On Termination Criteria of Evolutionary Algorithms

(http://www.pohlheim.com)

Brijnesh J. Jain
Department of Computer Science
Technical University Berlin
Franklinstr. 28/29
D-10587 Berlin, Germany

Hartmut Pohlheim
DaimlerChrysler AG
Research and Technology
Alt-Moabit 96a
D-10559 Berlin, Germany

Joachim Wegener
DaimlerChrysler AG
Research and Technology
Alt-Moabit 96a
D-10559 Berlin, Germany

Abstract

An overview of partly well-known termination criteria and a new developed criterion (ClusTerm) based on cluster analysis is given. In contrast to common termination criteria, ClusTerm combines both, information about objective values and the spatial distribution of individuals in the search space in order to terminate the evolutionary search. Extensive experiments using discrete objective functions are presented to compare and evaluate the termination criteria with respect to their reliability and performance. Guidelines for the practical employment of these criteria as well as references when certain criteria should not be used are given.

1 INTRODUCTION

Evolutionary Algorithms (EA) represent a class of probabilistic optimization algorithms which consciously imitate the principles of biological evolution[1]. In this contribution EA are used to minimize objective functions $f : S^{\mu} \rightarrow \mathbb{R}$. In an iterative process over several generations a given start population is subsequently modified to obtain a good or possibly optimal solution until a given termination criterion decides the end of the optimization process.

A suitable point in time to terminate an optimization run prevents premature termination as well as further computations to no avail. Hence the efficiency of a numerical algorithm for optimization is not only dependent on its computational performance but also on its behavior to terminate a run. Furthermore in some real-world applications of EA there are attempts to automate the optimization process as in evolutionary testing (O'Sullivan et al., 1998; Wegener & Grochtmann, 1998). In evolutionary testing, temporal correctness of real-time systems is verified with the assistance of EA whereby the objective functions are discrete. Since real-time systems are often safety-relevant, temporal correctness plays an important role. Consequently, reliable and intelligent termination criteria as well as general recommendations how to employ termination criteria for the automated use of EA are required.

In this paper we introduce a new reliable and intelligent termination criterion, called *ClusTerm*. Conventional criteria either use objective values or the distribution of individuals in the search space to decide the end of a run. To get other, probably more reliable and intelligent termination criteria, it is reasonable to combine and evaluate both, information about objective values and distribution of individuals. This approach is implemented by using cluster analysis on the fittest individuals.

Testing ClusTerm and comparing it with already existing termination criteria requires an adequate test suite. Since little research is done on termination criteria for practical application domains, a collective representation about some termination criteria and an empirical analysis using discrete objective functions is missing completely. To fill this gap, we present an extensive test set for termination criteria. Based on the results of our experiments, the termination criteria are examined and compared with respect to their reliability and performance. Finally, some guidelines how to employ the outlined termination criteria are suggested.

The paper is organized as follows: In sec. 2 an overview about commonly used termination criteria is presented and the ClusTerm criterion is defined. In sec. 3 all criteria are compared in an empirical analysis. Further-

[1] We assume the reader is familiar with the principles, techniques and the functionality of EA. Otherwise we refer to (Bäck, 1996; Pohlheim, 1999).

more, recommendations for the practical use of termination criteria are given. Sec. 4 concludes the paper.

2 TERMINATION CRITERIA

Using EA for optimization tasks, a search process over several generations is performed until a defined termination criterion determines the end of a run. A termination criterion is a decision rule of the form

```
IF (TC) THEN (terminate EA) ELSE (proceed EA)
```

where *TC* (*Termination Condition*) is a boolean expression. We say, a termination criterion is fulfilled, if the expression *TC* is true.

For the practical use a termination criterion should determine the end of a search process as soon as the EA is not sufficiently efficient. The efficiency of an EA is exhausted, when the EA is degenerated to a random search or no significant improvements of the best objective value can be expected within a foreseeable space of time. The significance of an improvement is highly dependent on the desired precision of the results and on the envisaged goals of the problem solving process. Each improvement corresponds to an economical profit and each iteration causes costs. From this point of view an optimization can run without loss up to a fixed precision. Improvements obtained without loss are considered to be significant.

In the following a number of termination criteria are presented and classified into *direct*, *derived* and *cluster-based termination criteria*. For each criterion a termination condition is given. Notations and terminologies are introduced stepwise at points where needed for the first time and implicitly hold for the rest of this paper.

