG, Sensor application, Differential evolution, Composite differential evolution, Metaheuristics

1. Introduction

Fiber Bragg grating has an important role in diverse fields ranging from optical communications to optical sensing. In order to handle the advantages such as low insertion loss, low polarization sensitivity, compactness, low cost, all-fiber geometry and ease of fabrication, the Fiber Bragg grating is realized as optical filters in wavelength division multiplex systems and fiber sensor systems [1]. Then, experiments have been verified that the TFBG filter as wavelength readout devices have the advantages of high sensitivity as well as immunity from the light source instability, power fluctuations and the uneven power distribution of source spectrum. The design and fabrication of FBGs have recently attracted much attention in the field of fiber optics [2]. Many methodologies have been proposed to solve the triangular FBG in the literature. In inverse scattering based FBG filter design methods, for a fixed grating length, coupling/index modulation coefficients are evaluated. Therefore, it is very difficult to find optimal grating length fro the desired TFBG filter specifications.

It is well known that the classical optimization methods need a stating point that is reasonably close to the final solution, or they are likely to be stunk in a local minimum. The quality of the solution strongly depends on the estimation of the initial values. If the initial values fall in a region of the solution space where all the local solutions are poor, a local search is limited to finding the best of these poor solutions. Since these disadvantages of the classic optimization techniques, the heuristic optimizations techniques such as such as genetic algorithm (GA), particle swarm optimization algorithm (PSO), and differential evolution [3-8] have been proposed to accurately solve FBG filter.

Particularly, Differential evolution (DE) [9] is a method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. DE is a simple yet powerful population based, direct search algorithm with the generation and test feature for global optimization problems. The basic idea of DE is to create new candidate solutions by combining the parent individual and several other individuals of the same population, and a candidate replaces the parent only if it has better fitness. Previous work showed the differential evolution to be an effective algorithm for some kind of problems. Furthermore, the differential evolution is well suitable to solve this problem because of the algorithm is easier to implement than GA and applied design problem with both discrete and continuous design parameters. In order to demonstrate the advantages of the proposed design, the results obtained using CMAES are compared.

In this paper, we will use the composite differential evolution (CoDE) to perform triangular FBG filter. Here, this algorithm employed three trial vector generation strategies and three control parameter settings. These strategies and parameter settings have distinct advantages and therefore they can complement one another. In CoDE, each strategy generated its trail vector with a parameter setting randomly selected from the parameter candidate pool. This algorithm is very easy to implement. Previous work showed the CoDE algorithm to be an effective algorithm for the numerical global optimization. Furthermore, this algorithm is well suitable to solve the triangular FBG filter because of the algorithm is easier to implement than GA and applied design problem with continuous design parameters. In order to demonstrate the advantages of the proposed design, the results obtained are compared with CMAES. The experimental results show the CoDE algorithm is very competitive.

The rest of this paper is organized as follows: in section 2 we will introduce the problem formulation, differential evolution and the composite differential evolution. Section 3 describes experimental result. In the last section we conclude this paper and point out some future research directions.

2. Theory

The problem described is as follows: in order to design of a FBG filter, the grating length is divided into n piecewise uniform sections. Let us assume the index modulation profiles are known for all the n sections. Then, for the entire grating, the transfer matrix can be obtained by chain multiplying the individual transfer matrices of the grating sections using the following equation:

$$\begin{bmatrix} E_a(0) \\ E_b(0) \end{bmatrix} = T_1 T_2 \cdots T_k \cdots T_n \begin{bmatrix} E_a(L) \\ E_b(L) \end{bmatrix} \qquad (1)$$

Here, E_a and E_b denote the forward and backward complex electrical fields, and T_k represent the transfer matrix of the *k*th section. By using the boundary constraint $E_b(L)$ =0, the complex reflection coefficient r can be obtained as

$$r = \frac{E_b(0)}{E_a(0)} \qquad (2)$$

The main requirements of an FBG filter for sensor application are a triangular spectrum with linear edges and a large bandwidth.

2.1 Objective function of TFBG filter

The main objective of the FBG filter design is to find an optimum index modulation profile and to design a linear edge reflectivity spectrum and desired bandwidth. In general, evolutionary algorithms use the concept of fitness to represent a particular solution that satisfies the design objective. For the TFBG filter spectrum. In designing the TFBG filter for specified bandwidth, the sum of square errors between the desired triangular and calculated spectra is used as the fitness function as follows:

$$F = \sum_{i \in window} (R_{d,i} - R_i)^2 \qquad (3)$$

Where $R_{d,i}$ and R_i are the desired and calculated reflective power at *i*th wavelength in the selected wavelength window. In the reflective power, the change in wavelength should be high to achieve better accuracy in wavelength window. Therefore, the highest value of the reflective power is at the centre wavelength. The triangular spectrum mask with 0.25 nm bandwidth and 90% reflective power at centre wavelength is show in Fig. 1.

