Competing Heuristics in Evolutionary Algorithms

Josef Tvrđík, Ladislav Mišík, and Ivan Krivy
University of Ostrava, 701 03 Ostrava, Czech Republic
tvrdik@osu.cz, Ladislav.Misik@osu.cz, krivy@osu.cz

Abstract. The paper deals with a class of evolutionary algorithms (EAs) for the global optimization. The special attention is paid to the controlled random search (CRS). A generalization of the EA is proposed with several heuristics competing with each other when generating new trial points. The condition for asymptotic convergence of the algorithm are briefly discussed. Two instances of the EA with competing heuristics were implemented and the experimental results obtained on several test functions are presented.

1 Introduction

We deal with a class of evolutionary algorithms defined as follows. Let \( D \subset \mathbb{R}^d \).
Denote by \( \mathcal{L} \) the system of all Lebesgue measurable subsets of \( D \) and \( \lambda \) the Lebesgue measure on \( \mathcal{L} \). Let \( f \) be a real Lebesgue measurable function defined on \( D \). The number \( \mu = \inf \{ t; \lambda(f^{-1}(-\infty, t)) > 0 \} \), where \( f^{-1}(A) = \{ x \in D; f(x) \in A \} \), is called the essential minimum of \( f \). The task is to find an arbitrary good approximation of \( \mu \). Let \( \mu_0 \) be a probability measure on \((D, \mathcal{L})\) which is positive on each open subset of \( D \). Let \( N \) be a positive integer called the size of population and \( D^N \) the set of all populations. Let \( \pi \) be a mapping defined on \( D^N \) assigning to each population \( P \) the probability measure \( \pi(P) \) on \((D, \mathcal{L})\). Finally, let \( \{ m_k \} \) be a sequence of numbers from interval \((0, 1)\) and let \( C \) be a rule according to which some points in the old population are replaced by new ones. We consider a class of evolutionary algorithms described as follows:

\( I \): Generate an initial population \( P_0 = \{ x_1, x_2, \ldots, x_N \} \) chosen as an independent identically distributed sample according to the probability measure \( \mu_0 \) and set \( k = 0 \).

\( C_k \): Copy a portion of \( M \) best points of \( P_k \) directly into the new population. Here the best points are those with the lowest values of \( f \) and \( M \) is an integer from \( \{1, 2, \ldots, N - 1\} \).

\( S_k \): Select a new point at random according to the probability measure \( \mu_{k+1} = \pi(P_k) \) and include it to the new population when fulfilling the condition \( C \). Repeat this procedure until the new population is complete.

\( M_k \): With the probability \( m_{k+1} \) replace a randomly chosen point by its mutation.

\( R \): Set \( k = k + 1 \) and go back to \( C_k \).
Mutation of a given point is a point in $D$ generated according to specified rules. Starting from the results [4, 6] the sufficient condition for the algorithm convergence is as follows: For any measurable subset $S \subset D$ denote by $p_k(S)$ the probability that the algorithm produces in the $k$-th step a new trial point belonging to the set $S$. Then the algorithm is convergent provided that for every set $S$ such that $\lambda(S) > 0$ we have $\prod_{k=1}^{\infty} (1 - p_k(S)) = 0$.

2 Evolutionary Algorithm with Competing Heuristics

We consider an evolutionary algorithm without explicit mutation:

```
procedure EA
generate $P$ (an old population of $N$ points taken at random from $D$)
repeat
  find the worst point in $P$, $x_{\text{worst}}$, with the highest value of $f$
  copy $M$ best points of $P$ into new population $Q$. $1 \leq M < N$
  repeat
    repeat
      generate a new trial point $y$ by some heuristics applied to $P$
      until $f(y) < f(x_{\text{worst}})$
      insert the next trial point into $Q$
      until $Q$ is completed to $N$ points
    replace $P$ by $Q$
    until stopping condition is true
```