2.1 DIRECT TERMINATION CRITERIA

Direct termination criteria are obtained immediately from the context of the optimization and do not require any further calculation of the underlying data. We present the direct criteria *Maximal Time*, *Maximal Number of Generations* and *Hitting a Bound*. In particular *Maximal Number of Generations* is considered to be the most used criterion for practical applications of EA (Hoffmeister & Bäeck, 1992; Schwefel, 1995; Pohlheim, 1999).

2.1.1 $\mathbf{T_1}$: Maximal Time & Maximal Number of Generations.

The termination criterion T_1 is fulfilled, if a given time budget t_{max} is consumed. The time t can be measured in terms of the absolute time, the CPU-time, the number of generations or the number of evaluations of the objective function. As all parameters are pairwise linearly dependent we do not distinguish between the criteria maximal time and maximal number of generations. Hence, in formal notation we have for all variants the following termination condition:

$$TC_1 ::= t_{max} - t \leq 0$$

2.1.2 $\mathbf{T_2}$: Hitting a bound.

The termination criterion T_2 is fulfilled, if the best objective value f_*^t at generation t reaches or surpasses a given bound f_{lim}. For minimization problems the termination condition is stated as follows:

$$TC_2 ::= f_{lim} \geq f_*^t$$

2.2 DERIVED TERMINATION CRITERIA

Derived termination criteria use the underlying data to calculate auxiliary values as a measure of the state of convergence of the evolutionary search process. Subsequently the criteria *Running Mean*, *Standard Deviation*, *Best-Worst*, *Phi* and *Kappa* are presented. The criteria *Phi* and *Kappa* were defined in (Vössner, 1996) as a convergence measure. In (Jain, 2000; O'Sullivan et al., 1998; Pohlheim, 1994-2000; Pohlheim, 1999) both measures were used as termination criteria.

2.2.1 T_3: Running Mean.

The termination criterion T_3 is fulfilled, if the difference between the best objective value f_*^t of the current generation t and the average of the best objective values $f_*^{t-1}, \ldots, f_*^{t-t_{last}}$ of the last t_{last} generations is equal to or less than a given threshold $\varepsilon \geq 0$.

$$TC_3 ::= \left| f_*^t - \frac{1}{t_{last}} \sum_{i=t-t_{last}}^{t-1} f_*^i \right| \leq \varepsilon$$

2.2.2 T_4: Standard Deviation.

The termination criterion T_4 is fulfilled, if the standard deviation of all objective values of the current generation is equal to or less than a given threshold $\varepsilon \geq 0$.

Let $P^t = (x_1^t, \ldots, x_\mu^t)$ be the population at generation t with μ individuals $x_i^t \in S$ $(1 \leq i \leq \mu)$ and f_i^t the objective value of the i-th individual x_i^t at generation t. Then the termination condition of T_4 is stated as follows:

$$TC_4 ::= \sigma_f = \sqrt{\frac{1}{\mu} \sum_{i=1}^{\mu} \left(f_i^t - \frac{1}{\mu} \sum_{i=1} \mu f_i^t \right)^2} \leq \varepsilon$$

2.2.3 T_5: Best-Worst.

The termination criterion T_5 is fulfilled, if the difference between the best and the worst objective value of the current generation is equal to or less than a given threshold $\varepsilon \geq 0$.

$$TC_5 ::= |f_*^t - \max\{f_i^t \mid 1 \leq i \leq \mu\}| \leq \varepsilon$$

2.2.4 T_6: Phi.

The termination criterion T_6 is fulfilled, if the quotient of the best objective value and the mean of all objective values of the current generation is equal to or greater than a given threshold $1-\varepsilon$ with $1 \gg \varepsilon \geq 0$.

$$TC_6 ::= 1 - \mu \cdot \frac{f_*^t}{\sum_{i=1}^{\mu} f_i^t} \leq 1$$

2.2.5 T_7: Kappa.

The termination criterion T_7 is fulfilled, if the quotient of the sum of all normalized distances between all individuals of the current generation and $\kappa_{max} = \frac{\mu^2-\mu}{2}$ is equal to or less than a given threshold $\varepsilon \geq 0$.