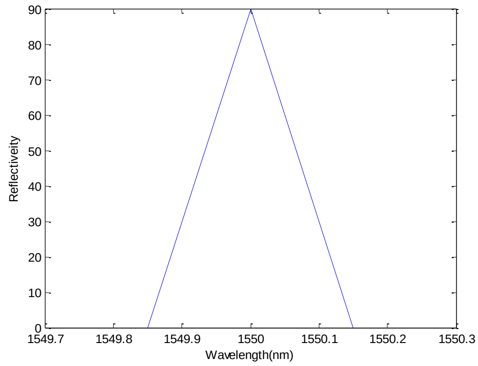

Fig. 1. Triangular spectrum mask with a bandwidth of BW=0.25 nm.

2.2 Differential evolution algorithm

Differential Evolution (DE) is an Evolutionary Algorithm first introduced by Storn and Price [9]. Similar to other evolutionary algorithms particularly genetic algorithm, DE uses some evolutionary operators like selection recombination and mutation operators. Different from genetic algorithm, DE use distance and direction information from current population to guide the search process. The crucial idea behind DE is a scheme for producing trial vectors according to the manipulation of target vector and difference vector. If the trail vector yields a lower fitness than a predetermined population member, the newly trail vector will be accepted and be compared in the following generation. Different kinds of strategies of DE have been proposed based on the target vector selected, the number of difference vectors used. In this paper, we use two strategies, DE/rand/1/bin, described as follows.

For each target vector $x_i(t)$, trail vector $v_i(t)$, i = 1, …, NP, let N be the dimension of target vector, and G be the G generation. the mutant vectors are generated in these DE/rand/1/bin strategies respectively:

For DE/rand/1/bin

$$v_{i,G} = x_{a,G} + F(x_{b,G} - x_{c,G}) \qquad (4)$$

Where $a,b,c,d \in [1,\cdots\cdots,NP]$ are randomly chosen integers, and $a \neq b \neq c \neq d \neq i$. F is the scaling factor controlling the amplification of the differential evolution.

The cross-over operator, implements a recombination of the trial vector and the parent vector to produce offspring. This operator is calculated as:

$$u_{j,i,G} = \begin{cases} v_{j,i,G}, (rand_j[0,1] \leq CR) or (j = j_{rand}) \\ x_{j,i,G}, otherwise \end{cases} \qquad (5)$$

Where $j = [1,\cdots,D]$; $rand_j \in [0,1]$; $j_{rand} = [1,\cdots,D]$ is the randomly chosen index ,CR is the crossover rate $v_{j,i,G}$ is the difference vector of the jth particle in the ith dimension at the Gth iteration, and $u_{j,i,G}$ denotes the trail vector of the jth particle in the ith dimension at the Gth iteration. Selection operator is used to choose the next population between the trail population and the target population:

$$x_{i,G+1} = \begin{cases} u_{i,G}, f(u_{i,G}) < f(x_{i,G}) \\ x_{i,G}, otherwise \end{cases} \tag{6}$$

The standard differential evolution algorithm can be described as the followings:

procedure *Algorithm description of DE algorithm*

begin

Step 1: *Set the generation counter G=0; and randomly initialize a population of NP individuals* X_i. *Initialize the parameter F, CR*