If we set the parameter $M = N - 1$ we get a generalization of the controlled random search described by Price [5]. The heuristics in this procedure is any non-deterministic rule which gives a new point $y \in D$. There is no reason for using the only heuristics in a given evolutionary algorithm. The idea of using more alternating heuristics during an optimization process appeared a few years ago (see e.g. [11, 9]). Let us have $h$ of heuristics at disposal. A new trial point can be generated by any of the heuristics. Each heuristics can be selected at random with probability $q_i$. When the probabilities $q_i$, $i = 1, 2, \ldots, h$, are changing depending on the success of the $i$-th heuristics during the process, the EA procedure becomes an evolutionary algorithm with competing heuristics. Such algorithm has the feature of self-adaptation based on evolutionary ideas.

The heuristics is successful in the current step of evolutionary process when it generates such a trial point $y$ that $f(y) < f(x_{\text{worst}}))$. When $n_i$ is the current number of the $i$-th heuristics’ successes, the probability $q_i$ can be evaluated as

$$q_i = \frac{n_i + n_0}{\sum_{j=1}^{h} (n_j + n_0)},$$

where $n_0 > 0$ is a constant. Setting $n_0 > 1$ prevents a dramatic change in $q_i$ by one random successful use of the $i$-th heuristics. There is another way how to appreciate the success of a given heuristics. It can be evaluated by using
the relative position of the trial point insertion into the new population \( Q \). Let us suppose that the new population \( Q \) of actual size \( L \) (after inserting a new trial point), \( L \in (M + 1, N) \), is ordered in such a way that \( f_1, f_2, \ldots, f_L \) is a nondecreasing sequence of the function values. The new trial point generated by the \( i \)-th heuristics is inserted into the new population \( Q \) on the position \( l \in (1, L) \). We can measure the success of the \( i \)-th heuristics by its weight \( w_i \) defined as \( w_i = (L - l + 1)/L \), \( w_i \in (0, 1) \), and probability \( q_i \) can be evaluated as

\[
q_i = \frac{W_i + w_0}{\sum_{j=1}^{h} (W_j + w_0)},
\]

where \( W_i = \sum w_j \) over a previous period of the process and \( w_0 > 0 \) is an input parameter which can be set to the mean value of \( w_i \), i.e. \( w_0 = 0.5 \).

To avoid the degeneration of evolutionary process it is sometimes useful to reset the current values of \( q_i \) to their starting values. When any probability \( q_i \) decreases below a minority limit \( \delta \), i.e. when \( q_i < \delta \), the values of \( n_i \) or \( w_i \) are reset to 0 what means that the probabilities are reset to their starting values, \( q_i = 1/h \), and the number of the resets is increased. It can be easily seen that the EA with competing heuristics is convergent if at least one of the heuristics in use is convergent itself, i.e. when it is used alone. In practice this can be ensured by including the heuristics generating a new trial points \( y \) uniformly distributed over \( D \). Let us denote this heuristics by random search.

A natural question arises how to estimate the probability that the random search produces a proper trial point. Suppose that the random search is included in the set of heuristics in use with its serial number 1. Then \( q_1(k) \) is the probability of using the random search in the \( k \)-th step of the EA algorithm and \( p \) is the probability that this heuristics produces a new trial point in a set \( S \subset D \) with \( \lambda(S) > 0 \) during all the process. The probability \( p_k(S) \) that the random search generates a new trial point in \( S \subset D \) just in the \( k \)-th step is given by

\[
p_k(S) = q_1(k) \frac{\lambda(S)}{\lambda(D)} \quad \text{and} \quad \therefore,
\]

\[
1 - p = \prod_{k=1}^{n} \left( 1 - q_1(k) \frac{\lambda(S)}{\lambda(D)} \right),
\]

where \( n \) is the number of new trials points generated during all the process. Denote \( \delta = \gamma/h \) for a suitable \( \gamma \in (0, 1) \). From the nature of the random search and the above condition for calculation of the probability \( q_i \) it follows that the value of \( q_i(k) \) will almost uniformly decrease from the initial value \( 1/h \) to the minimum possible value \( \delta \) in every period between two successive resets. Thus, the average value \( \overline{q}_1 \) of \( q_1(k) \) over all the process can be reasonably estimated by using the mean of both these limit values,

\[
\overline{q}_1 = \frac{1}{2} \left( \delta + \frac{1}{h} \right) = \frac{\gamma + 1}{2h}.
\]

