Article

An Innovative Hybrid Approach to Global Optimization

Vasileios Charilogis[1],Glykeria Kyrou[2], Ioannis G. Tsoulos[3,*], Anna Maria Gianni[4]

[1] Department of Informatics and Telecommunications, University of Ioannina, Greece; v.charilog@uoi.gr
[2] Department of Informatics and Telecommunications, University of Ioannina, Greece; g.kyrou@uoi.gr
[3] Department of Informatics and Telecommunications, University of Ioannina, Greece; itsoulos@uoi.gr
[4] Department of Informatics and Telecommunications, University of Ioannina, Greece; am.gianni@uoi.gr
* Correspondence: itsoulos@uoi.gr;

Abstract: Global optimization is critical in engineering, computer science, and various industrial applications, as it aims to find optimal solutions for complex problems. The development of efficient algorithms has emerged from the need for optimization, with each algorithm offering specific advantages and disadvantages. An effective approach to solving complex problems is the hybrid method, which combines established global optimization algorithms. **This paper presents a hybrid global optimization method, which produces trial solutions for an objective problem, utilizing Genetic Algorithm genetic operators, as well as solutions obtained through a linear search process. Then, the generated solutions are used to form new test solutions, by applying Differential Evolution techniques.** These operations are based on samples derived either from internal line searches or genetically modified samples in specific subsets of Euclidean space. Additionally, other relevant approaches are explored to enhance the method's efficiency. The new method was applied on a wide series of benchmark problems from the recent literature and comparison was made against other established methods of Global Optimization.

Keywords: Optimization; Differential evolution; Genetic algorithm; Line search; Evolutionary techniques; Stochastic methods; Hybrid methods.

1. Introduction

The primary objective of global optimization is to locate the global minimum by thoroughly exploring the relevant range associated with the underlying objective problem. This method of global optimization is focused on identifying the global minimum within a continuous function that spans multiple dimensions. Essentially, the global optimization process is dedicated to seeking out the minimum value of a continuous, multidimensional function, ensuring that the search covers all potential ranges of the problem at hand. The objective is to find the lowest point through systematic exploration of the entire domain of the function, which is defined in a Euclidean space R^n. The optimal value of a function $f : S \to R, S \subset R^n$ is defined as follows:

$$x^* = \arg\min_{x \in S} f(x) \tag{1}$$

where the set S is defined as follows:

$$S = [a_1, b_1] \times [a_2, b_2] \times \ldots [a_n, b_n]$$

Global optimization refers to algorithms that aim to find the overall optimum of a problem. According to the literature survey, there are a variety of real-world problems that can be applied in mathematics [? ? ?], physics [? ? ?], chemistry [? ? ?], medicine [? ? ?], biology [? ?], agriculture [? ?] and economics [? ?]. Optimization methods can be categorized into deterministic [? ? ?] and stochastic [? ? ?] based on how they approach solving the problem. The techniques used for deterministic are mainly interval

Citation: Charilogis V.; Kyrou G.; Tsoulos I.G.; Gianni A.M. An Innovative Hybrid Approach to Global Optimization. *Journal Not Specified* **2023**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

methods [? ?]. Stochastic methods utilize randomness to explore the solution space, while in interval methods, the set S is divided into smaller regions that may contain the global minimum based on certain criteria. Recently, a comparison between deterministic and stochastic methods was proposed by Sergeyev et al [?].

A series of stochastic optimization methods are the so - called evolutionary methods, which attempt to mimic a series of natural processes. Such methods include the Genetic algorithms [? ?], the Differential Evolution method [? ?], Particle Swarm Optimization (PSO) methods [? ? ?], Ant Colony optimization methods [? ?], the Fish Swarm Algorithm [?], the Dolphin Swarm Algorithm [?], the Whale Optimization Algorithm (WOA) algorithm [? ? ?] etc. Also, due to the wide **spread of parallel computing units[?][?],** a variety of research papers related to evolutionary techniques appeared that use such processing units [? ? ?].

Genetic algorithms were formulated by John Holland [?] and his team and they initially generated randomly candidate solutions to an optimization problem. These solutions were modified through a series of operators that mimic natural processes, such as mutation, selection and crossover. Genetic algorithms have been used widely in areas such as networking [?], robotics [? ?], energy topics [? ?] etc. They can be combined with machine learning to solve complex problems, such as neural network training [? ?].

Furthermore, differential evolution (DE) is used in symmetric optimization problems [? ?] and in problems that are discontinuous and noisy and change over time. After studies, it was observed that differential evolution can be successfully combined with other techniques for machine learning applications, such as classification [? ?], feature selection [? ?], deep learning [? ?] etc.