Step 2: *Evaluate the fitness for each individual in P.*

Step 3: while *stopping criteria is not satisfied* ***do***

  for *i= 1 to NP*

    select randomly $a \neq b \neq c \neq d \neq i$

     for *j=1 to D*

      $j_{rand} = \lfloor rand(0,1) * D \rfloor$

      If *rand(0,1)* $\leq$ *CR or j==* j_{rand} ***then***

        $u_{i,j} = x_{a,j} + F \times (x_{b,j} - x_{c,j})$

      Else

      $u_{i,j} = x_{i,j}$

      end if

     end for

  end for

  for i=1 to NP do

  Evaluate the offspring u_i

   If u_i is better than P_i **then**

    $P_i = u_i$

   end if

  end for

  Memorize the best solution achieved so far

Step 4: end while

end

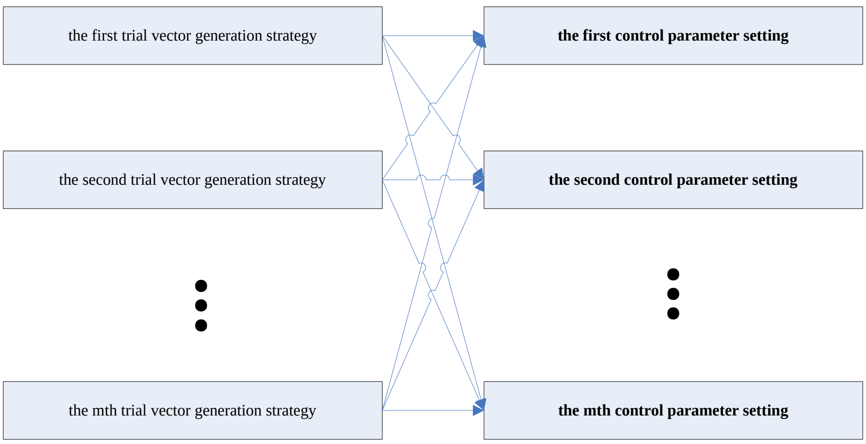

Fig. 2. Illustration of combining trial vector generation strategies with control parameter settings.

The three strategies and three parameter settings are frequently used in many DE variant and the properties have been discussed in [10]. At each generation, each trail vector in strategy candidate pool is used to create a new

2.3 Composite DE (CoDE)

Wang et al [10] proposed a new composite DE, CoDE, which is combining several effective trail vector generation strategies with some suitable control parameter settings in a random way to generate trail vectors. This algorithm has a simple structure and is easy to implement. This basic idea of the algorithm is to randomly combine several trail vector generation strategies with a number of control parameter settings at each generation to create new trail vector. The above idea is illustrated in Fig. 2. In the paper, the author choose three trail vector generation strategies and three control parameter settings to constitute the strategy candidate pool and the parameter candidate pool, respectively. The three selected trail generation strategies are:

(1) “rand/1/bin”

(2) “ rand/2/bin” and

(3) “current-to-rand/1”

Note that the “current-to-rand/1” strategy, the binominal crossover operator is not applied. The three control parameter settings are:

(1) [F=1.0, Cr=0.3]

(2) [F=1.0, Cr=0.9]

(3) [F=0.8, Cr=0.2]

trail vector with a control parameter setting randomly chosen from the parameter candidate pool. Then three trail vectors are generated for each target vector. The best ones enter the next generation if it better than its target vector. The pseudo code of CoDE is presented as follow:

procedure *Algorithm description of CoDE algorithm*
begin

Control parameter:

NP: the number of individuals at each generation.

Max_FES: maximum number of function evaluation evaluations.

The strategy candidate pool: "rand/1/bin", "rand/2/bin", "current-to-rand/1"

The parameter candidate pool: [F=1.0, Cr=0.3], [F=1.0, Cr=0.9], and [F=0.8, Cr=0.2]

Step 1) *Initialization*

Step 1.1) *Set the current generation number G=0;*

Step 1.2) *Generate an initialize population* $x_{1,0},\cdots,x_{NP,0}$ *by uniformly and randomly sampling from the feasible solution space.*

Step 1.3) *Evaluate the objective function values of these points*

Step 1.4) *FES=NP*

Step 2) *For* $i=1,\cdots,NP$ *,do*

Step 2.1) *use the three strategies, each with a control parameter setting randomly selected from the parameter pool, to generate three trail vectors* $u_{i\_1,G}$, $u_{i\_2,G}$, *and* $u_{i\_3,G}$ *for the target vector* $x_{i,G}$.

Step 2.2) *Evaluate the objective function values of three trail vectors* $u_{i\_1,G}$, $u_{i\_2,G}$, *and* $u_{i\_3,G}$;

Step 2.3) *Choose the best trail vector (denoted as* $u^*_{i,G}$*) from the three trail vectors* $u_{i\_1,G}$, $u_{i\_2,G}$, *and* $u_{i\_3,G}$

Step 2.4) *Selection and replacement:*

$$x_{i,G+1}=\begin{cases}u_{i,G}, f(u_{i,G})<f(x_{i,G})\\ x_{i,G}, otherwise\end{cases}$$

Step 2.5) *Set FES=FES+3*

Step 3) *If FES<= Max_FES, stop and output the vector with the small objective function value in the population, otherwise, set G=G+1 and go to Step 2.*
end

3. Experimental results

In this section, we will compare composite DE with other existing algorithm, a design problem that design of FBG for a specified bandwidth is considered. The problem is solved with unchirped and chirped grating. In order to evaluate the consistency of the CMAES algorithm [11], for each design 20 runs are conducted with different initial solutions in the beginning. The total number of function evaluations for each run is 5000. The reported computation time indicates the average of the time taken for the best solution in every run.