The equation (3) can be rewritten in the form

\[
1 - p \approx \left( 1 - \frac{\gamma + 1}{2h} \frac{\lambda(S)}{\lambda(D)} \right)^n.
\]
For very small values of \( \lambda(S)/\lambda(D) \) (these sets \( S \) are of the major interest) the right side of the previous relation can be linearly approximated by using its differential and after a simple rearrangement we have

\[
p \approx \frac{n}{2h} \left(1 + \gamma\right) \frac{\lambda(S)}{\lambda(D)}.
\]

Thus we have the following approximate relation between \( p \) and \( \lambda(S) \)

\[
\frac{n \lambda(S)}{2h \lambda(D)} < p < \frac{n \lambda(S)}{h \lambda(D)}.
\]

Of course, with respect to using the linear approximation, the validity of this relation is restricted to such sets \( S \) that \( \lambda(S)/\lambda(D) \) are sufficiently small.

3 Heuristics

Some examples of heuristics for generating a new trial point \( y \) can be found in [1–3, 7, 8]. The heuristics used in our implementation of the EA with competing heuristics are described below and the results of preliminary experiments as well as their parameter setting is given in Table 1.

Heuristics inspired by evolutionary strategy (ES) generate a new trial point \( y \) according to the following rule

\[
y_i = x_{i}^{best} + Y, \quad i = 1, 2, \ldots, d,
\]

where \( x_{i}^{best} \) is \( i \)-th element of vector \( x^{best} \), i.e. the point with the lowest value of \( f \) in \( P \), and \( Y \) is random variable, \( Y \sim N(0, \sigma_i^2) \). Instead of the classic evolutionary strategy, where the values of \( \sigma_i^2 \) are adapted within evolutionary process by the one-fifth rule [3], we derive the values of these parameters from the current population \( P \). We used two variants of the ES heuristics. One denoted as \( esbest-pop \) computes \( \sigma_i \) from the whole population as follows:

\[
\sigma_i = c \left( \max_{x \in P} x_i - \min_{x \in P} x_i \right) + \varepsilon, \quad i = 1, 2, \ldots, d,
\]

where \( c > 0 \) is an input parameter and \( \varepsilon > 0 \) saves the value of \( \sigma_i \) positive (in our implementation \( \varepsilon = 1 \times 10^{-4} \)). In the heuristics denoted \( esbest-2pts \) the standard deviation \( \sigma_i \) is evaluated from two points \( r, s \) taken at random from the population \( P \) (except \( x^{best} \)) using the formula

\[
\sigma_i = c |r_i - s_i| + \varepsilon, \quad i = 1, 2, \ldots, d,
\]

where the symbols have the same meaning as above.

Randomized reflection of simplex [2] of \( d + 1 \) points chosen at random from \( P \) is described by

\[
y = g + U(g - x),
\]
where \( x \) is the simplex point with the highest \( f \) value in the \textit{refl-worst} heuristics or a randomly taken simplex point in the \textit{refl-rand} heuristics, \( g \) the centroid of the remaining simplex points and \( U \) random variable uniformly distributed on \( (0, \alpha) \), \( \alpha \) being an input parameter.