Hybrid methods [? ?] in global optimization refer to techniques that combine multiple optimization strategies to solve complex problems. These methods aim to take advantage of different approaches to find the global optimum in a more efficient way, particularly when dealing with large-scale problems or strongly nonlinear optimization landscapes. A typical example of a hybrid method is the work of Shutao Li et al. who proposed a new hybrid PSO-BFGS strategy for the global optimization of multimodal functions [?]. To make the combination more efficient, they proposed an LDI to dynamically start the local search and a repositioning technique to maintain the particle diversity, which can effectively avoid the premature convergence problem. Another innovative hybrid method is the work of M. Andalib Sahnehsaraei et al. where a hybrid algorithm using GA operators and PSO formula is proposed was presented through the use of efficient operators, for example, traditional and multiple crossovers, mutation and PSO formula [?].

In the current work, two evolutionary methods were incorporated into the final algorithm: Genetic Algorithms and the Differential Evolution method were combined into a hybrid optimization method.

More specifically through a series of steps trial solutions are generated using the genetic operators of the Genetic Algorithm as well as solutions determined by a line search procedure. Additionally, an Armijo line search method is used. This method is incorporated to estimate an appropriate step when updating the trial points, and it was introduced in the work of Armijo[?]. The solutions produced in the previous step are used to formulate new trial solutions using a process derived from Differential Evolution.

The remainder of this paper is divided into the following sections: in section **??**, the proposed method is described, in section **??** the experimental results and statistical comparisons are presented, and finally in section **??** some conclusions and guidelines for future improvements are discussed.

2. The overall algorithm

(WRITE A TEXT BEFORE THE ALGORITHM)

The proposed method combines some aspects from different optimization algorithms and the main steps are subsequently:

1. **Initialization step.**
   (a) **Set** the population size $N \geq 4$.
   (b) **Set** n the dimension of the benchmark function.
   (c) **Initialize** the samples $x_i, i = 1, \ldots, N$ using uniform or k-means[**?**] distribution.
2. **Calculation step.**
   (a) **For** $i = 1 \ldots \mathrm{N}$ **do**
      i. **Obtain** sample x_i.
      ii. **Find** nearest sample c_i from x_i**:**

$$d(x_i, c_i) = \sqrt{\sum_{j=1}^{n} \left(x_{i,j} - c_{i,j}\right)^2} \quad (2)$$

      iii. **Set** direction vectors: $p_1 = -\nabla f(x_i)$ end $p_2 = -\nabla f(c_i)$
      iv. **Set** initial step size for Armijo $a = a_0$
      v. **Compute** with line search Armijo the sample:
         - **Find** new points using line search minLS(x_i, c_i): $x_i^{new} = x_i + ap_1$ and $c_i^{new} = c_i + ap_2$
         - **Adjust** step size a until Armijo condition is met:

$$f(x_i^{new}, c_i^{new}) \leq f(x_i, c_i) + c_1 a \nabla f(x_i, c_i)^T (p_1, p_2) \quad (3)$$

      vi. **Make** sample-child with crossover with random number $g_k \in [0.0, 1.0]$:

$$\mathrm{child}\left(x_i, x^{best}\right) = g_k x_{i,k} + (1 - g_k) x_k^{best} \quad (4)$$

      vii. **For** $j = 1, \ldots, n$ **do**
         - **Set** trial vector:

$$y_j = x_{i,j} + F \times \left(\mathrm{minLS}(x_\iota, c_\iota)_j - \mathrm{child}(x_i, c_i)_j\right) \quad (5)$$

           where F is the so - called Differential Weight of Differential Evolution algorithm.
         - **If** $y_j \notin \left[a_j, b_j\right]$**,** then $y_j = x_{i,j}$
      viii. **EndFor**
      - **Set** $r \in [0, 1]$ a random number. If $r \leq p_m$ then $x_i = \mathrm{LS}(x_i)$, where $\mathrm{LS}(x)$ is a local search procedure like the BFGS procedure[**?**].
      - **If** $f(y) \leq f(x)$ then $x = y$, $x^{best} = y$.
   (b) **EndFor**
3. **Check the termination rule stated in** [**?**], which means that the method checks the difference between the current optimal solution $f_{min}^{(t)}$ and the previous $f_{min}^{(t-1)}$ one. The algorithm terminates when the following:

$$\left|f_{\min}^{(t)} - f_{\min}^{(t-1)}\right| \leq \epsilon \quad (6)$$

   holds for N_t iterations. The value ϵ is a small positive value. In the conducted experiments the value $\epsilon = 10^{-5}$ was used. If the termination rule of equation **??** does not hold, then the algorithm continues from Step **??**.
4. **Return** the sample x^{best} in the population with the lower function value $f\left(x^{best}\right)$.