For each instance, the average running time on the 20 runs are recorded. The computational conditions are listed as follows.

System: Windows XP
* CPU: Intel(R) Core(TM) 2 Quad
* RAM: 1G
* Language: Matlab
* Compiler: Matlab 7.0

For the all algorithms, in order to comparison fair, the maximum fitness is 5000.

The FBG model [4] with 20 uniform sections is used for the symmetrical TFBG filter design. The index profile is bounded to take values between 0 to $5e^{-4}$. Hence, the optimum index profile obtained using the optimization method is always simple and can be fabricated using the method presented in [12]. In the all algorithm, the desired reflective power at the center wavelength is set at 90%. We obtained the CMAES Code from [11]. In order to compare the performance of CoDE on TFBG filter designs, the results of CMAES is considered.

3.1 Results

First, to illustrate the effectiveness of the proposed method, we consider to the design of TFBG filter with 0.2nm bandwidth and 40 mm grating length without chirp. Function Evaluations in order to make the comparison fair enough. The experimental results of 20 runs are listed in Table 1. As can be seen in Table1, we can find that the CoDE algorithm performs better than other algorithm. The best entries in Table 1 have been marked in bold. As the sum of squared errors is small, the calculated triangular spectrum edges are almost linear. Moreover, compared with the CMAES algorithm, the consistency of the CoDE is high in achieving good solutions. Even the mean solution obtained in CoDE is better than the best solution obtained by the CMAES algorithm. The modulation profile and reflection spectrum of the designed unchirped TFBG filter is shown in Fig. 3. The maximum reflective power obtained is 86.49. The computation time of the CoDE is less than the CMAES.

Table 1. Results of unchirped TFBG filter designed for a specified bandwidth (BW).

Algorithm	CMAES	CoDE
Best SSE	0.008080	0.007476164
Mean SSE	0.01133458	0.008285403
Worst SSE	0.0210133	0.0104393644
SD	0.00327815	8.352603e-004
Computation time	58.023	38.596

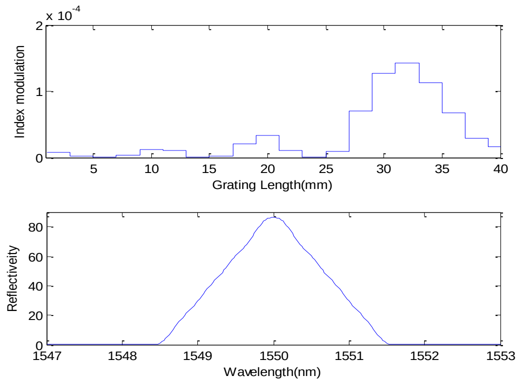

Fig. 3. Index modulation profile and reflection spectrum of the designed unchirped TFBG filter.

Fig. 4 shows the convergence characteristics of the CoDE and the CMAES algorithm. As can be seen in Fig. 4, the best fitness value is changed with respect to iterations. It can show the marked improvement in the performance of the CoDE over CMAES.

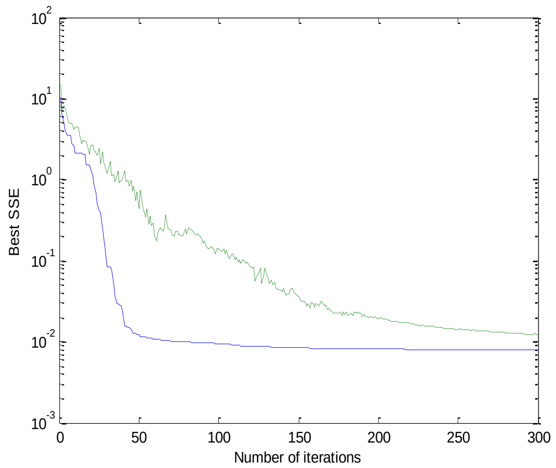

Fig. 4. Convergence characteristics of the CMAES and CoDE.

