Multistart global optimization with tunnelling and an evolutionary strategy supervised by a martingale

Ruixiang Zheng\textsuperscript{a} and Mian Li\textsuperscript{a,b}

\textsuperscript{a}University of Michigan–Shanghai Jiao Tong University Joint Institute, Shanghai Jiao Tong University, Shanghai, P.R. China; \textsuperscript{b}Department of Automation, Shanghai Jiao Tong University, Shanghai, P.R. China

ABSTRACT

Global optimization is one of the most difficult research topics in the optimization area. In the past few decades, many stochastic methods have been proposed that proved to be effective. In this article, a new multistart global optimization method, using tunnelling and an evolutionary strategy and supervised by a martingale (MGO-TEM) is proposed, which also belongs to the stochastic family. MGO-TEM is under a multistart framework such that a local search method will be called when a new sample point is generated. A sampling technique that is a hybrid of tunnelling and an evolutionary strategy is proposed in MGO-TEM with a set of stopping criteria whose derivation is based on martingale theory. Through experiments, MGO-TEM is compared with 10 other well-known algorithms on 55 single-objective test functions, and the results show that MGO-TEM is superior to all of them. Moreover, MGO-TEM is user-friendly since there are only two parameters to be tuned.

1. Introduction

With the flourishing of artificial intelligence nowadays, there is a higher demand than ever on the performance of global optimization algorithms (Chow \textit{et al.} 2002; Bengio 2009; Cassioli \textit{et al.} 2012). Non-convex optimization problems arise quite often in any layer of artificial intelligence applications as well as in many other engineering applications such as multidisciplinary design optimization (Bottou 2010; Mu Li \textit{et al.} 2014). However, the global optimization of arbitrary real continuous functions is still a difficult task and full-length research on this topic has been conducted by many researchers for decades. Many different methods have been proposed and proved to be effective.

Given the fact that there does not exist any characteristics that can theoretically differentiate a global minimum from a local minimum, deterministic approaches are hard to develop. One existing deterministic method called $\alpha$BB was proposed by Floudas (2005), which is non-heuristic and guarantees $\epsilon$-suboptimality, but the convexifying procedure for various objective and constraint functions is demanding and the algorithm is often very slow. Other efficient deterministic methods, such as DIRECT and pattern search (PS) were also developed and had been applied to engineering problems (Campana \textit{et al.} 2016; Suresh Kumar, Nagesh Kumar, and Madichetty 2017). However, those methods do not have guaranteed optimality in general. Contrary to deterministic approaches, many
methods were proposed to obtain the global minimum in a stochastic manner, such as genetic algorithms (GAs), simulated annealing (SA), differential evolution (DE), the covariance matrix adaptation evolution strategy (CMA-ES), etc. (Holland 1992; Bäck 1996; Storn and Price 1997; Hansen and Ostermeier 2001; Yang 2014). Typically, those methods are heuristic and require parameter tuning. As a result, the optimality of the obtained global (or even local) minima from those methods cannot be guaranteed. Compared with deterministic methods, stochastic ones are mainly used for high-dimensional problems, noisy or disturbed objective functions, or when the gradient information of objective functions is not available. Therefore, research on stochastic methods remains in the mainstream of global optimization. To improve the performance of stochastic methods, there are hybrid methods that first apply a stochastic method, followed by a local method such as those described in Kelner et al. (2008), Mahinthakumar and Sayeed (2005) and Alsumait, Sykulski, and Al-Othman (2010). However, those articles focus on specific applications, and therefore global convergence properties of the hybrid methods are not theoretically analysed.

In general, stochastic methods usually have heuristic structures so as to evolve iteration by iteration. By sampling the feasible domain, a better minimum is expected to be found. For example, GAs, CMA-ES, DE and particle swarm optimization (PSO) are population-based algorithms and they generate new sample points via operations such as mutation and crossover among the population (Cortez 2014; Wang, Tan, and Liu 2018; Sengupta, Basak, and Peters 2019). To this end, the population size, mutation and crossover rates, possibly with other parameters, need to be tuned. CMA-ES and Bayesian optimization (BOA) assume a jointly Gaussian correlation among all the sample points and requires tuning the kernel and acquisition functions (Brochu, Cora, and de Freitas 2010). Different heuristics and parameters for SA, global search (GS) and multistart (MS) can be found in Ugray et al. (2007). It should be noted that only the major parameters of the aforementioned stochastic methods are discussed. In reality, once the parameters are fine tuned, enabling the method to adjust itself smoothly to the landscape of the function, its performance will be good. Nevertheless, those parameters are usually empirical, making the tuning task very difficult. To summarize, stochastic methods have effective sampling techniques but their performance is based on parameter tuning. There exists, nevertheless, research on guaranteed convergence to local optimal solutions such as in Morinaga and Akimoto (2019). However, only the class of strongly convex functions with Lipschitz continuous gradients was considered in that article. Other research on guaranteed convergence to local optimal solutions can be found in Rudolph, Informatik, and Xi (1997) and Jägersküpper (2006, 2007).

In the stochastic family, memetic algorithms are close to a form of population-based hybrid GA coupled with an individual learning procedure capable of performing local refinements, first introduced in Moscato (2000), and have become one of the recent growing research topics in evolutionary computation. A broad review on many memetic algorithms is presented in Neri and Cotta (2012) with the pre-mature convergence issue being addressed. Therefore, theoretical analysis on the global convergence of a hybrid method is important and can be achieved if two aspects, namely the sampling strategy and the stopping criteria, are well designed. In this article, major efforts will be devoted towards these two aspects, especially the second one. As a result, a new global optimization method called multistart global optimization with tunnelling and an evolutionary strategy supervised by a martingale (MGO-TEM) is proposed, which belongs to the stochastic family. In this method, sample points will be generated using a proposed hybrid sampling technique, and a local search method will be initiated at each sample point. Rigorous analysis is conducted to provide well-designed stopping criteria that ensure optimality. Martingale theory (Chung 2001; Steele 2001) is used to derive the stopping criteria, and therefore the optimization process is supervised by a martingale. In MGO-TEM, two straightforward parameters (representing two probabilities) are left for the users to tune. Since these two parameters have clear physical meanings, the tuning task is straightforward and simple. In the literature, global optimization using the concept of a martingale was mentioned in Lockett and Miikkulainen (2014). However, merely a concept was discussed and no theoretical results were derived using martingale theory. In this article, the proposed MGO-TEM method will be verified on
55 single-objective test functions and compared with another 10 well-known methods. The results obtained for the test functions are outstanding, which demonstrates the applicability and capability of MGO-TEM.

The rest of the article is organized as follows. Details of MGO-TEM are presented in Section 2. The stopping criteria are proposed in this section together with the sampling technique. In the same section, a test series is also provided and the pseudocodes are provided in the online supplemental data. In Section 3, experiments are carried out on 55 single-objective test functions. The performances of the 11 global optimization methods are compared. In Section 4, mathematical derivations of all the stopping criteria are presented. Conclusions are given in Section 5.

2. Proposed MGO-TEM

The standard optimization problem to be discussed in this work is given as

$$\minimize_{x \in \mathbb{Q}} f(x),$$

where \(f : \mathbb{R}^n \rightarrow [0, 1]\) is a continuous function, and the domain \(\mathbb{Q} \subseteq \mathbb{R}^n\) is closed and bounded. Note that neither \(f\) nor \(\mathbb{Q}\) is necessarily convex. For functions with a range that is not contained in \([0, 1]\), the ‘normalization up to iteration \(t\)’ technique can be found on the first page of the online supplemental data. Assume there are \(N\) local minima, denoted by \(f_i\), \(i = 1, \ldots, N\). Then, at the \(t\)th iteration, the local minimum \(f_t^*\) found by the local method will be one of them, i.e.

$$f_t^* \in \{f_1^*, f_2^*, \ldots, f_N^*\} \subseteq [0, 1].$$

If the associated local minimizer of \(f_t^*\) is denoted by \(x_t^*\), then \(f_t^* = f(x_t^*), t = 1, 2, \ldots\). Hence, it is possible that several minimizers have the same function value.

