A New Collaborative Evolutionary-Swarm Optimization Technique

Rodica Ioana Lung
Babes-Bolyai University
Department of Computer Science
Cluj Napoca, Romania
srodica@cs.ubbcluj.ro

D. Dumitrescu
Babes-Bolyai University
Department of Computer Science
Cluj Napoca, Romania
ddumitr@cs.ubbcluj.ro

ABSTRACT

A new hybrid approach to optimization in dynamical environments called Collaborative Evolutionary-Swarm Optimization (CESO) is presented. CESO tracks moving optima in a dynamical environment by combining the search abilities of an evolutionary algorithm for multimodal optimization and a particle swarm optimization algorithm. A collaborative mechanism between the two methods is proposed by which the diversity provided by the multimodal technique is transmitted to the particle swarm in order to prevent its premature convergence. Numerical experiments indicate CESO as an efficient method compared with other evolutionary approaches.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods and Search—*Heuristic Methods*; G.1.6 [**Numerical Analysis**]: Optimization global optimization

General Terms

Algorithms

Keywords

Crowding Differential Evolution, Particle Swarm Optimization, Dynamic Environments

1. INTRODUCTION

One of the challenges presented by real-world applications is their dynamical character. The problem of detecting and tracking moving optima in a dynamical environment has been successfully addressed by evolutionary algorithms during the last years [3].

A new approach to solving optimization problems in dynamical environments called Collaborative Evolutionary - Swarm Optimization (CESO) is proposed.

GECCO'07, July 7–11, 2007, London, England, United Kingdom.

CESO is based on the collaboration between two optimization methods: an evolutionary algorithm for multimodal optimization and a particle swarm optimization algorithm. The evolutionary multimodal optimization algorithm provides a diversity preservation mechanism preventing the particle swarm's premature convergence to local optima.

2. COLLABORATIVE EVOLUTIONARY - SWARM OPTIMIZATION

CESO algorithm is a simple method for detecting and tracking moving optima in a changing environment by using two populations of equal size. One of the population is responsible for preserving diversity of the search and the other one tracks the global optimum. CESO proposes a collaborative mechanism between the two populations in order to avoid premature convergence and to efficiently track moving optima.

2.1 CESO populations

The rules by which the two populations used by CESO are evolved are described in what follows.

The CRDE population.

The first population called the CRDE population is evolved by an evolutionary multimodal optimization algorithm in order to maintain a good population diversity.

Evolutionary multimodal optimization techniques have already been applied for solving optimization problems in dynamical environments: the Self Organizing Scouts [2] uses a forking procedure borrowed from the Forking GA [9] while Multinational GA [10] was originally designed as an algorithm for multimodal optimization in static environments.

As a multimodal search operator CESO uses the Crowding Differential Evolution (Crowding DE)[8] algorithm, a very efficient method for detecting multiple optima in static environments.

Crowding Differential Evolution extends the Differential Evolution (DE) algorithm [6] with a crowding scheme. The only modification to the conventional DE is made regarding the individual (parent) being replaced. Usually, the parent producing the offspring is substituted, whereas in CrowdingDE the offspring replaces the most similar individual among the population (if it is fitter). A *DE/rand/1/exp* [7] scheme is used.

The best individual in the CRDE population is denoted $cbest$.

The SWARM.

The second population used by CESO, called SWARM, is a particle swarm updated by using classical particle swarm optimization (PSO) rules [4]. Particle swarm optimization (PSO) is a population based stochastic optimization technique inspired by social behavior of bird flocking or fish schooling.

Within PSO each individual has a velocity vector associated. Each iteration, the following equations are used to compute the new position of individual $x = (x_1, ..., x_n)$:

$$v_{i+1} = v_i + c_1 * rand * (pbest_i - x_i) + c_2 * rand * (gbest_i - x_i),$$

$$x_{i+1} = x_i + v_i,$$

where $v = (v_1, ..., v_n)$ is called the velocity of particle x ; *pbest* represents the best position of individual x so far, *gbest* represents the best individual in the whole population detected so far; *rand* is a random number between (0,1) and c_1, c_2 are learning factors. Usually $c_1 = c_2 = 2$. Two parameters, *vmin* and *vmax* are used to limit the velocity.

Endowing PSO with an efficient diversity preserving mechanism it becomes a very powerful optimization technique.

2.2 The Collaboration

One of the main problem in dealing with dynamical environments is the premature convergence to local optima. To cope with it CESO uses the CRDE population to maintain a set of local and global optima during the entire search process. The SWARM population is used to detect the global optimum and to indicate - if necessary - its position to the CRDE population.

Both CRDE and SWARM populations evolve in their 'natural' manner, i.e. no additional mechanism is added to them individually.

The collaborative mechanism proposed by CESO implies two-ways communication between the SWARM and CRDE:

Transmitting information from the CRDE to the SWARM population.