Second, In order to obtain smooth linear edges in the reflective spectrum, the index modulation is up-sampled 5times by using a linear interpolation [13]. After combining the up-sampling procedure, the number of uniform sections is 100, while the number of variables used in the optimization formulation is still 20. Table 2 shows the results of the chirped TFBG filter combining up-sampling. The index modulation profile and reflection spectrum of the designed chirped TFBG filter are shown in Fig. 5. For the same grating length, a larger bandwidth is obtained with chirped grating. Then, the edges of the reflective spectrum are linear because of the combination of the up-sampling scheme.

Table 2. Results of chirped TFBG filter designed for a specified bandwidth (BW) with uo-sampling.

Algorithm	CMAES	CoDE
Best SSE	0.04874119	**0.04864212**
Mean SSE	0.05048776	**0.04879409**
Worst SSE	0.061987544	**0.049201020**
SD	0.003924219	**0.000183**
Computation time	**72.63**	123.32

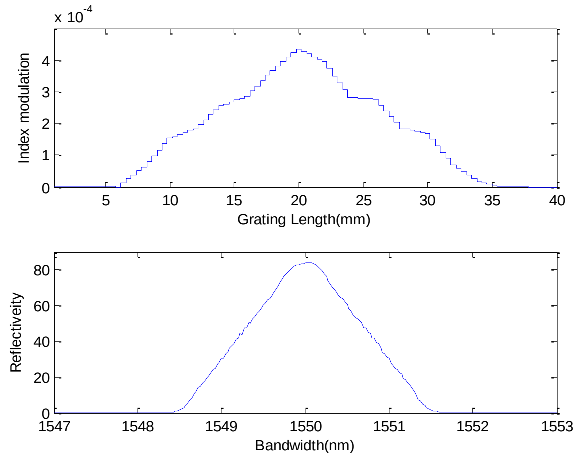

Fig. 5. Index modulation profile and reflection spectrum of the designed chirped TFBG filter with up-sampling.

4. Conclusions

This paper illustrated the application of composite differential Evolution algorithm called CoDE in designing triangular FBG filter fro sensor application. The effectiveness of the proposed algorithm is demonstrated on the design of triangular FBG filter based on a difficult instance. The results obtained using CoDE algorithm is compared with the results of CMAES. Comparisons show that the CoDE algorithm is more consistent in obtaining the best solution with reduced computation time. Hence, the CoDE algorithm is effective for the design of TFBG filter for sensor application.

In this paper, we only consider the triangular FBG. Our future work consists on other geometries and this algorithm will become a useful tool for a FBG designer.

Acknowledgments

This research is fully supported by the National Natural Science Foundation of China under Grant Nos. 60803102.

References

[1] S. Kim, S. Kim, J. Kwon, B. Lee, IEEE Photon. Technol. Lett. **13**, 839 (2001).
[2] R. Huang, Y. Zhou, H. Cai, R. Qu, Z. Fang, Optics Commun. **229**, 197 (2004).
[3] J. Skaar, K. M. Risvik, J. Lightwave Technol. **16**, 1928 (1998).
[4] C. L. Lee, Y. C. Lai, Optics Commun. **235**, 99 (2004).
[5] S. Baskar, R. T. Zheng, A. Alphones, N. Q. Ngo, P. N. Suganthan, IEEE Photonics Technol. Lett. **17**(3), 615 (2005).
[6] J. J. Liang, C. C. Chan, V. L. Huang, P. N. Suganthan, Optoelectron. Adv. Mater. – Rapid Commun. **1**(8), 373 (2007).
[7] J. J. Liang, P. N. Suganthan, C. C. Chan, V. L. Huang, IEEE Photon. Technol. Lett. **18**(12), 1305 (2006).
[8] M. K. Venu, R. Mallipeddi, P. N. Suganthan, Optoelectron. Adv. Mater. – Rapid Commun. **2**(11), 682 (2008).
[9] R. Storn, K. V. Price, J. of Global Optimization **11**, 341 (1997).
[10] Y. Wang, Z. X. Cai, Q. F. Zhang, IEEE Trans. Evol. Comput., **15**(1), 55 (2011).
[11] N. Hansen, "CMA-ES in MATLAB," Available from: http://www.bionik.tu-berlin.de/user/niko/cmaes_inmatlab.html.
[12] A. Asseh, H. Storoy, B. E. Sahlgren, S. Sandgren, R. A. H. Stubbe, J. Lightwave Technol. **15**, 1419 (1997).
[13] K. Aksnes, J. Skaar, Appl. Optics **43**(11), 2226 (2004).

*Corresponding author: Minghao.Yin1@gmail.com