Furthermore, \(w_t\) is used to denote the transformed running minimum at the \(t\)th iteration, i.e.

$$w_t = \exp \left( -\min_{s \leq t} f_s^* \right).$$

Therefore, the sequence \((w_t)\) is non-decreasing and takes values in \([e^{-1}, 1]\).

For the purpose of global minimization, how to generate sample points and how to terminate the searching process are two fundamental and important aspects. MGO-TEM fully considers these two aspects and provides novel sampling and terminating mechanisms. The contributions of MGO-TEM are as follows:

- by using martingale theory to analyse the trends of the obtained local minima and the running minima, well-designed stopping criteria and a well-designed sampling technique are proposed;
- compared with other stochastic methods, the sampling technique is more effective and the proposed stopping criteria enhance the efficiency and probability of obtaining the global minima upon termination.

Overall, MGO-TEM consists of

- a deterministic local method;
- a proposed sampling technique that generates initial points for the local method;
- a proposed set of stopping criteria.

Since a local method is called during the optimization process, multiple local minima will be obtained. By using martingale theory to analyse the obtained local minima, the sampling technique
as well as the set of stopping criteria of MGO-TEM are proposed. Next, the sampling technique used in MGO-TEM is introduced. Then, four stopping criteria are presented and the physical meaning of each one will be discussed. To aid readability of this section, mathematical derivations of these four stopping criteria are deferred to Section 4.

2.1. The sampling technique

To proceed with the multistart process before its termination, initial points for the local method should be spread in the feasible domain, which demands a sampling technique. Generally speaking, it enhances efficiency to use a good sampling technique that balances the performance of exploration against that of exploitation. In the following, a hybrid sampling technique that is a combination of a martingale property (exploitation) and an existing evolutionary strategy (exploration) is presented.

Assuming that the multistart process has proceeded to the $t$th iteration, a design vector $\mathbf{x}_{CB} \in \mathbb{R}^n$ is defined as

$$\mathbf{x}_{CB} = \arg\min_{\mathbf{x}_s, s \leq t} f(\mathbf{x}_s),$$

which is the best design vector so far during the multistart process. This $\mathbf{x}_{CB}$ will be updated in each iteration.

2.1.1. The tunnelling process

The martingale, due to its special property, has been used to aid sampling such as in Qiu, Zhou, and Wu (2008). In MGO-TEM, the martingale property is utilized to design a tunnelling process based on the discussions in Section 4.1. Specifically, at the $t$th iteration ($t \geq 2$), a sub-domain $Q_{sub}$ is created as follows. When the local minimizer $\mathbf{x}_{*t}$ and its neighbouring local minimizer $\mathbf{x}_{*nb} = \arg\min_{\mathbf{x}_s, s \leq t-1} ||\mathbf{x}_{*t} - \mathbf{x}_s||_2$ satisfy

$$\frac{f(\mathbf{x}_{*t}) + f(\mathbf{x}_{*nb})}{2} \leq \frac{1}{t} \sum_{s=1}^{t} f(\mathbf{x}_s),$$

(1)
a hypercube with $\mathbf{x}_{*CB}$ and $\arg\min_{\mathbf{x}\in\{\mathbf{x}_{*t}, \mathbf{x}_{*nb}\}} f(\mathbf{x})$ being two vertices on the main diagonal is created. This hypercube should be truncated when necessary to ensure $Q_{sub} \subseteq Q$. Then, a point is randomly spread in $Q_{sub}$ and a local method with that point being the initial point runs to obtain a new local minimizer $\mathbf{x}_{*t+1}$. This step is called Tunnelling, and the above procedures are written as

$$\mathbf{x}_{*t+1} = \text{Tunnelling}(\mathbf{x}_{CB}, \tilde{x}),$$

where $\tilde{x} = \arg\min_{\mathbf{x}\in\{\mathbf{x}_{*t}, \mathbf{x}_{*nb}\}} f(\mathbf{x})$. Tunnelling is regarded as an operator from now on. Further, $\mathbf{x}_{*t+1}$ is called a one-level-deeper design vector compared with $\mathbf{x}_{CB}$ and $\tilde{x}$. Thus, the tunnelling step can be performed iteratively if $\mathbf{x}_{*t+1}$ and its neighbouring local minimizer also satisfy Equation (1). To avoid tunnelling too deeply, the maximum level of tunnelling is set to be $\lceil \sqrt{n} \rceil$ where $n$ is the dimension of the problem. The pseudocode for the iterative tunnelling process is provided as Algorithm 3 in the online supplemental data.

The purpose of the Tunnelling process is to improve the exploitation ability of the sampling process. Therefore, the tunnelling process may ignore certain good solutions in the searching space and that is why the tunnelling process needs to work together with the evolutionary strategy to achieve a balanced performance between exploration and exploitation. Please refer to the pseudocodes of Algorithm 1, Algorithm 2 and Algorithm 3 in the online supplemental data to see how they work together. The following section presents the technical details of the evolutionary sampling strategy.
2.1.2. Evolutionary sampling

The evolutionary sampling strategy is much like that in the DE method. The details of the strategy can be found in Ali and Törn (2004), Storn and Price (1997) and Ronkkonen, Kukkonen, and Price (2005), and it has been proved effective in many applications (Gao et al. 2019; Stanov, Akhmedova, and Semenkin 2019; Ye et al. 2019). The only difference is the occurrence of the vector acting as the interface between the evolutionary sampling strategy and the tunnelling process. The terminologies in the DE method will be used directly as follows. The population size is fixed to be $10n$, i.e. $Pop = \{x_1, x_2, \ldots, x_{10n}\}$, and the crossover probability is chosen to be 0.2. Then, the usual mutation and crossover in DE are carried out on $Pop$. Denote the new sample point generated from this evolutionary sampling strategy as $x$, and both $x$ and $x_{CB}$ are updated if certain conditions are satisfied. To summarize, $x_i$ is initialized to be $x_1$ at the beginning, then the following steps are performed for each $x_j \in Pop$:

1. randomly pick $x_a, x_b, x_c \in Pop$ different from $x_j$;
2. mutation is carried out by generating $y = x_d + D(x_b - x_c)$, where the matrix $D \in \mathbb{R}^{n \times n}$ is a diagonal matrix with each diagonal entry being a random value in $[0,1]$. That is, $D = \text{diag}\{d_1, \ldots, d_n\}$, where $d_i \sim U(0,1)$, $i = 1, \ldots, n$. Each entry $y_i$ of $y$ is restricted to be within the feasible domain by setting $y_i = \min(y_i, \bar{y}_i)$ and $y_i = \max(y_i, \underline{y}_i)$ where $\bar{y}_i, \underline{y}_i$ represent the upper and lower bounds of $y_i$, respectively;
3. crossover is carried out by triggering one of the two events $E_1 : \{z_i = x_i\}$ or $E_2 : \{z_i = y_i\}$ for all $i = 1, \ldots, n$; event $E_1$ happens with probability 0.8 and event $E_2$ happens with probability 0.2.
   The new extremum seed $z$ collects $z_i$'s as its entries;
4. evaluate $f(z), f(x_j)$ and update $x_j = z$ if $f(z) < f(x_j)$;
5. update $x_{CB}$ as $x_j$ if $f(x_j) < f(x_{CB})$;
6. update $x_i$ as $x_j$ if $f(x_j) < f(x_i)$.

