DIFFERENTIAL EVOLUTION AND TABU SEARCH TO FIND MULTIPLE SOLUTIONS OF MULTIMODAL OPTIMIZATION PROBLEMS

Erick R. F. A. Schneider
UniversidadeFederal do Espírito Santo
Programa de Pos-Graduação em Informática, PPGI
Av. Fernando Ferrari, s/n. CEP 29060-270
Vitória, ES, Brazil
e-mail: erickrfas@gmail.com

Renato A. Krohling
Universidade Federal do Espírito Santo
Departamento deEngenharia de Produção &
Programa de Pos-Graduação em Informática, PPGI
Av. Fernando Ferrari, s/n. CEP 29060-270
Vitória, ES, Brazil
e-mail: krohling.renato@gmail.com

RESUMO

Muitos problemas de otimização da vida real são multimodais. Algoritmos Evolutivos tem sido aplicados a estes problemas com êxito, porém apresentam a desvantagem de convergirem para um único ótimo, mesmo existindo mais que uma solução ótima. Propomos assim um algoritmo híbrido combinando o algoritmo evolução diferencial (DE) e o algoritmo busca tabu (TS) para tratar esses problemas, permitindo achar mais de uma solução ótima. O algoritmo proposto foi testado em problemas de otimização com múltiplos ótimos e resultados comparados com aqueles fornecidos pelo algoritmo DE.

PALAVRAS CHAVE: problemas de otimização multimodais, evolução diferencial e busca tabu.

MH – Metaheuristica

ABSTRACT

Many real life optimization problems are multimodal. Evolutionary Algorithms (EA) have successfully been applied to solve these problems, but have the disadvantage that converge to only one optimum, even though there are many optima. We proposed a hybrid algorithm combining differential evolution (DE) with tabu search (TS) to solve these problems, allowing to find multiple solutions. The proposed algorithm was tested on optimization problems with multiple optima and the results compared with those provided by the DE algorithm.

KEY WORDS: multimodal optimization problems, differential evolution and tabu search.

MH – Metaheuristics

1. Introduction

Evolutionary Algorithms (EA) and Particle Swarm Optimization (PSO) techniques are effective and robust techniques for solving optimization problems (Eberhart and Shi, 2007). Typically, these algorithms converge to a single final solution. However, many real-world problems are multimodal in nature and may have many satisfactory solutions (Li, 2010). Niching methods have been developed in recent decades to find solution of multimodal optimization problems with multiple optima. These niching methods play an important role when incorporated into evolutionary algorithms to promote the diversity of the population and maintain multiple solutions within a stable population.

Several niching methods have been proposed in recent years, and the most relevant include (Li, 2010): fitness sharing, derating, restricted tournament selection, crowding, deterministic crowding, clustering, clearing, parallelization, speciation, among others. Most of the methods listed above present difficulties for solving multimodal optimization problems with multiple local or global optima because they need to specify the parameters of niching, which are difficult to tune and they generally are dependent on the problem to be optimized.

An interesting approach recently proposed by Li (2010), which does not require specification of any niching parameters uses a PSO with ring topology. In this case, the "local memory" of the individual particles of the PSO is able to maintain the best positions found so far, while the particles explore the search space. In the niching PSO without niching parameters proposed by Li (2010) it is shown that large populations using PSO with ring topology is capable of forming stable niche and able to find multiple local and global optima. The promising results suggest that this method present good results without requiring parameters to tune.

The method developed by Li (2010), motivated us to extend this method to Differential Evolution (DE). In fact, DE with ring topology (Das et al. 2009) and (Dorronsoro and Bouvry**,** 2011) were applied to multimodal problems with promising results to find a single solution of optimization problems. However, for problems with multiple optima, as far as we know, DE with ring topology has not been used yet. The DE version proposed by Das et al. (2009) called DEGL has a good capacity of exploration/exploitation of the search space.

In case of PSO with ring topology, the personal best positions *pbest* of all particles in the population are used to form the memory of the swarm and retain the best solutions found so far in the population, and the positions of the particles act to explore the search space. In this work, inspired by the PSO with ring topology, the DEGL is used to explore the search space, but as DEGL has no memory, this motivated us to combine the algorithm withTabu Search (TS) to fulfill the function of memory. In this context, the hybrid method, which consists of a genetic algorithm (GA) embedded with a niching method and Tabu search has been proposed by Li et al. (2010). The main disadvantage of that method is that is necessary to specify the niching parameters. So, in order to overcome this problem, we propose the DEGL with ring topology combined with Tabu search. As in (Li et al. 2010), the method proposed here also consists of two stages. The first stage uses DEGL to explore the search space. In the second stage, the TS takes the initial solutions provided by DEGL in the first stage and performs a local search aiming to improve the population of solutions of the DEGL. At the end of the algorithm run, a population containing local/global solutions is provided.