The criterion T_7 valuates the spatial spreading of individuals of the current population in the search space. For this purpose the normalized Euclidean distances d_{ij}/d between two individuals x_i and x_j for all $1 \leq i < j \leq \mu$ are measured where d is the length of the diagonal of the search space S. Some calculations finally lead to the following termination rule[2]:

$$TC_7 ::= 1 - \frac{1}{d \cdot \kappa_{max}} \sum_{i=1}^{\mu-1} \sum_{j=i+1}^{\mu} d_{ij} \leq \epsilon$$

2.3 CLUSTER-BASED TERMINATION CRITERIA

Cluster-Based termination criteria use clustering techniques to examine the distribution of individuals in the search space at a given generation. Individuals usually form clusters in the search space after a few generations of stagnation. A run is terminated when the clusters point out the convergence of the evolutionary search. In this case the fittest individuals are concentrated in few small regions of the search space.

2.3.1 CLUSTER ANALYSIS FOR EA

Hierarchical clustering techniques[3] construct cluster sequences in which larger clusters are obtained by merging smaller ones. Essentially hierarchical techniques may be subdivided into *agglomerative* and *divisive* methods. We only consider agglomerative methods as divisive methods are less common due to their higher computational effort (Steinhausen, 1977).

Agglomerative Methods

The basic procedure with all agglomerative methods is similar. The first step is to choose a proximity measure of distance or similarity between the individuals (entities). For convenience the description will be in terms of distance measures. Based on the chosen measure a distance matrix between the μ entities is computed.

Agglomerative methods start with the finest partition consisting of μ clusters where each of the clusters contains exactly one individual. Successively the closest clusters are merged until the data is reduced to a single cluster. The methods differ in the ways of defining the distance between the new formed cluster and all others clusters which didn't participate in the last merger. Among several agglomerative hierarchical techniques we have chosen the *single linkage method* for the following reasons: Firstly, the single linkage method is one of the very few clustering techniques which can outline non ellipsoidal clusters of ramified or curved shape. Thus, in case of EA different topologies of global optima can be identified. Secondly, the single linkage cluster analysis is invariant to any monotonic transformations of the distance matrix and finally it is the simplest of all known hierarchical techniques (Anderberg, 1973).

The Single Linkage Method

The single linkage method for EA works as follows: 1. Order the μ entities in ascending sequence and treat each entity as a cluster with exactly one member. 2. Find the two clusters which are closest together where the distance between them is the distance between their nearest members[4]. 3. Go back to step 2 until there is only one cluster left containing all μ entities or the minimal distance between two clusters exceeds a given threshold (*maximal distance stage*).

Data Description and Distance Measures

Three crucial points remain to discuss: (1) The choice of variables, (2) homogenizing variables, and (3) the choice of an adequate distance measure.

(1) The Choice of Variables: For our purposes the distribution of individuals in the search space is rather os-

[2]For a detailed description of Kappa we refer to the work of (Vössner, 1996).

[3]See e.g. (Anderberg, 1973; Steinhausen, 1977) for detailed representations of cluster analysis.

[4]Note that the algorithm is not well defined if the pair of clusters to be merged is not unique.

tensible than the discovery of a classification scheme. Hence all variables describing an individual are relevant for a cluster analysis.

(2) Homogenizing Variables: Subject to the chosen operators of the EA it is reasonable to normalize the variables of the individuals. Even though each transformation modifies the natural structure of the entities, normalizing the individuals results in comparable units and reflects the point of view of an EA when applying its variation operators.

(3) The Choice of an adequate Proximity Measure: Careful consideration is needed to select an appropriate measure to quantify the distances between individuals, as the output of the clustering algorithm (the number of clusters and the cluster membership of individuals) depends on the applied measure[5]. Depending on the scaling of the variables we suggest some distance measures summarized in table 1. For a mixture of different scalings niveau-regression and niveau-progression of the variables are the most important approaches to determine a distance between individuals (Steinhausen, 1977).