Two kinds of heuristics are based on differential evolution [7]. The heuristics denoted as \textit{de-rand} generates a point \( u \) using the relation

\[
u = r_1 + F(r_2 - r_3),
\]

where \( r_1, r_2, r_3 \) being mutually different points taken from \( P \) at random and \( F > 0 \) is an input parameter. The heuristic denoted as \textit{de-best} is based on the modification of the point \( x_{\text{best}} \) by the following way

\[
u = x_{\text{best}} + F(r_1 + r_2 - r_3 - r_4),
\]

where \( r_1, r_2, r_3, r_4 \) are again mutually different points taken at random from \( P \) (except \( x_{\text{best}} \)) and \( F \) is an input parameter. The new trial vector \( y \) is given by the crossover of vectors \( u \) and \( x \) chosen at random from \( P \) (except \( r_1, r_2, r_3, r_4 \) and \( x_{\text{best}} \)) according to the following rule

\[
y_i = \begin{cases} u_i & \text{if } U < C \text{ or } i = j \\ x_i & \text{otherwise} \end{cases}, \quad i = 1, 2, \ldots, d,
\]

where \( C \in (0, 1) \) is an input parameter, \( U \) random variable uniformly distributed on \( (0, 1) \). To prevent the case with no change of \( x \) (e.g. when \( C = 0 \)) at least \( x_j \) is replaced by \( u_j \). \( j \) is taken at random from \( (1, d) \).

Any heuristics described above does not guarantee that a new point \( y \) belongs to \( D \). In case \( y \notin D \) so called perturbation is applied [3].

4 Experiments and Results

The EA with eleven competing heuristics (see Table 1) was implemented in Matlab, version 6. In this table the reliability \( R \) is the percentage of the successful runs stopping at a point very near to the global minimum and \( NE \) the corresponding average number of objective function evaluations. Our testbed contains the first two of five De Jong’s functions (the second one is also known as Rosenbrock’s saddle), Ackley’s function and Griewangk’s function (see [7]). Some of them present a serious problem for many optimization algorithms. For each task 100 independent runs were carried out. Two variants of the EA with competing heuristics were tested: \textit{Competing1} using Eq. (1) and \textit{Competing2} using Eq. (2) for the evaluation of probabilities \( q_i \). The common input parameters were set as follows: \( N = 10d \) and stopping condition in the form

\[
f_{(N/2)} \leq 1 \times 10^{-7} \quad (f_{(1)} \leq f_{(2)} \leq \ldots \leq f_{(N)}) \text{.}
\]

The search is considered successful when \( f_{(1)} < 1 \times 10^{-3} \) for Ackley’s function and \( f_{(1)} < 1 \times 10^{-6} \) for others. The additional input parameters were \( \delta = 0.02, n_0 = 5, w_0 = 0.5 \) and \( M = N - 1 \). The CRS algorithm with alternating heuristics [10] (denoted \textit{Alternating}) use the same heuristics excepting the random search.
Table 1. Reliability and convergence rate of the CRS with one heuristic