2. Differential Evolution

In the following, the standard Differential Evolution and the Differential Evolution with ring topology are described.

2.1. Standard Differential Evolution

Differential Evolution (DE) is an optimization method introduced by Storn and Price (1995). Similar to other Evolutionary algorithms (EAs), it is based on the idea of evolution of populations of possible candidate solutions, which undergoes the operations of mutation, crossover and selection (Storn and Price, 1997). The candidate solutions of the optimization problem in DE are represented by vectors. The components of the vectors are the parameters of the optimization problem and the set of vectors make up the population. Unless stated otherwise in our study, we are considering minimization problems. Therefore, the higher the value of fitness, the smaller the values of the objective function.

Consider a population of NP individuals in an N-dimensional search space. The individuals in the population are initialized according to:

$$x_{i,j} = x_{j,min} + \mathrm{U}(0,1)_{i,j}(x_{j,\max} - x_{j,min}) \quad (1)$$

where j is the index of the j-th component of the i-th individual of the population, $x_{j,max}$ and $x_{j,min}$ are the upper and lower bounds of the j-th component of the N-dimensional vector $\vec{X}$, respectively, and U(0,1) is a random number drawn from a uniform probability distribution.

By means of the mutation operator, a new vector $\vec{V}_i$ is generated by the following equation:

$$\vec{V}_i = \vec{X}_{r1,i} + F(\vec{X}_{r2,i} - \vec{X}_{r3,i}) \quad (2)$$

where F is a scaling factor, and $r_1, r_2, r_3 \in [1, \mathrm{NP}]$, such that $r_1 \neq r_2 \neq r_3$.

In the crossover operation, a new vector $\vec{U}_i$ is generated according to:

$$u_{i,j,G} = \begin{cases} v_{i,j,G} & \text{if } \mathrm{U}(0,1)_{i,j} \leq C_r \text{ or } j = j_{rand} \\ x_{i,j,G} & \text{otherwise} \end{cases} \quad (3)$$

where C_r is the crossover rate and j_{rand} is a random component of each individual to ensure that at least one component of the vector $\vec{V}_i$ is part of the new vector.

In the selection, the individual is chosen according to:

$$\vec{X}_{i,G+1} = \begin{cases} \vec{U}_{i,G} & \text{if } f(\vec{U}_{i,G}) \leq f(\vec{X}_{i,G}) \\ \vec{X}_{i,G} & \text{otherwise} \end{cases} \quad (4)$$

where *f*(.) is the objective function.

2.2. Differential Evolution with ring topology

One disadvantage of standard DE may be the premature convergence to local optima in multimodal functions, losing the diversity of the population. To increase the diversity of the population, it is used a neighborhood with ring topology as shown in Fig. 1. So, the mutation operator is implemented through the creation of two vectors before the generation of the vector $\vec{V}_i$.

First, the vector is calculated as:

$$\vec{L}_{i,G} = \vec{X}_{i,G} + \alpha\left(\vec{X}_{n_{best_i}} - \vec{X}_{i,G}\right) + \beta(\vec{X}_{p,G} - \vec{X}_{q,G}) \quad (5)$$

where n_{best_i} indicates the best vector in the neighborhood of $\vec{X}_{i,G}$. *p*, *q* are indices in the neighborhood chosen such that $p \neq q \neq i$, and $p, q \in [i-k, i+k]$, where k is the length of the neighborhood.

Next, the vector $\vec{g}_i$ is calculated as:

$$\vec{g}_{i,G} = \vec{X}_{i,G} + \alpha(\vec{X}_{gbest} - \vec{X}_{i,G}) + \beta(\vec{X}_{r_1,G} - \vec{X}_{r_2,G}) \quad (6)$$

where *gbest* indicates the best of the entire population, r_1 and r_2 are such that $r_1 \neq r_2 \neq i$, and $r_1, r_2 \in [1, \mathrm{NP}]$.