The above six steps summarize how the evolutionary sampling strategy works as part of the proposed sampling technique. After obtaining $x_i$, the local method is conducted with $x_i$ being the initial point. Denote the above procedures as

$$x_{i+1}^* = \text{Evo}(Pop).$$

In summary,

$$x_{i+1}^* = \begin{cases} \text{Tunnelling (} x_{CB}, \tilde{x} \text{)}, & \text{Equation (1) is satisfied} \\ \text{Evo}(Pop), & \text{otherwise.} \end{cases}$$

Hence, the entire sampling is a hybrid of these two as declared at the beginning of this subsection. The pseudocode for the evolutionary sampling strategy is provided as Algorithm 2 in the online supplemental data.

2.2. Four stopping criteria

Martingale theory is a theory that studies the trends of stochastic processes. By using it to analyse the trends of $f_i^{*\prime}$'s and $w_i$'s, evidence on whether the global minimum has already been found can be obtained; when the evidence is strong enough, the searching process can be terminated. Therefore, in this subsection, four stopping criteria used in MGO-TEM are proposed using the statistics of $f_i^{*\prime}$'s and $w_i$'s based on the application of martingale theory, and they are used to decide whether the searching process should continue or not. The combined usage of the stopping criteria will enhance
the probability of obtaining the global minima upon termination. Their analytical derivations are provided in Section 4.

**Stopping Criterion 1.** Based on Proposition 4.1 (which guarantees that the probability that \( f^*_t \) has reached the \( \epsilon \)-neighbourhood of \( \mathbb{E}[f^*_t] \) is larger than \( 1 - \epsilon \)), the following stopping criterion is proposed:

\[
\left| \frac{1}{t} \sum_{s=1}^{t} f^*_s - \frac{1}{r} \sum_{r=1}^{t} \frac{1}{r} \sum_{s=1}^{r} f^*_s \right| \leq \epsilon^2, \tag{S1}
\]

where \( \epsilon \in (0,1) \) is a user specified value and \( t, r \) are the running indexes of \( f^*_t \). Within the absolute value sign is the difference between the sample mean and the mean of temporal means. By the strong law of large numbers, these two mean processes both converge to the true mean, which is unique. Hence, they converge to each other. However, the sample mean process is biased when the number of samples is small, which alerts us to the fact that this stopping criterion alone is not enough.

**Stopping Criterion 2.** Based on Proposition 4.3 (which guarantees that the probability of finding a better local minimum is lower than \( \delta \)), the following stopping criterion is proposed:

\[
1 - \frac{1}{t w_{t-1}} \sum_{s=1}^{t} w_s \leq \delta, \tag{S2}
\]

where \( \delta \in (0,1) \) is another user specified value. Since \( w_t \) is monotonically non-decreasing, therefore the quotient will gradually approach one as \( t \) increases. Hence, this stopping criterion supervises the maturity of the running minimum process. That is, when the running minimum is mature enough \( (i.e. w_t \) does not change any more), the difference on the left-hand side should be small.

**Stopping Criterion 3.** Based on Proposition 4.5 (which controls the quality of the obtained local minimum), the following stopping criterion is proposed:

\[
\exp(-f^*_t) \geq w_t - w_1 + e^{-1}. \tag{S3}
\]

Different from the first two stopping criteria, this one is memoryless, which controls the quality of the newly obtained local minimum since the right-hand side is monotonically non-decreasing.

When performing global optimization using MGO-TEM, the satisfaction of (S1)–(S3) simultaneously should be required before termination. However, these three stopping criteria are stochastic, which may require a huge amount of computational resources. Hence, the following deterministic stopping criteria are also proposed.

**Stopping Criteria 4.** In addition to the above three stopping criteria, the following deterministic stopping criteria based on the discussions in Section 4.5 are proposed:

\[
R_a \geq \frac{1 - e^{-1}}{\delta}, \tag{S4a}
\]

\[
R_b \geq \frac{1}{\epsilon^2} + \frac{1 - e^{-1}}{\delta} - 1, \tag{S4b}
\]

where \( R_a, R_b \) are the minimal number of consecutive \( w_t \)’s that do not change during the optimization history. In practice, MGO-TEM will terminate when either stopping criterion (S4b) is satisfied or stopping criteria (S1)–(S3) and stopping criterion (S4a) are satisfied. The simultaneous satisfaction of stopping criteria (S1)–(S3) and stopping criterion (S4a) can avoid premature termination but can be time consuming in reality. Therefore, stopping criterion (S4b) can be applied when necessary.

This set of stopping criteria provides a comprehensive way to terminate the searching process. Compared with the usual stopping criteria, such as a maximum number of function evaluations or
the vanishing of the step size or gradient, the proposed stopping criteria will enhance the efficiency of the searching procedure and the probability of obtaining the global minima upon termination.

### 2.3. A test series

With the pseudocodes provided in the online supplemental data, MGO-TEM is implemented in MATLAB® 2017b. The Whitley 5D test function is solved by MGO-TEM with parameters $\epsilon = 0.3$ and $\delta = 0.06$. The optimization histories of the test series are shown in Figure 1 and Figure S1 in the online supplemental data. In both figures, there are 12 subfigures, in which the first 10 subfigures form 5 pairs. Each pair consists of two subfigures where the one above shows the process of each stopping criterion followed by the logic state indicating whether the stopping criterion is fired or not in each iteration. When the process enters the stopping zone (the unshaded area in each subfigure), the corresponding stopping criterion is fired, and the logic state is shown in the bar chart. The eleventh subfigure shows the logic state of whether MGO-TEM should terminate and the bottom subfigure shows the optimization history of the local minima found and the running minimum. There are several interesting observations from these two figures:

- the optimization history in Figure 1 terminates owing to the activation of stopping criteria (S1), (S2), (S3) and (S4);
- the optimization history in Figure S1 in the online supplemental data terminates owing to the activation of stopping criterion (S4b);
- the stopping bound of stopping criterion (S3) will vary once a better local minimum is found, which adaptively controls the quality of the local minimum obtained;
- two optimization histories both find high quality local minima (i.e. close to zero) upon termination;
- the 2nd, 6th, 15th, 17th, 18th and 25th samples in Figure 1 show that stopping criterion (S3) controls the quality since it is not fired when the local minima found are of low quality;
- the stopping criterion (S3) in Figure S1 in the online supplemental data is difficult to satisfy and thus MGO-TEM terminates owing to stopping criterion (S4b). This shows that MGO-TEM can avoid slow convergence, which enhances its computational efficiency.