<table>
<thead>
<tr>
<th>Heuristic</th>
<th>$c$, $F$, $a$</th>
<th>$d = 3$</th>
<th>$R$</th>
<th>$NE$</th>
<th>$d = 2$</th>
<th>$R$</th>
<th>$NE$</th>
<th>$d = 2$</th>
<th>$R$</th>
<th>$NE$</th>
</tr>
</thead>
<tbody>
<tr>
<td>esbest-2pts</td>
<td>1</td>
<td>-</td>
<td>100</td>
<td>585</td>
<td>100</td>
<td>3380</td>
<td>0</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>esbest-pop</td>
<td>0.2</td>
<td>-</td>
<td>100</td>
<td>457</td>
<td>100</td>
<td>2603</td>
<td>0</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>de-best</td>
<td>0.5</td>
<td>0.5</td>
<td>100</td>
<td>702</td>
<td>100</td>
<td>4060</td>
<td>93</td>
<td>752</td>
<td></td>
<td></td>
</tr>
<tr>
<td>de-best</td>
<td>0.9</td>
<td>0.5</td>
<td>100</td>
<td>1403</td>
<td>100</td>
<td>2586</td>
<td>96</td>
<td>1266</td>
<td></td>
<td></td>
</tr>
<tr>
<td>de-rand</td>
<td>0.5</td>
<td>0.5</td>
<td>96</td>
<td>879</td>
<td>39</td>
<td>1393</td>
<td>60</td>
<td>850</td>
<td></td>
<td></td>
</tr>
<tr>
<td>de-rand</td>
<td>0.9</td>
<td>0.5</td>
<td>100</td>
<td>1174</td>
<td>82</td>
<td>2049</td>
<td>75</td>
<td>1038</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ref-rand</td>
<td>2</td>
<td>-</td>
<td>100</td>
<td>892</td>
<td>100</td>
<td>745</td>
<td>99</td>
<td>1392</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ref-rand</td>
<td>6</td>
<td>-</td>
<td>100</td>
<td>2364</td>
<td>100</td>
<td>1565</td>
<td>99</td>
<td>3556</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ref-worst</td>
<td>2</td>
<td>-</td>
<td>100</td>
<td>875</td>
<td>100</td>
<td>576</td>
<td>99</td>
<td>1228</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ref-worst</td>
<td>6</td>
<td>-</td>
<td>100</td>
<td>2463</td>
<td>100</td>
<td>1430</td>
<td>98</td>
<td>3413</td>
<td></td>
<td></td>
</tr>
<tr>
<td>average</td>
<td></td>
<td></td>
<td>99.6</td>
<td>1179</td>
<td>92</td>
<td>1739</td>
<td>72</td>
<td>1687</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The experimental results as well as the comparison with Alternating are given in the Table 2. The column $p$ contains the $p$-values of one-way ANOVA tests (hypotheses that the means of $NE$ are the same for all the three algorithms). The symbols * behind the values denote the statistically significant differences from the corresponding values for Alternating at the significance level of 0.05. It is evident that the EAs with competing heuristics are at least as reliable as the Alternating one and in most cases the values of $NE$ are significantly lower. The $NE_1$ values for getting the best point of the population close to the global minimum are approximately the same as for differential evolution [7], in the case of Griewangk’s function even significantly lower ($NE_1 = 12.752$ for the differential evolution and $NE_1 = 8.481$ for the Competing2).

5 Conclusions

The EA with competing heuristics was proposed. Two instances of the algorithm were implemented and tested on five functions. The results were compared with those based on alternating heuristics [10]. The new algorithm was found more

Table 2. Comparison of EAs with competing and alternating heuristics

<table>
<thead>
<tr>
<th>Function</th>
<th>$d$</th>
<th>$R$</th>
<th>$NE$</th>
<th>$R$</th>
<th>$NE$</th>
<th>$R$</th>
<th>$NE$</th>
<th>$p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>De Jong 1</td>
<td>3</td>
<td>100</td>
<td>858</td>
<td>100</td>
<td>800</td>
<td>100</td>
<td>768</td>
<td>100</td>
</tr>
<tr>
<td>Rosenbrock</td>
<td>2</td>
<td>100</td>
<td>1111</td>
<td>100</td>
<td>996</td>
<td>100</td>
<td>930</td>
<td>100</td>
</tr>
<tr>
<td>Ackley</td>
<td>2</td>
<td>95</td>
<td>1137</td>
<td>98</td>
<td>1052</td>
<td>96</td>
<td>1007</td>
<td>96</td>
</tr>
<tr>
<td>Ackley</td>
<td>10</td>
<td>99</td>
<td>11881</td>
<td>99</td>
<td>10975</td>
<td>100</td>
<td>10428</td>
<td>100</td>
</tr>
<tr>
<td>Griewangk</td>
<td>10</td>
<td>62</td>
<td>10131</td>
<td>77</td>
<td>10520</td>
<td>72</td>
<td>9462</td>
<td>72</td>
</tr>
</tbody>
</table>
reliable and faster. The conditions for asymptotic convergence of the algorithm were discussed with respect to the implementation of the algorithm. The algorithm can be easily applied to searching for the constrained global minimum. The self-adaptation feature based on the heuristics' competition makes the setting of input parameters easier comparing with other evolutionary algorithms.

References


Acknowledgment

This research was supported by the grant no. 402/00/1165 of the Czech Grant Agency and the institutional research grant no. CEZ: J09/98:179000002.