Then, one calculates the vector $\vec{V}_i$ as follows:

$$\vec{V}_{i,G} = w\vec{g}_{i,G} + (1-w)\vec{L}_{i,G} \quad (7)$$

where w is a weighting factor. Small values of w increase the diversity of the population. An illustration of the ring topology is shown in Fig. 1, where the length of the neighborhood is k = 2 and the number of individuals is NP.The DE algorithm pseudo-code of DE with ring topology is shown in Fig. 2.

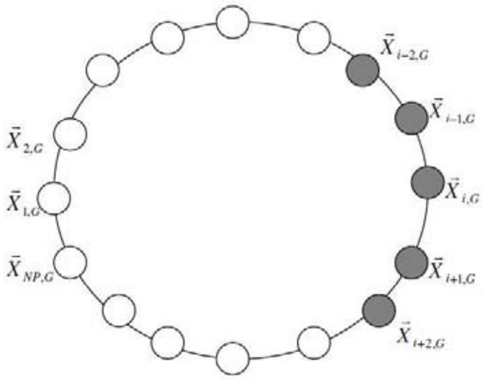

Fig.1. Illustration of the ring topology.

Input: Population size NP
for i= 1 to NP
for j = 1 to N
$$x_{i,j} = x_{j,min} + \mathrm{U}(0,1)_{i,j}(x_{j,max} - x_{j,min})$$
end for
end for
while termination condition not met
for i=1 to NP
choice p and q such that $p \neq q \neq i$ from $[i-k, i+k]$
$$\vec{L}_{i,G} = \vec{X}_{i,G} + \alpha\left(\vec{X}_{n_{best_i}} - \vec{X}_{i,G}\right) + \beta\left(\vec{X}_{p,G} - \vec{X}_{q,G}\right)$$
choice r_1 and r_2 such that $r_1 \neq r_2 \neq i$ from [1,NP]
$$\vec{g}_{i,G} = \vec{X}_{i,G} + \alpha\left(\vec{X}_{gbest} - \vec{X}_{i,G}\right) + \beta\left(\vec{X}_{r_1,G} - \vec{X}_{r_2,G}\right)$$
$$\vec{V}_{i,G} = w\vec{g}_{i,G} + (1-w)\vec{L}_{i,G}$$
$$j_{rand} = \mathrm{U}(1,\mathrm{N})$$
for j = 1 to N
if $\mathrm{U}(0,1)_{i,j} \leq C_r$ or $j = j_{rand}$
$$u_{i,j} = v_{i,j}$$
Else
$$u_{i,j} = x_{i,j}$$
end if
end for
if $f(\vec{U}_i) \leq f(\vec{X}_i)$
$$\vec{X}_i = \vec{U}_i$$
end if
end for
end while

Fig. 2. Pseudo-code of the DE algorithm with ring topology.

3. Proposed method

In recent years, several approaches combining EA with local search methods have been proposed (Chelouah and Siarry, 2005), (Li, 2010), (Li et al., 2010), (Mashinchi et al., 2011) and (Wei and Zhao, 2005). In order to solve the optimization problem with multiple optima we propose a hybrid algorithm based on DE and TS. In our approach, the DE algorithm is used for global search in order to increase the diversity of the population. Then, the Tabu Search algorithm is used for local search to refine the search around promising solutions. The pseudo-code of Tabu Search algorithm is presented in Fig. 3.
In order to generate the neighborhood two parameters are used: the number of neighbors (nn) and the step size Δ. So, the neighbors are generated according to:

$$s'_{i,j} = s_{i,j} + \mathrm{U}(-1,1)_{i,j}\Delta \qquad (8)$$

where U(-1,1) is a random number drawn from a uniform probability distribution.
To verify if the best neighbor is tabu, it is necessary to perform the comparison:

$$\left(s'_j \geq t_{i,j} - \varepsilon\right) \text{ or } \left(s'_j \leq t_{i,j} + \varepsilon\right) \qquad (9)$$

where s'_j is the *j*-th component of the best neighbor, $t_{i,j}$ is *j*-th component of the *i*-th element of the tabu list and ε is a parameter of the algorithm.
The tabu list has a fixed length (L). So, elements are inserted into the head of the list, so that the last element is deleted when an element is inserted.

```
Input: population size NP, step size Δ and number of neighbors nn
for i= 1 to NP
        s = V_i
        best = s
        T = Ø
while stopping condition not reached
s' = best neighbor of s that is not tabu
if f(best) > f(s')
        best = s'
end if
        s = s'
    update tabu list T with s'
end while
  return best
end for
```

Fig. 3. Pseudo-code of Tabu Search algorithm.