### 3. Experiments

In this section, a complete set of experimental results is provided. The proposed MGO-TEM is compared with another 10 algorithms: GA, GS, SA, PSO, MS, PS, CMA-ES, DÉ, DIRECT and BOA. This set of algorithms covers broad benchmarks in the literature of global optimization algorithms that have been shown to be effective. The test platform is MATLAB 2017b installed on a workstation with an i7-2600 CPU. The performance of the algorithms is compared by running them on a set of 55 single-objective test functions. The list of test functions are provided in Table 1 and their mathematical definitions are given in the last section of the online supplemental data. To evaluate the performance of each algorithm on each test function, the performance measures proposed in Serani et al. (2016) are used. The performance measures are defined as follows:

$$
\Delta_x = \sqrt{\frac{1}{n} \sum_{i=1}^{n} \left( \frac{x_{i,\text{min}} - x_{i,\text{min}}^*}{R_i} \right)^2}, \quad \Delta_f = \frac{f_{\text{min}} - f_{\text{min}}^*}{f_{\text{max}} - f_{\text{min}}^*}, \quad \Delta_t = \sqrt{\frac{\Delta_x^2 + \Delta_f^2}{2}},
$$

where $R_i$ is the range of coordinate values of the design variable along the $i$th dimension, $x_{i,\text{min}}$ is the coordinate value of the minimizer found along the $i$th dimension while $x_{i,\text{min}}^*$ is that of the global minimizer, $f_{\text{min}}$ is the local minimum found and $f_{\text{min}}^*, f_{\text{max}}^*$ are the global minimum and global maximum.
Figure 1. Optimization history on the Whitley 5D test function where MGO-TEM terminated owing to the activation of stopping criteria (S1), (S2), (S3) and (S4a).
Table 1. Single-objective test functions used in this article.

| Test function | Name                          | Dimension, \( n \) | Bounds \( [l_i, u_i], i = 1, \ldots, n \) | Minimum, \( f_{\text{min}} \) |
|---------------|-------------------------------|---------------------|---------------------------------------------|-------------------------------|
| \( f_1 \)     | Sphere                        | 2                   | \([-5, 4]^n\)                               | 0.000                         |
| \( f_2 \)     | Freudenstein–Roth             | 2                   | \([-5, 5]^n\)                               | 0.000                         |
| \( f_{3,4,5} \)| Ackley                        | 5, 10, 30           | \([-5, 4]^n\)                               | 0.000                         |
| \( f_6 \)     | Three-Hump Camel Back         | 2                   | \([-5, 4]^n\)                               | 0.000                         |
| \( f_7 \)     | Quartic                       | 2                   | \([-10, 10]^n\)                             | -0.352                        |
| \( f_8 \)     | Beale                         | 2                   | \([-4.5, 4.5]^n\)                           | 0.000                         |
| \( f_9 \)     | Shubert Penalty 1             | 2                   | \([-10, 10]^n\)                             | -186.731                      |
| \( f_{10} \)  | Booth                         | 2                   | \([-10, 10]^n\)                             | 0.000                         |
| \( f_{11} \)  | Goldstein–Price               | 2                   | \([-2, 2]^n\)                               | 3.000                         |
| \( f_{12} \)  | Rosenbrock                    | 2                   | \([-100, 100]^n\)                           | 0.000                         |
| \( f_{13} \)  | Schaffer n.2                  | 2                   | \([-100, 90]^n\)                            | 0.000                         |
| \( f_{14} \)  | Easom                         | 2                   | \([-100, 100]^n\)                           | -1.000                        |
| \( f_{15} \)  | Test Tube Holder              | 2                   | \([0, 10]^n\)                               | -10.872                       |
| \( f_{16} \)  | Treccani                      | 2                   | \([-1, 4]^n\)                               | 0.000                         |
| \( f_{17} \)  | Tripod                        | 2                   | \([-100, 100]^n\)                           | 0.000                         |
| \( f_{18,19} \)| Exponential                   | 2,4                 | \([-9, 7]^n\)                               | 1.000                         |
| \( f_{20,21} \)| Styblinski–Tang               | 2,4                 | \([-5, 5]^n\)                               | -39.166                       |
| \( f_{22,23} \)| Cosine Mixture                | 2,4                 | \([-1, 0.5]^n\)                             | -0.1n                         |
| \( f_{24} \)  | Hartman n.3                   | 3                   | \([0, 1]^n\)                                | -3.860                        |
| \( f_{25,26,27,28} \)| 5' loc. minima (Levy)        | 2,5,10,20           | \([-10, 10]^n\)                             | 0.000                         |
| \( f_{29,30,31,32} \)| Griewank                     | 2,5,10,20           | \([-9, 7]^n\)                               | 0.000                         |
| \( f_{33,34,35,36} \)| Alpine                     | 2,5,10,20           | \([-9, 7]^n\)                               | 0.000                         |
| \( f_{37,38,39,40} \)| Quintic                    | 2,5,10,20           | \([-2, 1]^n\)                               | 0.000                         |
| \( f_{41,42,43,44,45} \)| Dixon–Price                 | 2,5,10,25,50        | \([-1, 1]^n\)                               | 0.000                         |
| \( f_{46} \)  | Shekel n.5                    | 4                   | \([0, 10]^n\)                               | -10.153                       |
| \( f_{47,48,49} \)| Powell                     | 8,16,24             | \([-4, 5]^n\)                               | 0.000                         |
| \( f_{50,51} \)| Rastrigin                     | 5,10                | \([-5.12, 5.12]^n\)                         | 0.000                         |
| \( f_{52,53} \)| Schwefel                      | 5,15                | \([-500, 500]^n\)                           | 0.000                         |
| \( f_{54} \)  | Salomon                       | 5                   | \([-100, 100]^n\)                           | 0.000                         |
| \( f_{55} \)  | Whitley                       | 5                   | \([-10, 10]^n\)                             | 0.000                         |

of \( f \), respectively. Since the performance of the algorithms may vary from one test run to another, each algorithm runs 400 times on each test function and the median value is reported as representative of the performance of the algorithm on the test function (Vasile, Minisci, and Locatelli 2011). In total, 55 median values are reported for each algorithm. To ensure results are of practical values, the maximum number of function evaluations for each test function is set to be \( 500n \). All algorithms are fine-tuned according to the literature and testing experience. The parameter settings and the tuning method reference for each algorithm are summarized in the following list.

1. For MGO-TEM, two parameters are set as \( \epsilon = 0.1 \), \( \delta = 0.005 \) and the active-set method is chosen as the local method.
2. For GA, the population size and the number of generations are set to be \( \lceil 10\sqrt{n} \rceil \) and \( \lfloor 50\sqrt{n} \rfloor \), respectively. The crossover fraction is 0.8 and the mutation is Gaussian with the variance being 0.05 (Lockett and Miikkulainen 2014).
3. GS generates a bunch of trial points using the scatter search algorithm and decides whether to start a local method from each trial point. The number of trial points of GS is set to be \( 34n \) while the number of stage-one points is \( 3n \), and the local method is set to be the active-set method (Ugray et al. 2007).
4. The annealing function of SA is configured to be Boltzmann with the reanneal interval being 100 (Lockett and Miikkulainen 2014).
5. PSO is set to have the population size \( 10n \), the inertia weight is randomly chosen in the interval \([0, 1]\) and the learning rates are set to be 1.494 with the constriction factor being 0.729 (Serani et al. 2016; Wang, Tan, and Liu 2018; Sengupta, Basak, and Peters 2019).
(6) MS starts a local method from uniformly spread seeds inside the feasible domain. The number of seeds is set to be $20n$ and the local method is the active-set method (Ugray et al. 2007).

(7) Pattern search looks for a minimum based on an adaptive mesh that, in the absence of linear constraints, is aligned with the coordinate directions. The mesh contraction factor is set to be 0.5 while the mesh expansion factor is 2. The poll method uses a maximal positive basis of $2n$ (Weiming Li et al. 2014; Suresh Kumar, Nagesh Kumar, and Madichetty 2017).

(8) The population size of CMA-ES is set to be $(4 + \lfloor 3 \log(n) \rfloor)$. In each generation, 50% of the population is used to update the joint normal distribution (Hansen and Ostermeier 2001; Lockett and Miikkulainen 2014). Furthermore, when the covariance matrix is not positive definite, CMA-ES will be restarted (Auger and Hansen 2005; Hansen 2009).