Table 1: Distance Measures

S	**Measure**	$\mathbf{d(x,y)}$
$\mathbb{R}^n$	Euclidean	$\sqrt{(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y})}$
$\mathbb{Z}^n$	Euclidean	as above
$\mathbb{Z}^n$	Canberra	$\sum_{i=1}^n \lvert x_i - y_i\rvert / \lvert x_i - y_i\rvert$
$\{0,1\}^n$	Simple Matching	$1 - \sum_{i=1}^n \lvert x_i - y_i\rvert / n$

Notations and Terminologies

We introduce some notations and terminologies to simplify the description of criterion T_8. Let d denote a distance measure and δ the maximal distance stage. With $Q^t \subseteq P^t$ we denote the subpopulation at generation t consisting of all individuals with objective value f_*^t. Clustering Q^t by the simple linkage method $agglom(d, Q^t, \delta)$ groups the normalized individuals up to the distance stage δ into *elitist clusters* and returns a (possibly 0-dimensional) vector $C_t = (c_1^t, \ldots, c_k^t)$ where the c_i^t are the number of individuals contained in the i-th cluster. Only clusters with at least two elements are considered. The sum $N_t = \sum_{i=1}^k c_i^t$ is called the *aggregate size of elitist clusters.*

[5]See (Anderberg, 1973) for a detailed discussion about this problem. A concise overview about several important proximity measures is given in (Steinhausen, 1977).

2.3.2 T_8: ClusTerm.

The termination criterion T_8 is fulfilled, if the change of the average of the aggregate size of elitist clusters averaged over the last t_{last}^* generations is equal to or less than a given threshold $1 \gg \varepsilon \geq 0$.

The aggregate size N_t of elitist clusters is subject to considerable fluctuations during stagnation. If a phase of stagnation lasts such a long time that in spite of the fluctuations the average $\bar{N}_t = 1/(t - t_0 + 1) \cdot \sum_{i=t_0}^t N_t$ converges then criterion T_8 decides the end of the evolutionary search process. Hereby t_0 denotes the generation of the last improvement. The computation of $\bar{N}_t$ starts not until a few generations t_{wait} since t_0 have been elapsed to avoid a random premature termination. Instead to average $\bar{N}_t$ from t_0 to t it is more convenient to average on the last $t_{last}^* := \min(t - t_0 + 1, t_{last})$ generations where t_{last} is the maximal number of the last generations to be considered. The resulting algorithm is given in figure 1. Suggested default values are embraced in brackets.

ALGORITHM OF CLUSTERM: Choose a measure d and set the maximal distance stage $\delta := [0.01 \cdot d]$ where d is the length of the normalized diagonal of S. Initialize $t_0 := 1$, $t_{wait} := [5]$ and $t_{last} := [15]$. P^t is passed by the EA.

1. If $f_*^t < f_*^{t-1}$ set $t_0 := t$, $\bar{N}_t := 0$ and return with $T_8 = 1$. Else go to (2).
2. Perform a cluster analysis on Q^t: $C^t = agglom(d, Q^t, \delta)$
3. Determine the aggregate size N_t of all elitist clusters.
4. If $t - t_0 < t_{wait}$ then return $T_8 = 1$. Else go to (5).
5. Determine the average $\bar{N}_t = \frac{1}{\tau}\sum_{i=\tau}^t N_t$ with $\tau := t - t_{last}^* + 1$.
6. Return $T_8 = \min\left(1, \frac{1}{\tau}\sum_{i=\tau+1}^t |\bar{N}_i - \bar{N}_{i-1}|\right)$.

Figure 1: Algorithm of ClusTerm

3 EXPERIMENTS

Initiated by the necessity of reliable and intelligent termination criteria in evolutionary testing, where the objective functions are discrete, we primarily consider multidimensional step functions in our empirical analysis. For testing the termination criteria $T_1 - T_8$ we constructed an artificial test set consisting of 9 newly defined discrete objective functions $f_1 - f_9$ (sec. 3.1). To verify the results of the empirical analysis the termination criteria were subsequently applied to real-world applications in evolutionary testing (sec. 3.2). All experiments were carried out by Pohlheim's EAimplementation GEATbx (Pohlheim, 1994-2000). The evaluation of the results and practical guidelines for using the termination criteria are given in sec. 3.3.

3.1 ARTIFICIAL OBJECTIVE FUNCTIONS

The test set covers a spectrum of various topologies of discrete step functions. In particular it combines the following antithetic features: (1) unimodal vs. multimodal[6], (2) plateaus of equal size vs. plateaus of different sizes, (3) symmetrically distributed plateaus vs. asymmetrically distributed plateaus, (4) large regions of global minima vs. small regions of global minima (5) low dimensional vs. high dimensional, and (6) discrete search space vs. continuous search space. Although all functions are pure mathematical constructions without any application based background, they allow a detailed examination of the behavior of termination criteria and also of different EA strategies for discrete problems, since the test set combines various characteristic features of possible topologies of multi-dimensional step functions.