4. Simulation Results

4.1. Experimental Settings

The DE algorithm with ring topology has been tested on four benchmark functions as described in Table 1. Fig. 4 illustrates the multimodality of the benchmarks. The algorithm parameters used in the experiments are presented in Table 2. The algorithm DE with ring topology has been tested on four different population size. Next, we tested the performance of the proposed method combining

DE with ring topology and Tabu Search. In order to compare our results, we have also carried out the experiments with PSO with ring topologies:

- PSO_r3: ring topology with two neighbors for each individual.
- PSO_r3_lhc: similar to the first, but without overlapping neighborhoods.
- PSO_r2: ring topology with one neighbor for each individual.
- PSO_r2_lhc: similar to the third, but without overlapping neighborhoods.

The experiments are run for 2000 iterations. For the proposed algorithm DE+TS after 2000 iterations of DE are performed more 100 iterations for TS. The results are presented in Tables 3 and 4.

Table 1: Benchmarks functions (Li, 2010).

$F_1(x) = \sin^6 5\pi x$	$0 \leq x \leq 1$
$F_2(x) = e^{-2\log 2(\frac{x-0.1}{0.8})} \sin^6 5\pi x$	$0 \leq x \leq 1$
$F_3(x) = \sin^6 5\pi \left(x^{\frac{3}{4}} - 0.05\right)$	$0 \leq x \leq 1$
$F_4(x) = e^{-2\log 2(\frac{x-0.08}{0.854})} \sin^6 5\pi \left(x^{\frac{3}{4}} - 0.05\right)$	$0 \leq x \leq 1$

Table 2: Algorithm parameters used in DE and DE+TS.

C_r	0.9
α	0.8
β	0.8
w	0.1
L	10
Δ	0.1
nn	10
ε	0.01

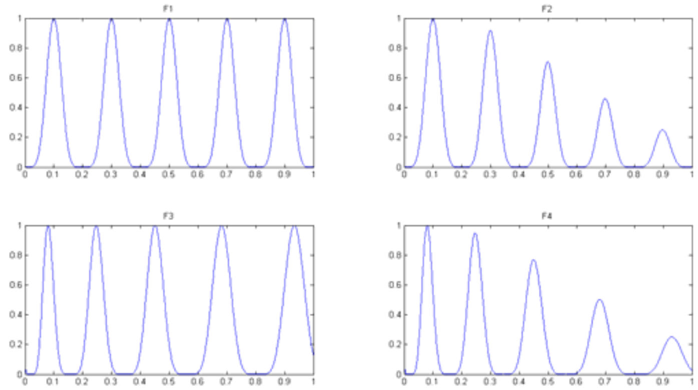

Fig. 4. Plot of the benchmarks functions used in the experiments.

4.2. Results Discussion

From Table 3, for a population size of 51 individuals, the hybrid algorithm presents better performance than the DE algorithm in 3 out of 4 benchmarks. Increasing the population size, we noticed that the DE algorithm and the DE+TS present very similar performance. The DE algorithm with population size 201 shows a superior performance on 2 out of 4 benchmarks. The cause of this may occur because TS refines the previous found solutions (local optima) allowing to reach the global ones.

Table 3: Success rate in terms of mean and standard deviation for the algorithms DE and DE + TS.

Function	NP	DE	DE + TS
F1	51	94,40% (10,61%)	**98,00% (6,00%)**
F2	51	**80,00% (13,86%)**	71,20% (13,95%)
F3	51	92,40% (10,50%)	**98,40% (5,43%)**
F4	51	81,20% (14,09%)	**82,40% (5,43%)**
F1	102	**99,60% (2,80%)**	**99,60% (2,80%)**
F2	102	**90,80% (10,74%)**	78,00% (10,77%)
F3	102	99,60% (2,80%)	**100%**
F4	102	**91,20% (10,70%)**	90,00% (10,00%)
F2	201	**100%**	**100%**
F3	201	**91,60% (9,87%)**	87,60% (9,71%)
F4	201	**100%**	**100%**
F5	201	**97,60% (6,50%)**	96,40% (7,68%)

Among the four topologies tested for the PSO, the one that found the best results was that with the topology r3_lhc. For a population size of 51, the performance of PSO with topology r3_lhc is better than DE and DE+TS in 3 out of 4 benchmarks. Increasing the population size to 201 the performance of DE and DE+TS is getting closer to PSO r3_lhc. In this case, the performance is the same for two benchmarks, and for the other two benchmarks PSO r3_lhc beats DE and DE+TS.