(9) The population size of DE is set to be $10n$ and the crossover rate is set to be 0.2 while the mutation range is set to be [0.2, 0.9], according to the suggestion in Price, Storn, and Lampinen (2014).

(10) The DIRECT method generates a dense set of points over the feasible set and, at each iteration, some ‘promising’ hyperintervals are selected and further partitioned. The number of iterations is set to be 50 (Liuzzi, Lucidi, and Piccialli 2010).

(11) As for the BOA, it is usually used for hyperparameter tuning in the machine learning tasks. It builds a surrogate model for the objective and then uses an acquisition function defined from this surrogate model to decide where to sample. The acquisition function is chosen to be expected-improvement while the kernel function is the radial basis function (Frazier 2018).

Moreover, after each generation of GA, PSO, CMA-ES and DE, an additional local method is applied to obtain a local minimum according to the suggestions in Garg (2010) and Neri and Cotta (2012). The local method is the active-set method too.

After running each algorithm on each test function, the results of $\Delta_x$, $\Delta_y$, $\Delta_t$ are collected and the corresponding box plots are shown in Figure 2. From the box plots, it can be concluded that, statistically, MGO-TEM performs better than all the other 10 algorithms both in terms of $\Delta_x$ and $\Delta_f$ and therefore has better $\Delta_t$. To compare the performance of the algorithms over the test functions, bar plots are presented in Figure 3 showing the distribution of the ‘best’ algorithm considering $\Delta_x$, $\Delta_f$ and $\Delta_t$.

Figure 2. Box plots of the performance of 11 algorithms on 55 test functions considering $\Delta_x$, $\Delta_f$ and $\Delta_t$. The closer a box plot is to zero, the better the performance.
Here, an algorithm is the ‘best’ among all the algorithms on a test function if it has the lowest $\Delta_x$, $\Delta_f$ and $\Delta_t$. The higher the bar suggests that the corresponding algorithm performs the best on more test functions. Given that the performance of an algorithm usually depends on the dimension of the problem, Figure 3 presents results separately for low-dimensional problems (with dimension 2D-3D), medium-dimensional ones (with dimension 4D-10D), and high-dimensional ones (with dimension 15D-50D). From the bar plots, it is clear that MGO-TEM is superior to all the other algorithms for low-dimensional and medium-dimensional problems. As for high-dimensional ones, MGO-TEM is still the best considering $\Delta_f$ and $\Delta_t$ although GS and PS are competitive in terms of $\Delta_x$. This is because MGO-TEM terminates at a local minimizer that can be away from the global one, while PS and GS terminate in any neighbourhood of the global minimizer that may be closer to it but have less competitive function values.

4. Derivations of the stopping criteria

The derivation of the stopping criteria relies on the application of martingale theory. Martingale theory analyses the evolution of stochastic processes. In MGO-TEM, the local method $X_t$ is regarded as a random variable that takes values in $\{f^i, i = 1, 2, \ldots, N\}$. Hence, the sequence $X_t$, $t = 1, 2, 3, \ldots$, can be regarded as a stochastic process denoted by $(X_t)_{t \geq 1}$, or simply $(X_t)$. Further, let $W_t := \exp(-\min_{s \leq t} X_s)$, and the stochastic process $(W_t)$ is the process of the running minimum. The following derivations will be conducted based on $(X_t)$ and $(W_t)$.

4.1. The tunnelling process

During the searching process, given the feasible domain $Q$ and that the objective function is not changing, then $X_t$’s are a sequence of independent and identically distributed (i.i.d.) random variables. For the purpose of improving exploitation in global optimization, new samples are spread in sub-domains that are likely to have smaller function values than the expectation of the local minima. With limited information about the function and its domain, a sample is spread in a sub-domain
Q_{\text{sub}} \subseteq Q \text{ satisfying}

\[
Z_t := \sum_{s=1}^{t} M_s = \sum_{s=1}^{t} \left( X_s - \max\{E[X_1], 2E[X_1] - E[X_1|Q_{\text{sub}}]\} \right),
\]

which is a super-martingale, where $E[X_1|Q_{\text{sub}}]$ is the expectation of $X_1$ conditioned on the sub-domain $Q_{\text{sub}}$. By the construction of $Z_t$, if $Q_{\text{sub}}$ satisfies $E[X_1|Q_{\text{sub}}] < E[X_1]$, then $E[M_s] < 0$ for every $s$ and therefore the conditional expectation of $Z_t$ has a decreasing trend, which makes it a super-martingale. That is, if a new sample is spread in such an $Q_{\text{sub}}$, the obtained local minimum is expected to be better than the expectation of all the local minima.

During the tunnelling process, two local minimizers are used to construct $Q_{\text{sub}}$. Furthermore, temporal means are used to estimate $E[X_1]$ and $E[X_1|Q_{\text{sub}}]$. Then, whether to sample in $Q_{\text{sub}}$ is decided by comparing two estimated values, which corresponds to the procedure where new samples will be spread in $Q_{\text{sub}}$ if Equation (1) is satisfied.

\subsection*{4.2. The $(X_t)$ process and its stopping criterion}

The stopping criterion for the convergence of $X_t$ is first studied and designed.

\textbf{Proposition 4.1:} If $X_t$ is a non-negative random variable where $E[X_t] < \infty$ and satisfies $E[|X_t - E[X_t]|] \leq \epsilon^2$, then $P(|X_t - E[X_t]| \geq \epsilon) \leq \epsilon$.

\textbf{Proof:} By Markov’s inequality,

\[
P(|X_t - E[X_t]| \geq \epsilon) \leq \frac{1}{\epsilon} E[|X_t - E[X_t]|] \leq \epsilon, \quad \text{for all } \epsilon > 0.
\]

Then, if

\[
E[|X_t - E[X_t]|] \leq \epsilon^2, \quad (2)
\]

it directly obtains that $P(|X_t - E[X_t]| \geq \epsilon) \leq \epsilon$. Hence, Equation (2) is a sufficient condition for the probability that $X_t$ has reached the $\epsilon$-neighbourhood of $E[X_t]$ to be larger than $1 - \epsilon$. \hfill \blacksquare

Given that, in practice, $X_t$‘s are i.i.d., the following condition is supervised:

\[
|E[X_t] - C[X_t]| \leq \epsilon^2, \quad (3)
\]

where $C[X_t]$ is the mean of the temporal means of $X_t$. This can be done as follows. By the strong law of large numbers, a good estimator for $m_t := E[X_t]$ is the temporal mean of the observations, written as $\hat{m}_t = \frac{1}{t} \sum_{r=1}^{t} X_r$. Similarly by the strong law of large numbers, the estimator for $c_t := C[X_t]$ can be given by $\hat{c}_t = \frac{1}{t} \sum_{r=1}^{t} \hat{m}_r = \frac{1}{t} \sum_{r=1}^{t} \frac{1}{r} \sum_{s=1}^{r} X_s$. Substituting the above two estimators into Equation (3) yields

\[
\left| \frac{1}{t} \sum_{s=1}^{t} X_s - \frac{1}{t} \sum_{r=1}^{t} \frac{1}{r} \sum_{s=1}^{r} X_s \right| \leq \epsilon^2,
\]

which is exactly the stopping criterion (S1).
4.3. The \((W_t)\) process and its stopping criterion

**Theorem 4.2:** If \(Y\) is an integrable non-negative random variable (i.e. \(Y \geq 0\) almost surely and \(\mathbb{E}[Y] < \infty\)), then for any \(\lambda > 0\), \(\mathbb{P}[Y > \lambda] \leq \mathbb{E}[\mathbb{I}_{\{Y \geq \lambda\}} Y] / \lambda\), which is a variant of the Markov inequality.