The algorithm is also shown as a flowchart in Figure **??**.

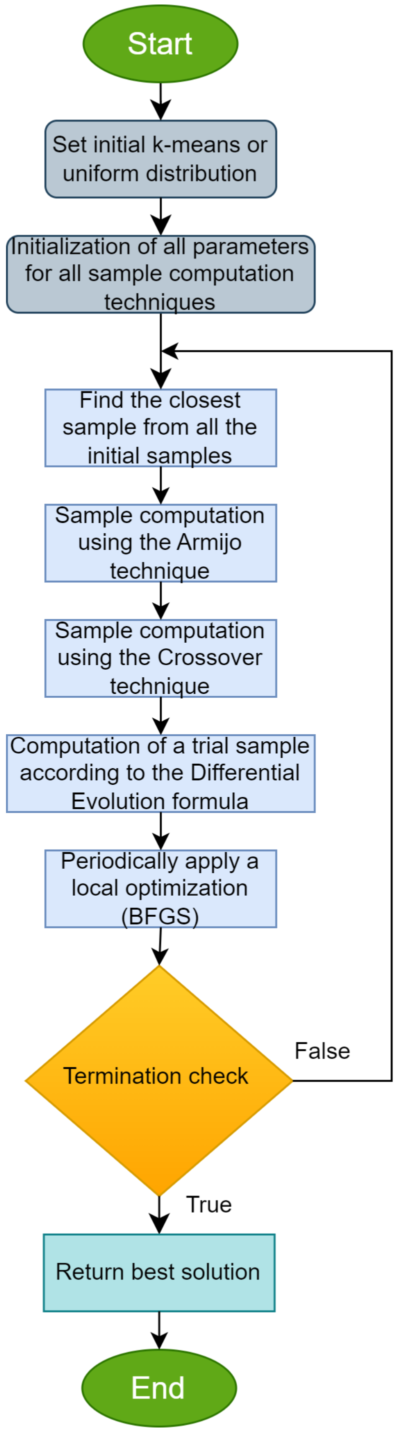

Figure 1. The flowchart of the proposed optimization process.

The optimization method described in section **??** combines evolutionary techniques, such as differential evolution, Armijo line search, and components of genetic algorithms, with the aim of finding the optimal solution. Initially, the population size and the dimensionality of the target function are defined, and the population samples are generated randomly using a uniform or k-means distribution. For each sample, the Euclidean distance (equation **??**) to the other samples is calculated to identify the nearest one, followed by an Armijo line search (equation **??**) to determine the optimal movement direction for the initial sample. Subsequently, a new offspring sample is created using a crossover process (equation **??**), combining the current sample with the best discovered so far. A trial vector (equation **??**) is then formed, which accounts for the adjustment of the computed samples and the differential coefficient parameter derived from the differential evolution algorithm. Periodically, local search methods, such as BFGS[**?**], are applied to improve the accuracy of the solution search. The method terminates when the best solution found remains nearly unchanged for a specified number of iterations. In summary, the basic steps for calculating a new sample are:

- Identification of the nearest point c_ι for each sample x_ι.
- Calculation of a sample $\mathrm{minLS}(x_i, c_i)$ through Armijo line search, between the sample x_i and the samplec_i.
- Generation of the sample using the crossover process of the Genetic Algorithm, between the sample x_ι and the best samplex^{best} .
- Computation of the trial point y_i using a process derived from Differential Evolution.

3. Experiments

Settings and benchmark functions

The benchmark functions used in the experimental measurements are presented in Table **??**.

Table 1. The benchmark functions used in the conducted experiments.