The definitions of the functions $f_1 - f_9$ are listed in table 2. We used the following notation: $\lfloor x \rfloor := \max\{n \in \mathbb{Z} \mid n \leq x\}$, $\lceil x \rceil := \min\{n \in \mathbb{Z} \mid n \geq x\}$, and $|x| := \lfloor x + 0.5 \rfloor$.

Table 2: The Set of Test Functions

$$f_1(\mathbf{x}) = \sum_{i=1}^{n} [x_i]^2$$

$$f_2(\mathbf{x}) = \left[\frac{1}{10} \sqrt{\sum_{i=1}^{n-1} ([x_{i+1}] - [x_i])^2 - (1 - [x_i])^2} \right]$$

$$f_3(\mathbf{x}) = -\sum_{i=1}^{n} \left[\sin([x_i]) \cdot \sin(i \cdot [x_i]^2/\pi)^{20} \right]$$

$$f_4(\mathbf{x}) = \sum_{i=1}^{n} \left[\sum_{j=-100}^{100} \frac{1-\cos(\alpha_j \cdot [x_i])}{\alpha^{(2-D) \cdot j}} \right]$$

$$f_5(\mathbf{x}) = \sum_{i=1}^{n} A_i - B_i$$

$$f_6(\mathbf{x}) = \sum_{i=1}^{n} \left\lfloor -\lfloor x_i \rfloor \cdot \sin\left(\sqrt{|\lfloor x_i \rfloor|} \right) \right\rfloor$$

$$f_7(\mathbf{x}) = 1 + \sum_{i=1}^{n} \left\lfloor \frac{\lfloor x_i \rfloor^2}{4000} \right\rfloor - \prod_{i=1}^{n} \left\lfloor \frac{\cos(\lfloor x_i \rfloor)}{\sqrt{i}} \right\rfloor$$

$$f_8(\mathbf{x}) = [\log(1 + \sum_{i=1}^{n} |[x_i]| + \prod_{i=1}^{n} |x_i|)]$$

$$f_9(\mathbf{x}) = \begin{cases} \left\lceil \sqrt{\prod i = 1^n \lceil |x_i| + 1 \rceil} \right\rceil; & \text{if } x_1 \neq 0 \\ 0 \quad ; & \text{if } x_1 = 0 \end{cases}$$

The functions $f_1 - f_7$ are step functions obtained by discretization of common continuous test functions[7]. In particular, f_1 is derived from the sphere model used by De Jong. The function is unimodal with symmetrical distributed plateaus of identical size. The region of the global minimum is of size 1. f_2 (discrete variant of Rosenbrock's valley) is also unimodal with a global minimum which is embedded in a long narrow U-shaped valley. The plateaus are of different size whereby the region of the global minimum is relatively small compared to the size of adjacent plateaus. f_3 (discrete variant of Michalewicz's function) is highly multimodal. Its global minima are characterized by spikes within narrow rifts. f_4 (discretization of a fractal function) is unimodal with varying size of plateaus. The region of the global minimum is of size 1. The parameter settings for our experiments are $\alpha = 1.5$ and $D = 1.85$. A definition of the originally continuous fractal function is given in (Bäck, 1996). f_5 (discrete variant of the Fletcher-Powell function) is a highly multimodal step function with asymmetrical distribution of distinct plateaus. The location of its extrema is determined by the random matrices A and B. The values A_i and B_i in the definition of f_5 are dependent on the ith row of A and B. See (Bäck, 1996) and references therein for a definition of the continuous counterpart and (Jain, 2000) for the complete definition of f_5. f_6 (discrete variant of Schwefel's function) is multimodal whereby its local minima are spatially distant. f_7 (discrete variant of Griewangk's function) is also multimodal with uniformly distributed local minima. f_8 and f_9 are both unimodal step functions with symmetrically distributed plateaus. Both functions do not originate from existing test functions.

To satisfy features (5), (6) of the test set we considered search spaces of the form $[-500, +500]^n \subseteq \mathbb{R}^n$ and $[-500, +500]^n \subseteq \mathbb{Z}^n$ with $n = 2, 10, 50$.