**Proof:** Using the Markov inequality yields

\[
P(Y > \lambda) = \mathbb{E}[\mathbb{I}_{\{Y > \lambda\}} Y] \leq \mathbb{E}
\left[
\frac{Y}{\lambda}
\right]
= \frac{1}{\lambda} \mathbb{E}[\mathbb{I}_{\{Y > \lambda\}} Y],
\]

which ends the proof.

**Proposition 4.3:** For the non-negative sub-martingale \((W_t)\) and \(\lambda > 0, \delta > 0\), if \(\mathbb{E}[\mathbb{I}_{\{W_t > \lambda\}} W_t] / \lambda \leq \delta\), then \(\mathbb{P}[W_t > \lambda] \leq \delta\).

**Proof:** According to Theorem 4.2, it is true that \(\mathbb{P}[W_t > \lambda] \leq \mathbb{E}[\mathbb{I}_{\{W_t > \lambda\}} W_t] / \lambda\). Then, if \(\mathbb{E}[\mathbb{I}_{\{W_t > \lambda\}} W_t] / \lambda < \delta\), it directly obtains that \(\mathbb{P}[W_t > \lambda] \leq \delta\). Hence, it guarantees that the probability of the monotonically increasing sequence \(W_t\) not increasing any more is larger than \(1 - \delta\).

According to Proposition 4.3, choosing \(\lambda = W_{t-1}\) (the outcome of \(W_{t-1}\)) yields \(\mathbb{P}[W_t > w_{t-1}] \leq \mathbb{E}[\mathbb{I}_{\{W_t > w_{t-1}\}} W_t] / w_{t-1}\). For the purpose of application, the above probability is expected to be small, representing that the likelihood of \(W_t\) is still increasing is small. A sufficient condition for \(\mathbb{P}[W_t > w_{t-1}] \leq \delta\) is

\[
\frac{\mathbb{E}[\mathbb{I}_{\{W_t > w_{t-1}\}} W_t]}{w_{t-1}} \leq \delta.
\]

Then, the quantity \(\mathbb{E}[\mathbb{I}_{\{W_t > w_{t-1}\}} W_t]\) needs to be estimated using the \(W_{t-1}\)’s. However, since the \(W_t\)’s are not i.i.d., the temporal mean cannot be used. Instead, this value is estimated as follows:

\[
\mathbb{E}[\mathbb{I}_{\{W_t > w_{t-1}\}} W_t] = 0 \cdot \mathbb{P}({W_t = w_{t-1}}) + 0 \cdot \mathbb{P}({W_t < w_{t-1}}) + \mathbb{E}[\mathbb{I}_{\{W_t = w_{t-1}\}} W_t]
\]

\[
= w_{t-1} - w_{t-1} \cdot \mathbb{P}({W_t = w_{t-1}}) + w_{t-1} \cdot \mathbb{P}({W_t < w_{t-1}})
\]

\[
+ (w_{t-1} - \eta) \cdot \mathbb{P}({W_t > w_{t-1}}),
\]

where \(\eta \in [w_{t-1}, 1]\) is a value representing the expected value of \(W_t\) over \(\{W_t > w_{t-1}\}\).

Since \(\lim_{t \to \infty} \mathbb{P}({W_t = w_{t-1}}) = 1\), therefore \(\mathbb{E}[\mathbb{I}_{\{W_t > w_{t-1}\}} W_t]\) will converge to zero. Hence, \(\psi := \mathbb{E}[\mathbb{I}_{\{W_t > w_{t-1}\}} W_t]\) needs to be estimated based on the history of the \(W_t\)’s. Such an estimator should be asymptotically unbiased, that is \(\mathbb{E}[\hat{\psi}_t] = \psi = 0\) as \(t\) increases to \(+\infty\). To this end, the estimator is chosen as \(\hat{\psi}_t = W_{t-1} - \frac{1}{t} \sum_{s=1}^{t} W_s\). The above estimator is asymptotically unbiased since

\[
\mathbb{E}[\hat{\psi}_t] - \psi = \mathbb{E}
\left[
\frac{1}{t} \sum_{s=1}^{t} (W_{t-1} - W_s)
\right] - \mathbb{E}[\mathbb{I}_{\{W_t > w_{t-1}\}} W_t]
\]

\[
= \mathbb{E}
\left[
\frac{1}{t} \sum_{s=1}^{t \wedge \max\{s:W_{t-1} > W_s\}} (W_{t-1} - W_s)
\right] - \mathbb{E}[\mathbb{I}_{\{W_t > w_{t-1}\}} W_t]
\]

\[
= \mathbb{E}
\left[
\frac{1}{t} \sum_{s=1}^{t \wedge \max\{s:W_{t-1} > W_s\}} (W_{t-1} - W_s) - \mathbb{I}_{\{W_t > w_{t-1}\}} W_t
\right]
\]
is zero as \( t \) increases to \(+\infty\) since the \( W_t \)'s are non-decreasing and bounded above by one, and therefore \( W_s = 1 \) for all \( s \geq N \) where \( N \) is a large enough iteration number. Substituting the above estimator into the sufficient condition, the stopping criterion \( (S2) \) for the sub-martingale \( (W_t) \) is obtained as
\[
1 - \frac{1}{\tau_{W_{t-1}}} \sum_{s=1}^{t} W_s \leq \delta.
\]

### 4.4. Stopping criterion \( (S3) \) controls the quality

To control the quality of the obtained local minimum, the stopping criterion \( (S3) \) is proposed, which relies on the application of Doob's Upcrossing Lemma (Steele 2001). The lemma is quoted below.

**Lemma 4.4 (Doob's Upcrossing Lemma):** Given the probability space \((\Omega, A, \mathbb{P})\), let \((Y_t)\) be a sub-martingale, then for every two real numbers \( x < y \), where \( U_{[0,T]} \subseteq \mathbb{T} \) \((x, y, Y_t)\) is the number of upcrossings of the closed interval \([x, y]\) by \( s \mapsto Y_s \) up to time \( t \in [0, T] \subseteq \mathbb{T} \), it holds that
\[
(y - x) \mathbb{E}[U_{[0,T]} \subseteq \mathbb{T}(x, y, Y_{s \leq t})] \leq \mathbb{E}[(Y_t - x)^+] - \mathbb{E}[(Y_0 - x)^+],
\]
where \([0, T]\) denotes the integer values in the interval \([0, T]\) and the number of upcrossings means the number of occurrences of \( Y_t \geq y \) during the discrete sampling steps.

Based on the lemma, the following proposition can be obtained.

**Proposition 4.5:** Given the stochastic process \((X_t)\) and an interval \([0, y]\), \( y > 0 \), if \( y = \mathbb{E}[W_t] - \mathbb{E}[W_1] \), then \( \mathbb{E}[U_{[1,T]} \subseteq \mathbb{T}(0, y, \exp(-X_{s \leq t}))] \leq 1 \).