The CRDE population maintains a good diversity over the search space by maintaining a set of local optimal solutions. CRDE information is transmitted to the SWARM by copying all individuals from the CRDE to the SWARM. Thus the SWARM is actually reinitialized. The reinitialization of the SWARM takes place if one of the followings occur:

i. a change is detected in the environment (the test is made by re-evaluating *cbest*); in this case all individuals are evaluated;

ii. the distance between *cbest* and *gbest* is lower than a prescribed threshold θ (for example 0.1)

Transmitting information from the SWARM to CRDE population.

At each iteration *gbest* replaces *cbest* if it has a better fitness value. Thus the CRDE population contains the best optima detected at each iteration.

3. CESO ALGORITHM

Within CESO both populations are randomly initialized and evaluated. At the beginning of each iteration a test is performed to check if a change in the environment has occurred during the last iteration. The best individual in the CRDE population is re-evaluated: if a difference appears between the new and the old fitness value it is considered that a change took place.

In case a change appears in the environment or if the distance between *cbest* and *gbest* is very small, the search of the SWARM is restarted by copying the CRDE individuals to it. By re-starting the search with particles scattered over the search space, the SWARM presents a good potential to locate the global optimum.

At the end of each iteration *gbest* replaces *cbest* if it is better than *cbest*. Therefore, at each iteration, the CRDE contains the best individual found so far.

CESO technique is outlined in the Algorithm 1.

Algorithm 1 Outline of the CESO Algorithm

```
Parameters setting;
Randomly initialize CRDE and SWARM;
Evaluate populations;
while final condition not met do
    if (change in landscape) then
        Copy CRDE to SWARM;
        Evaluate populations;
    end if
    if distance between best individuals in CRDE and
SWARM lower than 1 then
        Copy CRDE to SWARM;
    end if
    Update SWARM;
    Evolve CRDE;
    Evaluate populations;
    if gbest better than cbest then
        gbest replaces cbest in CRDE;
    end if
end while
```

4. NUMERICAL EXPERIMENTS

Numerical experiments concerning the Moving peaks bechmark (MPB), scenario2, as proposed by Branke in [2] were performed. This scenario has also been used by several authors and allows the comparison of results obtained by different methods. The settings for this scenario are presented in Table 1.

Results obtained by CESO are compared with those reported by the following methods:

- the Self Organizing Scouts (SOS) [2];
- the Multiswarms (MPSO) methods [1];
- The Particle Swarm with Speciation and Adaptation (SPSO) [5].

The best results obtained using the three methods considered, where applicable, are compared with those obtained by CESO.

MPSO and SPSO may have several configurations and variants. The best reported results have been chosen from the various configurations in order to compare them with those obtained by CESO.

Results are averaged over 50 runs with different random seed generator for CESO.

Table 1: Standard settings for the Moving Peaks Problem

Parameter	Setting
Number of peaks p	10
Number of dimensions d	5
Peak heights	$\in [30, 70]$
Peak widths	$\in [1, 12]$
No. of evals. between changes	5000
Change severity s	1.0
Correlation coefficient λ	0

Table 2: Parameter Settings for CESO

Parameter	Setting
CRDE and SWARM sizes	10
vmin,*vmax*	-0.1,0.1

4.1 Parameter settings for CESO

CESO uses only three parameters: the populations size and *vmin* and *vmax* to control the particles velocities. For the other parameters specific to Crowding DE and PSO the usual values are used.

Parameters setting for CESO is presented in Table 2. Numerical results indicate that using small CRDE and SWARM population size, CESO results are significantly better than those obtained by other methods as far as the average offline error is concerned.

4.2 Varying shift severity

Results obtained by CESO for different values of the shift severity parameter s of the MPB are presented in Table 3.

The Table 3 presents results reported by the multi-CPSO (mCPSO) $10(5+5^q)$ [1] which, to the best of our knowledge, are the best results reported until now for this problem. The SPSO-PD variant [5] reports an average of offline errors of 1.93(0.06) for $s = 1$. For this setting, CESO reports an average offline error of 1.38. Results obtained with CESO are better than those in terms of average and standard error values.

Table 3: Offline error and standard error for varying shift severity

s	CESO	mCPSO
0	**0.85± 0.02**	1.18± 0.07
1	**1.38± 0.02**	1.75± 0.06
2	**1.78± 0.02**	2.40± 0.06
3	**2.03± 0.03**	3.00± 0.06
4	**2.23± 0.05**	3.59± 0.10
5	**2.52± 0.06**	4.24± 0.10
6	**2.74± 0.10**	4.79± 0.10

Table 4: Offline error and standard error for varying number of peaks

No. peaks	CESO	MPSO
1	**1.04± 0.00**	4.93± 0.07
10	**1.38± 0.02**	1.75± 0.06
20	**1.72± 0.02**	2.42± 0.06
30	**1.24± 0.01**	2.48± 0.06
40	**1.30± 0.02**	2.55± 0.10
50	**1.45± 0.01**	2.50± 0.10
100	**1.28± 0.02**	2.36± 0.10

Table 5: Offline error and standard error for varying the λ parameter

λ	CESO	SOS
0.5	**1.43± 0.02**	4.14
0.9	**1.46± 0.03**	4.09
1	**1.52± 0.02**	4.17

4.3 Varying number of peaks

Results obtained for different number of peaks are presented in Table 4. For MPSO the best results have been obtained for mCPSO with anticonvergence for the one peak set-up, mQSO without anticonvergence for the 10 peaks set-up and for mQSO with anticonvergence for the rest of set-ups. Results obtained by SOS and SPSO-PD are not better than those obtained by the MPSO.