For each dimension $n = 2, 10, 50$ we have chosen 7 configurations C_j^n of parameter settings of the EA. Each configuration varies the number of subpopulations σ, the population size λ, the selection pressure π_s, the mutation rate π_m, and the generation gap γ. Exemplary, table 3 shows the different configurations for 10-dimensional objective functions. All other parameters were set by default values according to GEATbx (Pohlheim, 1996-2001; Pohlheim 1999).

To obtain statistical significant results one test comprises $N(f_i, C_j^n) \in [50, 100]$ runs of an EA where the number of runs depends on the test function f_i ($i = 1, \ldots, 9$) and the configuration C_j^n ($j = 1, \ldots, 7$)

[6] A step function is called unimodal (multimodal), if the set of all local minima is a connected region (a union of two or more connected regions).

[7] Unless stated otherwise, references, descriptions, and definitions of the continuous objective functions can be found in (Pohlheim, 1999; Pohlheim 1994-2000).

Table 3: The Configuration Set for $n = 10$

PARAM	C1	C2	C3	C4	C5	C6	C7
σ	3	3	3	3	3	3	1
λ	60	60	60	60	60	60	40
π_s	2	1.3	2.5	2	2	2	1.7
π_m	0.1	0.1	0.1	0.025	0.15	0.1	0.06
γ	6	6	6	6	6	12	1

for dimension $n = 2, 10, 50$.

3.2 APPLICATION: EVOLUTIONARY TESTING

To verify the results of the empirical analysis we applied the termination criteria to evolutionary testing. In evolutionary testing the temporal correctness of real-time systems is verified. The task of the tester is to find the inputs with the shortest or longest execution time to check whether the test object produces a temporal error. The search of such inputs can be interpreted as an optimization problem which can be tackled by EA. The search space are all possible inputs and the objective function is the execution time measures in processor cycles. For determining the maximal and minimal execution time we have chosen two modules from safety-relevant systems.

The Crash-Control-Module is from automobile electronics. The test object consists of 1511 LOC (lines of code) and 70 input parameters. This module controls the trigger for airbags, sidebags, etc. The Discrepancy-Module is an application from railway technology. It consists of 389 LOC and 512 binary and integer input parameters. Both safety-relevant systems were provided by the DaimlerChrysler AG.

The parameter settings of the EA are: subpopulations $\sigma = 3$, population size $\lambda = 150$, selection pressure $\pi_s = 2$, mutation rate $\pi_m = 0.01$ (Crash-Control), $\pi_m = 0.002$ (Discrepancy), and generation gap $\gamma = 15$. For each test object one run of the EA is performed.

The outcome of both applications in evolutionary testing confirmed the results of our experiments with artificial objective functions.

3.3 RESULTS AND GUIDELINES

Based on the results of our test series, the termination criteria $T_1 - T_8$ are compared and evaluated in terms of reliability and performance. More precisely: A termination criterion T is called *reliable* when it guarantees that a run of an EA will terminate in finite time. A reliable termination criterion is called *proper* if the optimization process is terminated when the EA is no longer sufficiently efficient. Proper termination criteria are compared by their ability to detect inefficient states of an EA. A proper termination criterion T outperforms a proper termination criterion T' if T terminates an optimization run with less computations of objective values than T'.

3.3.1 Reliability

Termination of an optimization run within finite time is only guaranteed by the criteria T_1 and T_3. All other termination criteria except T_8 often did not terminate a run although the global optimum was found by the EA long ago. Criterion T_8 terminated each test run. But in contrast to T_1 and T_3 we are not able to make a statement about the reliability of T_8. A proof of the reliability of T_8 subject to certain conditions is given in (Jain, 2000).

Evidently criterion T_1 terminates an optimization run within finite time. For T_3 this assertion follows from the finiteness of the representation of numbers in a computer (see (Jain, 2000) for a formal proof).

Criterion T_2 does not terminate the search if the EA converges to a suboptimal solution such that its objective value is worse than the required bound f_{lim}. In particular T_2 will not terminate a run if the bound f_{lim} is out of the valid range of the objective function.

Due to the definitions of $T_4 - T_6$ the search is terminated only if the objective values of all individuals at the current generation are sufficiently similar. Already a few outliers with noticeable bad objective values affect the termination value such that a termination of the search is prevented. Thus, moderate results were obtained with f_7 and f_8 on $\mathbb{Z}^n$, since both functions have large plateaus of identical objective values and the discontinuities between adjacent plateaus are of step size less than 1. Outliers occur if variation operators ruled by constant parameters are employed to preserve the diversity throughout the optimization process and if the topology of the objective function is not flat in a surrounding of the local optimum. In this case a number of worse offspring will be produced in each generation even at the end of the search when most of the individuals have similar or identical objective values. Reliability of the termination criteria $T_4 - T_6$ can be attained only with partial serious restrictions on the search process. High selective pressure together with low mutation rate or adaptive operators (which can not applied to each parameter space) increase the chance of a termination within finite time.