**Proof:** In MGO-TEM, the sub-martingale is \((W_t)_t\), where each sample path is bounded between zero and one. Therefore \( x = 0 \), and Doob's Upcrossing Lemma is reduced to \( y \mathbb{E}[U_{[1,T]} \subseteq \mathbb{T}(0, y, W_{s \leq t})] \leq \mathbb{E}[W_1] - \mathbb{E}[W_1] \), which can be further extended to
\[
y \mathbb{E}[U_{[1,T]} \subseteq \mathbb{T}(0, y, \exp(-X_{s \leq t}))] \leq y \mathbb{E}[U_{[1,T]} \subseteq \mathbb{T}(0, y, W_{s \leq t})] \leq \mathbb{E}[W_1] - \mathbb{E}[W_1].
\]
In this inequality, \( \mathbb{E}[U_{[1,T]} \subseteq \mathbb{T}(0, y, \exp(-X_{s \leq t}))] \) represents the expected number of upcrossings of the interval \([0, y]\) by the transformed \(X_t\)'s. Hence it is required that this expected number is greater or equal to one given a proper \(y\) (this is a natural intention for finding a smaller local minimum). Setting \( y \leq \mathbb{E}[W_1] - \mathbb{E}[W_1] \) will satisfy that
\[
\mathbb{E}[U_{[1,T]} \subseteq \mathbb{T}(0, y, \exp(-X_{s \leq t}))] = 1,
\]
which ends the proof.

Hence, the quality of the newly obtained local minimum can be guaranteed by setting \( y = \mathbb{E}[W_t] - \mathbb{E}[W_1] \) and then observing whether the current \(X_t\) exceeds \(y\). By the above lemma, such an \(X_t\) exists in expectation. Moreover, given that the temporal mean of the \(W_t\)'s cannot be used to estimate \(\mathbb{E}[W_1]\) or \(\mathbb{E}[W_1]\), since they are not i.i.d., therefore \(y = W_t - W_1\) is used in MGO-TEM. Hence the stopping criterion \( (S3) \) is given as \(\exp(-X_t) \geq W_t - W_1 + e^{-1}\). According to the experiments, this stopping criterion works very well in practice.

The choice of \(W_t - W_1\) is not close to what the upcrossing lemma suggests but it is a proper choice when MGO-TEM runs sequentially. Such a choice is conservative since it is the largest difference between the values from the history of the \(W_t\)'s, meaning that a high requirement on the quality of the obtained local minimum is set. Moreover, since \(W_t\) is determined by the best local minimum ever found, therefore at least one local minimum satisfies the requirement. Last but not least, if MGO-TEM can run in a parallel computing scenario, then \(\mathbb{E}[W_t]\) and \(\mathbb{E}[W_1]\) can be effectively estimated since multiple initial points are spread in the same iteration.

### 4.5. Complexity analysis and stopping criteria \((S4a)\) and \((S4b)\)

#### 4.5.1. Complexity analysis of stopping criterion \((S1)\) and stopping criterion \((S2)\)

Given that the mean process of the temporal means of the \(X_t\)'s will converge faster than the sample mean process, only the complexity of the sample mean process needs to be discussed. The behaviour
of the sample mean process can be studied by analysing the difference of two arbitrary sample means. Denote the sample mean in the \((t + m)\)th iteration as \(S_{t+m}\) and the other sample mean in the \((t + n)\)th iteration as \(S_{t+n}\). Since the sample mean process \(S_t\) converges almost surely (according to the strong law of large numbers), then for all \(\varepsilon^2 > 0\) there exists large enough \(m, n, L \in \mathbb{N}\) with \(m > n > L\) such that \(|S_{t+m} - S_{t+n}| < \varepsilon^2\). To satisfy this inequality, it is required that \(\varepsilon^2 > (m - n)/(t + m)\) since

\[
\varepsilon^2 > \frac{m - n}{t + m} \geq \left| \frac{1}{t + m} \sum_{s=t+n+1}^{t+m} X_s \right| = \left| \frac{1}{t + m} \sum_{s=t+n+1}^{t+m} X_s + \frac{1}{t + n} \sum_{s=1}^{t+n} X_s - \frac{1}{t + n} \sum_{s=1}^{t+n} X_s \right| \geq \left| \frac{1}{t + m} \sum_{s=1}^{t+m} X_s - \frac{1}{t + n} \sum_{s=1}^{t+n} X_s \right| = |S_{t+m} - S_{t+n}|.
\]

Rearranging the above gives

\[
t \geq \frac{m - n}{\varepsilon^2} - m
\]

for all \(m > n > t\) and \(\varepsilon > 0\). The above shows that the total number of times the local method is called should be at least \((m - n)/\varepsilon^2 - m\) so that the sample mean process of \(X_t\) converges.

For stopping criterion (S2), since the sequence of outcomes of \((W_t)\) is monotone, thus its behaviour is studied by evaluating its distance from one. The following propositions are proposed.

**Proposition 4.6:** Given that \((X_t)\) is a positive stochastic process and \(X_t \in [0, 1]\) for all \(t\), then the process \(W_t := \exp(- \min_{s \leq t} X_s)\) has non-decreasing sample path \(\tilde{w}_t := W_t(\omega)\) for fixed \(\omega\) and \(\tilde{w}_1/\tilde{w}_t \in [e^{-1}, 1]\) for all \(t\).

**Proof:** The sample path of \(W_t\) is non-decreasing since

\[
\tilde{w}_t - \tilde{w}_{t-1} = \exp \left( - \min_{s \leq t} X_s \right) - \exp \left( - \min_{s \leq t-1} X_s \right) \geq \max \left\{ \exp(-X_t), \exp \left( - \min_{s \leq t-1} X_s \right) \right\} - \exp \left( - \min_{s \leq t-1} X_s \right) \geq 0,
\]

which means \(\tilde{w}_1/\tilde{w}_t \leq 1\) for all \(t\). Moreover, given \(\tilde{w}_t\) is non-decreasing and \(\tilde{w}_t \in [e^{-1}, 1]\) for all \(t\) according to Section 2, then \(\tilde{w}_1/\tilde{w}_t \geq e^{-1}\), and the proposition is proved. ■
Proposition 4.7: Given the sample path \( w_t \) of the sub-martingale \( (W_t) \) and, for all \( t, 1 - e^{-1} > \delta > 0 \) satisfy

\[
\frac{t}{m_t} \leq \frac{1 - e^{-1}}{1 - e^{-1} - \delta},
\]

where \( m_t \) is the number of \( w_s \)'s equal to \( w_{t-1} \), then

\[
1 - \frac{1}{tw_{t-1}} \sum_{s=1}^{t} w_s \leq \delta.
\]

Proof: According to Proposition 4.6, \( w_t \) is non-decreasing and positive and \( w_1/w_t \in [e^{-1}, 1] \) for all \( t \). Then,

\[
\frac{t}{m_t} \leq \frac{1 - e^{-1}}{1 - e^{-1} - \delta} \leq \frac{1 - w_1/w_{t-1}}{1 - w_1/w_{t-1} - \delta}.
\]

Rearranging the above inequality gives

\[
\delta \geq 1 - \frac{m_t}{t} - \frac{1}{tw_{t-1}} \sum_{s=1}^{t-m_t} w_s
\]

\[
= 1 - \frac{m_t}{t} - \frac{1}{tw_{t-1}} \left( \sum_{s=1}^{t} w_s - m_tw_{t-1} \right)
\]

\[
= 1 + \frac{m_t(w_{t-1} - w_{t-1})}{tw_{t-1}} - \sum_{s=1}^{t} w_s
\]

\[
= 1 - \frac{1}{tw_{t-1}} \sum_{s=1}^{t} w_s,
\]

where the first equality is obtained since \( w_s = w_{t-1} \) for all \( s > t - m_t \), and the proposition is proved.