Table 4: Success rate in terms of mean and standard deviation for the algorithm PSO.

Function	NP	r3	r3_lhc	r2	r2_lhc
F1	51	98,00% (7,21%)	**99,20% (3,92%)**	**99,20% (3,92%)**	**99,20% (3,92%)**
F2	51	20,00% (0%)	**80,40% (12,96%)**	20,00% (0%)	77,20% (17,44%)
F3	51	92,80% (11,14%)	**96,80% (7,33%)**	96,40% (7,68%)	97,60% (6,50%)
F4	51	20,00% (0%)	**91,20% (11,43%)**	20,00% (0%)	87,60% (14,91%)
F1	102	99,60% (2,80%)	**100% (0%)**	**100% (0%)**	**100% (0%)**
F2	102	20,00% (0%)	**94,80% (8,77%)**	20,00% (0%)	89,60% (13,41%)
F3	102	98,80% (4,75%)	**100% (0%)**	99,60 (2,80%)	99,60% (2,80%)
F4	102	21,60% (11,20%)	**98,40% (5,43%)**	23,20% (15,68%)	**98,40% (5,43%)**
F2	201	**100% (0%)**	**100% (0%)**	**100% (0%)**	**100% (0%)**
F3	201	21,60% (11,20%)	**98,40% (5,43%)**	24,80% (19,00%)	96,80% (7,33%)
F4	201	99,60% (2,80%)	**100% (0%)**	**100% (0%)**	**100% (0%)**
F5	201	26,40% (21,70%)	99,60% (2,80%)	31,20% (27,76%)	**100% (0%)**

5. Conclusions

In this paper, we propose a DE with ring topology without overlapping neighborhood combined with Tabu Search. The results of the DE+TS have been compared to DE for 3 different population sizes. Increasing the population size, we noticed that DE provided better performance. This may be caused by the fact that DE+TS refines the solution found by DE, resulting in a loss of diversity of the population. We compare our results with a state of the art PSO with 4 different ring topologies (r2, r2_lhc, r3, r3_lhc). The best results using PSO with ring topologies was obtained with the topology r3_lhc. Increasing the population size of DE, the performance of the algorithm is getting closer to PSO r3_lhc. The next step of this research is to investigate the use of other local search methods, e.g., Nelder-Mead, or Hooke and Jeves for a larger suite of benchmark functions.

References

R. Chelouah, P. Siarry (2005). A hybrid method combining continuous tabu search and Nelder-Mead simplex algorithms for the global optimization of multiminima functions, *European Journal of Operational Research*, 161, 636-654.

S. Das, A. Abraham, U. K. Chakraborty, A. Konar (2009). Differential evolution using a neighborhoodbased mutation operator, *IEEE Transactions on Evolutionary Computation*, 13, 526-553.

B. Dorronsoro, P. Bouvry (2011). Improving classical and decentralized differential evolution with new mutation operator and population topologies, *IEEE Transactions on Evolutionary Computation*, 15, 67-98.

R. C. Eberhart and Y. Shi. Computational intelligence - concepts to implementations. Morgan Kaufmann, 2007.

X. Li (2010). Niching without niching parameters: particle swarm optimization using a ring topology, *IEEE Transactions on Evolutionary Computation*, 14, 150-169.

Z. Li, H. Li, Y. Chen, and A. Sallam (2010). A hybrid strategy based on niche genetic algorithm and tabu search and its convergence property. *IEEE Fifth International Conference on Bio-Inspired computing: Theories and Applications(BIC-TA)*, pp. 328-332.

M. H. Mashinchi, M. A. Orgun, and W. Pedrycz (2011). Hybrid optimization with improved tabu search. *Applied Soft Computing,*11, 1993-2006.

R. Storn and K. V. Price (1995). Differential evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces, *Technical Report TR-95-012, ICSI.*

R. Storn and K. Price (1997). Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11, 341-359.

L. Wei, M. Zhao (2005). A niche hybrid genetic algorithm for global optimization of continuous multimodal functions, *Applied Mathematics and Computation*, 649-661.