4.4 Correlation of shifts

The effects on CESO of changing the MPB correlation of shifts parameter λ are presented in Table 5. Results are compared with average values reported by SOS. Results provided by CESO are significantly better than those reported by SOS for all values of λ.

4.5 Higher dimensionality

Numerical results for different dimensionality values for MPB are presented in Table 6.

For dimension ten, mQSO variant of MPSO reports results in the range between 4.17 and 4.70 for different parameter settings of the algorithm. For the 10-dimensions instance of MPB average of offline errors reported by CESO is 2.51, which is significantly better than those obtained by mQSO.

A modified version of SOS reports an average offline error of 16.2 for a 20-dimensions search space and 20 peaks. For the same settings, the average offline error reported by CESO is 2.53, i.e. 15.61% of the result obtained by SOS.

4.6 Effect of the collaboration mechanism

The effect of the collaboration mechanism between the two population can be illustrated by running Crowding DE and PSO independently. Average offline error after 50 runs, for the standard setting of scenario 2 of the MPB for the three methods are presented in table 7. Results indicate that the proposed collaborative mechanism can be consid-

Table 6: **Offline error and standard error for varying dimension of the search space**

no. dimensions	CESO
10	2.51± 0.04
50	6.81± 0.07
100	24.60± 0.25

Table 7: Offline error and standard error for CESO and for the Crowding DE and PSO without any collaboration

Method	Value
CESO	**1.38± 0.02**
Crowding DE	3.98± 0.14
PSO	24.23± 1.30

ered responsible for the results obtained by CESO and that the results obtained by using only Crowding DE or only the PSO without any collaboration are not as good.

5. CONCLUSIONS AND FURTHER WORK

A new evolutionary method for solving optimization problems in dynamical environments is proposed.

The Collaborative Evolutionary-Swarm Optimization algorithm combines the abilities of the Crowding Differential Evolution algorithm for multimodal optimization and of the Particle Swarm Optimization by using a collaboration mechanism in order to detect and track optima in a changing environment.

Two populations of individuals are evolved by CESO. The CRDE population uses the Crowding Differential Evolution algorithm to detect and to maintain a set of approximation of local optima while the SWARM population follows the rules of Particle Swarm Optimization to track the global optima.

The collaboration mechanism is applied whenever a change is detected in the search space, or if the best individual in the SWARM is too close to the best in CRDE the search of the SWARM is restarted by re-initializing the SWARM with the positions of individuals in the CRDE population. Due to the diversity offered by these individuals, the SWARM population is capable to locate the global optimum. If the SWARM locates an optimum better than the best individual in the CRDE population then replaces *cbest* in order to enhance the search of the CRDE.

Numerical results obtained by CESO are significantly better in terms of average offline error than those obtained by other evolutionary techniques for optimization in dynamical environments as reported in the literature. However, the sensitivity to parameter settings of CESO still has to be studied and represents the object of current research.

6. REFERENCES

[1] T Blackwell and J. Branke. Multiswarms, exclusion, and anti-convergence in dynamic environments. *Evolutionary Computation, IEEE Transactions on*, 10, issue: 4, 2006.

[2] Jürgen Branke. *Evolutionary Optimization in Dynamic Environments*. Klüwer Academic Publishers, 2001.

[3] Yaochu Jin and Jürgen Branke. Evolutionary optimization in uncertain environments-a survey. *IEEE Trans. Evolutionary Computation*, 9(3):303–317, 2005.

[4] James Kennedy and Russell C. Eberhart. *Swarm intelligence*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001.

[5] Xiaodong Li, Jürgen Branke, and Tim Blackwell. Particle swarm with speciation and adaptation in a dynamic environment. In *GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pages 51–58, New York, NY, USA, 2006. ACM Press.

[6] Rainer Storn and Kenneth Price. Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces. Technical Report TR-95-012, Berkeley, CA, 1995.

[7] Rainer Storn and Kenneth Price. Differential evolution a simple evolution strategy for fast optimization. *Dr. Dobb's Journal of Software Tools,*, 22(4):18–24, 1997.

[8] Rene Thomsen. Multimodal optimization using crowding-based differential evolution. In *Proceedings of the 2004 IEEE Congress on Evolutionary Computation*, pages 1382–1389, Portland, Oregon, 20-23 June 2004. IEEE Press.

[9] S. Tsutsui, Y. Fujimoto, and A. Gosh. Forking genetic algorithms: GAs with search space division. *Evolutionary computation*, 5:61–80, 1997.

[10] Rasmus K. Ursem. Multinational GAs: Multimodal optimization techniques in dynamic environments. In *Proceedings of the Second Genetic and Evolutionary Computation Conference (GECCO-2000)*, volume 1, pages 19–26, Riviera Hotel, Las Vegas, USA, 2000. Morgan Kauffmann Publishers.