Criterion T_7 terminates an optimization run when the population has converged to one point or a relatively small region in the search space. Like $T_4 - T_6$ a few spatial distant individuals prevent the termination of the search. In particular for multimodal problems the individuals are likely to gather at different optimal or suboptimal points. In this case the termination value of T_7 has no chance to fall below the given threshold. Furthermore, the spatial distribution of the individuals related to the size of the search space S is increasing with the dimension of S. Thus, the chance to terminate an optimization run by T_7 decreases with increasing dimension of S. Another problem by using T_7 arises with discrete search spaces. Here the Euclidean distance is in general not the adequate measure and prevents termination. Other distance measures better suited to the search space might help.

A possible solution of the outlier problem in $T_4 - T_7$ is to consider only a certain proportion of individuals to verify the corresponding termination conditions. But it is difficult to quantify the adequate proportion of individuals.

3.3.2 Performance.

The performance of the criteria T_2 and $T_4 - T_7$ is considered to be insufficient as these criteria frequently did not terminate the search even if the EA is degenerated to a random search. Hence, we only discuss the performance of the remaining criteria T_1, T_3 and T_8.

The main problem is to find suitable values for the parameters of the corresponding criteria to obtain a proper termination of the EA. Suitable parameters can only be determined by a trial-and-error method. Nevertheless, we will suggest default values which will constitute a good starting point and be sufficient for many problems.

Criterion T_1 determines the termination of the search process completely independent from the progression of the search. Critical is the choice of t_{max} to obtain a proper termination of a run. A default value for t_{max} can not be given.

Criterion T_3 terminates an optimization run if there is no or little improvement for the last t_{last} generations. The appropriate choice of the parameter t_{last} is crucial to ensure a proper termination. If t_{last} is too small a temporary stagnation phase causes a premature termination of the evolutionary search. On the other hand to many needless computations of the objective function are performed if t_{last} is chosen too large. We suggest to set $t_{last} = 15$.

Termination criterion T_8 decides the end of an optimization run if the duration of a stagnation phase lasts until the average of the oscillating aggregate sizes N_t of elitist clusters converge to a certain value up to a given precision ε. Hereby the choice of an appropriate value for the precision $\varepsilon > 0$ is important to obtain a proper termination. The parameter ε controls the duration of stagnation until termination where smaller values of ε corresponds to the property to pass longer stagnation phases without termination and vice versa. We suggest to set the precision $\varepsilon = 0.1$.

In contrast to T_3 the criterion T_8 adjusts the duration of a stagnation phase until termination subject to the kind of search. Fluctuations of the aggregate size N_t are higher when the search is more volume-oriented and lower in the case of a more path-oriented search. Therefore in a volume-oriented search it takes longer time until the average $\bar{N}_t$ reaches a stable state than in a path-oriented search. This corresponds to the idea to grant explorative search strategies a higher chance to find a better solution than an exploitative strategy once the EA is stuck in a suboptimal state.

3.3.3 Recommendations for practical use.

Obviously, termination criteria behave differently for varying EA strategies and objective functions. Therefore it is impossible to formulate a general rule for optimal use of a termination criterion. But in real-world applications like evolutionary testing an automated employment of EA is desired, since software tester often are not familiar with the behavior of EA. For this reason we will give some practical guidelines of the application of termination criteria for unskilled practioners and the automated use of EA. Our recommendations are not only restricted to discrete objective functions. They can also be accepted under reserve if one is dealt with continuous optimization problems. Our suggestions for the continuous case are based on first unsystematical experiments with the continuous counterparts of $f_1 - f_7$ and further continuous test functions like Ackley's Path function, Langermann's function, Branin's rcos function, Easom's function, Goldstein-Price's function, and Six-hump camel back function all described and defined in the GEATbx of (Pohlheim, 1999; Pohlheim 1994-2000; and references therein).

To guarantee termination the criteria T_2 and $T_4 - T_8$ should be used in disjunction with one of the reliable criteria T_1 or T_3.