NAME	FORMULA	DIMENSION
ACKLEY	$f(x) = -a\exp\left(-b\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(cx_i)\right) + a + \exp(1) \quad a = 20.0$	2
BF1	$f(x) = -20\exp\left(-b\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(cx_i)\right) + 20 + \exp(1)$	2
BF1	$f(x) = x_1^2 + 2x_2^2 - \frac{3}{10}\cos(3\pi x_1) - \frac{4}{10}\cos(4\pi x_2) + \frac{7}{10}$	2
BF2	$f(x) = x_1^2 + 2x_2^2 - \frac{3}{10}\cos(3\pi x_1)\cos(4\pi x_2) + \frac{3}{10}$	2
BF3	$f(x) = x_1^2 + 2x_2^2 - \frac{3}{10}\cos(3\pi x_1)\cos(4\pi x_2) + \frac{3}{10}$	2
DIFFPOWER	$f(x) = \sum_{i=1}^{n} \lvert x_i - y_i \rvert^p$	$n = 2\ p = 2, 5, 10$
CAMEL	$f(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4, \quad x \in [-5, 5]^2$	2
EASOM	$f(x) = -\cos(x_1)\cos(x_2)\exp\left((x_2 - \pi)^2 - (x_1 - \pi)^2\right)$	2
ELP	$f(x) = \sum_{i=1}^{n} \left(10^6\right)^{\frac{i-1}{n-1}} x_i^2$	$n = 10, 20, 30$
EXP	$f(x) = -\exp\left(-0.5\sum_{i=1}^{n} x_i^2\right), \quad -1 \le x_i \le 1$	$n = 4, 8, 16, 32$
F3	$f(x) = \left(e^{-2.0\log(2.0)\left(\frac{(x_1 - 0.08)}{0.854}\right)^2}\right)\left(\sin\left(5.0\pi\left(x_1^{\frac{3.0}{4.0}} - 0.05\right)\right)\right)^6 \quad x \in [0, 1]^n$	2
F5	$f(x) = \left(\left(4.0 - 2.1x_1^2 + \frac{x_1^4}{3.0}\right)x_1^2\right) + (x_1x_2) + \left((4.0x_2^2 - 4.0)x_2^2\right) \quad -5 \le x_i \le 5$	2
F9	$f(x) = -\exp\left(-0.5\sum_{i=1}^{n} x_i^2\right), \quad x \in [0, 1]^n$	2
GKLS[**?**]	$f(x) = \mathrm{Gkls}(x, n, w)$	$n = 2, 3\ w = 50, 100$
GRIEWANK2	$f(x) = 1 + \frac{1}{200}\sum_{i=1}^{2} x_i^2 - \prod_{i=1}^{2}\frac{\cos(x_i)}{\sqrt{(i)}}$	2
GRIEWANK10	$\mathrm{f}(x) = 1 + \frac{1}{200}\sum_{i=1}^{10} x_i^2 - \prod_{i=1}^{10}\frac{\cos(x_i)}{\sqrt{(i)}}$	10
HANSEN	$f(x) = \sum_{i=1}^{5} i\cos[(i-1)x_1 + i]\sum_{j=1}^{5} j\cos[(j+1)x_2 + j]$	2
HARTMAN3	$f(x) = -\sum_{i=1}^{4} c_i\exp\left(-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2\right)$	3
HARTAMN6	$f(x) = -\sum_{i=1}^{4} c_i\exp\left(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right)$	6
POTENTIAL[**?**]	$V_{LJ}(r) = 4\epsilon\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^6\right]$	$n = 9, 15, 21, 30$
RARSTIGIN	$f(x) = x_1^2 + x_2^2 - \cos(18x_1) - \cos(18x_2)$	2
ROSENBROCK	$f(x) = \sum_{i=1}^{n-1}\left(100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right), \quad -30 \le x_i \le 30$	$n = 4, 8, 16$
SCHWEFELH	$f(x) = \sum_{i=1}^{n}\left(\sum_{j=1}^{i} x_j\right)^2$	2
SCHWEFELH221	$f(x) = 418.9829n + \sum_{i=1}^{n} -x_i\sin\left(\sqrt{\lvert x_i \rvert}\right)$	2
SCHWEFELH222	$f(x) = \sum_{i=1}^{n-1}\left(100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right), \quad -30 \le x_i \le 30$	2
Shekel5	$f(x) = -\sum_{i=1}^{5}\frac{1}{(x - a_i)(x - a_i)^T + c_i}$	4
Shekel7	$f(x) = -\sum_{i=1}^{7}\frac{1}{(x - a_i)(x - a_i)^T + c_i}$	4
Shekel10	$f(x) = -\sum_{i=1}^{10}\frac{1}{(x - a_i)(x - a_i)^T + c_i}$	4
Sinusoidal[**?**]	$f(x) = -\left(2.5\prod_{i=1}^{n}\sin(x_i - z) + \prod_{i=1}^{n}\sin(5(x_i - z))\right), \quad 0 \le x_i \le \pi$	$n = 4, 8$
Test2N	$f(x) = \frac{1}{2}\sum_{i=1}^{n} x_i^4 - 16x_i^2 + 5x_i$	$n = 4, 5, 7$
Test30N	$\frac{1}{10}\sin^2(3\pi x_1)\sum_{i=2}^{n-1}\left((x_i - 1)^2\left(1 + \sin^2(3\pi x_{i+1})\right)\right) + (x_n - 1)^2\left(1 + \sin^2(2\pi x_n)\right)$	$n = 3, 4$