From the above proposition, to satisfy stopping criterion (S2), it is sufficient that

\[
t \leq m_t \frac{1 - e^{-1}}{1 - e^{-1} - \delta}.
\]

However, since \( t \) will be at least as large as \( m_t + 1 \),

\[
m_t \frac{1 - e^{-1}}{1 - e^{-1} - \delta} \geq m_t + l,
\]

where \( l = 1, 2, \ldots \). Rearranging this inequality gives

\[
m_t \geq \frac{l(1 - e^{-1} - \delta)}{\delta}.
\]
4.5.2. Stopping criteria (S4a) and (S4b)
Combining Equation (6) with Equation (5), it is sufficient to let, for all natural numbers, \( m > n > t \) and \( l = 1, 2, 3, \ldots \) such that

\[
m_t \geq \frac{m - n}{\epsilon^2} + \frac{l(1 - e^{-1} - \delta)}{\delta}
\]

(7)

to satisfy the convergence of the sample mean process of \( X_t^+ \) and stopping criterion (S2). For practical use, \( m - n = 1, l = 1 \) are used, which yields the stopping criterion (S4b) as

\[
m_t \geq \frac{1}{\epsilon^2} + \frac{1 - e^{-1}}{\delta} - 1.
\]

The satisfaction of stopping criterion (S4b) will terminate MGO-TEM. Additionally, the stopping criterion (S4a) requires that \( m_t \geq (1 - e^{-1})/\delta \); this criterion should be used together with stopping criteria (S1), (S2) and (S3) to prevent possible premature stopping, since those three stopping criteria are stochastic.

5. Conclusions
In this article, a multistart global optimization method with tunnelling and an evolutionary strategy supervised by a martingale (MGO-TEM) is proposed. The major contributions of MGO-TEM in this article include a new sampling technique and a set of stopping criteria whose derivation is based on martingale theory. The satisfaction of these stopping criteria enhance the efficiency and the probability of obtaining the global minima upon termination. Two parameters \( \epsilon \) and \( \delta \) representing two probabilities are left as two user-specified parameters. Theoretical derivations and analysis are presented to prove the effectiveness of the proposed stopping criteria. The pseudocodes of MGO-TEM are provided in the online supplemental data and a test series is presented to show the optimization history of MGO-TEM, which illustrates the status of each stopping criterion during optimization. In the experiments, MGO-TEM is compared with another 10 very popular and well-known global optimization methods over 55 single-objective test functions. The results verify that MGO-TEM performs better than the others in terms of commonly used performance measures. There are two future research directions of MGO-TEM. First, an extension of MGO-TEM to multi-objective cases deserves further study; second, a variation of MGO-TEM that is suitable for a parallel computing scenario should be considered in the future.

Note
1. The code for MGO-TEM can be downloaded from https://github.com/CamelZrx/MGO-TEM.

Disclosure statement
No potential conflict of interest was reported by the author(s).

ORCID
Mian Li http://orcid.org/0000-0001-5184-5861

References
Ali, M. M., and A. Törn. 2004. “Population Set-Based Global Optimization Algorithms: Some Modifications and Numerical Studies.” Computers & Operations Research 31 (10): 1703–1725.
Alsumait, J. S., J. K. Sykulski, and A. K. Al-Othman. 2010. “A Hybrid GA-PS-SQP Method to Solve Power System Valve-Point Economic Dispatch Problems.” Applied Energy 87 (5): 1773–1781.
Auger, A., and N. Hansen. 2005. “A Restart CMA Evolution Strategy with Increasing Population Size.” In Proceedings of the 2005 IEEE Congress on Evolutionary Computation, Vol. 2, 1769–1776. Piscataway, NJ: IEEE.
Neri, Ferrante, and Carlos Cotta. 2012. “Memetic Algorithms and Memetic Computing Optimization: A Literature Review.” Swarm and Evolutionary Computation 2: 1–14.

Price, Kenneth, Rainer M. Storn, and Jouni A. Lampinen. 2014. Differential Evolution: A Practical Approach to Global Optimization. Berlin: Springer. https://books.google.co.uk/books?id=hakXI-dEhTkC.

Qiu, Y., H. Zhou, and Yueqin Wu. 2008. “An Importance Sampling Method Based on Martingale with Applications to Rare Event Probability.” In Proceedings of the 7th World Congress on Intelligent Control and Automation, 4041–4045. Piscataway, NY: IEEE. doi:10.1109/WCICA.2008.4593574.

Ronkkonen, J., S. Kukkonen, and K. V. Price. 2005. “Real-Parameter Optimization with Differential Evolution.” In Proceedings of the 2005 IEEE Congress on Evolutionary Computation, Vol. 1, 506–513. Piscataway, NY: IEEE. doi:10.1109/CEC.2005.1554725.

Rudolph, Günter, Fachbereich Informatik, and Ls Xi. 1997. “Convergence Rates of Evolutionary Algorithms for a Class of Convex Objective Functions.” Control and Cybernetics 26: 375–390.

Sengupta, S., S. Basak, and R. A. Peters II. 2019. “Particle Swarm Optimization: A Survey of Historical and Recent Developments with Hybridization Perspectives.” Machine Learning and Knowledge Extraction 1 (1): 157–191. doi:10.3390/make1010010.

Serani, Andrea, Cecilia Leotardi, Umberto Iemma, Emilio F. Campana, Giovanni Fasano, and Matteo Diez. 2016. “Parameter Selection in Synchronous and Asynchronous Deterministic Particle Swarm Optimization for Ship Hydrodynamics Problems.” Applied Soft Computing 49: 313–334.

Stanovov, Vladimir, Shakhnaz Akhmedova, and Eugene Semenkin. 2019. “Application of Differential Evolution with Selective Pressure to Economic Dispatch Optimization Problems.” IFAC-PapersOnLine 52 (13): 1566–1571. doi:10.1016/j.ifacol.2019.11.423.

Steele, J. Michael. 2001. Stochastic Calculus and Financial Applications. Vol. 45 in the series Stochastic Modelling and Applied Probability. New York: Springer-Verlag. doi:10.1007/978-1-4684-9305-4.

Storn, Rainer, and Kenneth Price. 1997. “Differential Evolution—A Simple and Efficient Heuristic for Global Optimization Over Continuous Spaces.” Journal of Global Optimization 11 (4): 341–359.

Suresh Kumar, L. V., G. V. Nagesh Kumar, and Sreedhar Madichetty. 2017. “Pattern Search Algorithm Based Automatic Online Parameter Estimation for AGC with Effects of Wind Power.” International Journal of Electrical Power & Energy Systems 84: 135–142. doi:10.1016/j.ijepes.2016.05.009.

Ugray, Zsolt, Leon Lasdon, John Plummer, Fred Glover, James Kelly, and Rafael Marti. 2007. “Scatter Search and Local NLP Solvers: A Multistart Framework for Global Optimization.” INFORMS Journal on Computing 19 (3): 313–484. doi:10.1287/ijoc.1060.0175.

Vasile, M., E. Minisci, and M. Locatelli. 2011. “An Inflationary Differential Evolution Algorithm for Space Trajectory Optimization.” IEEE Transactions on Evolutionary Computation 15 (2): 267–281.

Wang, Dongshu, Dapei Tan, and Lei Liu. 2018. “Particle Swarm Optimization Algorithm: An Overview.” Soft Computing 22 (2): 387–408.

Yang, Xin-She. 2014. Nature-Inspired Optimization Algorithms. 1st ed. Oxford, UK: Elsevier. https://www.elsevier.com/books/nature-inspired-optimization-algorithms/yang/978-0-12-416743-8.

Ye, Z., Y. Cao, A. Zhang, C. Jin, L. Ma, X. Hu, and J. Hu. 2019. “An Image Enhancement Optimization Method Based on Differential Evolution Algorithm and Cuckoo Search Through Serial Coupled Mode.” In Proceedings of the 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Vol. 2, 916–920. Piscataway, NY: IEEE.