To prevent needless computations in an inefficient state of the EA we suggest to employ T_3 or T_8. While the computational effort to verify TC_3 is less, TC_8 provides additional informations about the location of potential optimal solution for a local search in these regions.

Provided that the global optimum f_* is known in advance criterion T_2 is best used for comparative studies of different optimization strategies or for benchmarking of algorithms. In this case the bound f_{lim} corresponds to $f_* - \varepsilon$ where ε is the desired precision of the approximation for minimizing problems.

In some cases one is not interested in finding good or possibly optimal solutions but in the range of the objective function. With f_{lim} as the lower bound the EA can be terminated by T_2 when an objective value falls below the given bound f_{lim} even if the EA is still efficient. One example for the use of T_2 is given in (Jain, 2000; Wegener & Grochtmann, 1998).

The criteria $T_4 - T_7$ can be used only if the EA works with adaptive operators. The employment of these criteria for discrete valued objective function is not recommended. Furthermore T_7 is restricted only to continuous search spaces.

Finally, Table 4 summarizes the main characteristics of each termination criterion.

Table 4: Main Characteristics of $T_1 - T_8$

characteristic	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
reliable	+	-	+	-	-	-	-	0
discrete S	+	+	+	+	+	+	-	+
continuous S	+	+	+	+	+	+	+	+
discrete f	+	+	+	-	-	-	+	+
continuous f	+	+	+	+	+	+	+	+
any operators	+	+	+	-	-	-	-	+
adaptive operators	+	+	+	+	+	+	+	+

4 CONCLUSION

This paper gives a concise overview about a number of prominent termination criteria. Furthermore, we defined a new, reliable, and intelligent termination criterion (ClusTerm, T_8) based on combining information about objective values and distribution of individuals in the search space. ClusTerm is a first step to the development of intelligent termination criteria. The intelligence behavior of ClusTerm is incorporated in the fluctuations of the aggregate sizes of elitist clusters.

All termination criteria were systematically tested by extensive experiments using a test set of various self defined discrete objective functions. This test set can also be used for benchmarking EA strategies on discrete problems. The results of the experiments were verified by examples of evolutionary testing. Based on the results of our empirical analysis we compared and evaluated the termination criteria with respect to their reliability and performance. The criterion ClusTerm proved to be promising especially for problem domains with discrete objective functions. Finally, for each termination criterion guidelines for the practical employment and automated application of EA are suggested.

References

Anderberg, M. (1973). *Cluster Analysis for Applications.* New York: Academic Press, Inc.

Bäck, T. (1996). *Evolutionary Algorithms in Theory and Practice.* New York: Oxford University Press (1996).

Hoffmeister F. and Bäck, T. (1992). *Genetic Algorithms and Evolution Strategies: Similarities and Differences.* Technical Report No. SYS -1/92, System Analysis Research Group, University of Dortmund, Department of Computer Science, D-44221 Dortmund, Germany.

Jain, B.J. (2000). *Clusterbasierte Abbruchkriterien für den Evolutionären Test.* Diploma Thesis. Technical University Berlin, Department of Computer Science, D-10587 Berlin, Germany.

O'Sullivan, M., Vössner, S., and Wegener, J. (1998). *Testing Temporal Correctness of Real-Time-Systems.* Munich, Germany: Proceedings of EuroSTAR '98, 30 November - 4 December.

Pohlheim, H. (1994-2000). *GEATbx: Genetic and Evolutionary Algorithm Toolbox for use with Matlab. Documentation.* http://www.geatbx.com.

Pohlheim, H. (1999). *Evolutionäre Algorithmen. Verfahren, Operatoren, Hinweise aus der Praxis.* Berlin: Springer.

Schwefel, H.-P. (1995). *Evolution and Optimum Seeking.* New York: John Wiley.

Steinhausen, D., Langer K. (1997). *Clusteranalyse.* Berlin-Heidelberg: de Gruyter.

Vössner, S. (1996). *Convergence Measures for Genetic Algorithms.* Technical Report SOL 97-5, System Optimization Laboratory, Department of Engineering Economic Systems and Operations Research, Stanford University, USA.

Wegener, J., Grochtmann, M. (1998). *Evolutionary Testing of Temporal Corectness.* Brussels, Belgium: Proceedings of the 2nd Quality Week Europe '98, 9-13 November 1998.