

# Optimal Scaling of an Algorithmic Parameter in Restart Strategies

Lisa Schönenberger, Hans-Georg Beyer

Vorarlberg University of Applied Sciences

Research Center Business Informatics

6850 Dornbirn, Austria

E-Mail: {lisa.schoenenberger, hans-georg.beyer}@fhv.at

## Abstract

This paper investigates restart strategies for algorithms whose success depends on an algorithmic parameter  $\lambda$ . It is assumed that there exists a unique unknown optimal  $\lambda$ . After each restart  $\lambda$  is increased. The main question is whether there is an optimal strategy for choosing  $\lambda$  after each restart. To this end, possible restart strategies are classified into parameter-dependent strategy types. A loss function is introduced, that measures the wasted computational costs compared to the optimal strategy. One criterion that a viable restart strategy must satisfy is that the loss relative to the optimal  $\lambda$  is bounded. Experimental evidence demonstrates that this is not the case for all strategy types. However, for a specific strategy type, where the parameter  $\lambda$  is increased multiplicatively with an increasing constant  $\rho$ , the relative loss function has an upper bound. It will be shown, that for this strategy type there is an optimal choice for the parameter  $\rho$  that is independent of the optimal  $\lambda$ .

---

This work was supported by the Austrian Science Fund (FWF) under grant P33702-N.

© 2024 Copyright held by the owner/author(s). This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

# 1 Introduction

Restart strategies are techniques used to restart an algorithm after it has failed. Beyond simple random restarts, there are several approaches to implement such strategies. One common method is to restart the algorithm from a different start point, as it is usual for gradient strategies in nonlinear optimization. If the performance of the algorithm depends on a certain parameter, this parameter can be adjusted after each restart. It quickly becomes clear that the exact implementation of a restart strategy is highly problem-specific, therefore restrictions must be made when analyzing restart strategies. This paper focuses on algorithms whose success depends on a single parameter  $\lambda$  that is modified after each restart according to a predetermined rule. The primary goal of this study is to evaluate these rules, called strategy types, in order to identify an optimal restart strategy. Optimality criteria have already been studied for algorithms, whose success depends on the execution time, see for example [1] or [2]. For the restart strategies of parameter-dependent algorithms, optimality has not been studied so far.

The specific constraints relevant to the algorithms studied are outlined in Section 2. After a general definition of a restart strategy in Section 3, the various strategy types are introduced. Section 4 introduces the concept of loss and presents the relative loss function, with the results of related experiments. The paper concludes with a summary of the results and an outlook.

## 2 Preliminary conditions

The restart strategy (RS) considered here are applied to algorithms that satisfy the following constraints. The algorithm  $\mathbf{A}$  depends on an algorithmic parameter  $\lambda \in \mathbb{N}$ .  $\mathbf{A}$  is successful only if this algorithmic parameter exceeds a certain bound  $\hat{\lambda}$ . Formally, this means

$$\begin{aligned} \mathbf{A}(\lambda) \text{ is successful if } \lambda &\geq \hat{\lambda} \\ \mathbf{A}(\lambda) \text{ is unsuccessful if } \lambda &< \hat{\lambda}. \end{aligned} \tag{1}$$

$F_E(\lambda)$  measures the computational costs. Since evolution strategies are considered it is defined by the number of objective function evaluations that  $\mathbf{A}(\lambda)$  uses until termination. It is assumed that  $F_E(\lambda)$  increases with  $\lambda$ , so that the optimal choice is to execute the algorithm with  $\hat{\lambda}$ . Therefore,  $\hat{\lambda}$  is also called *optimal*  $\lambda$ . An additional assumption is that

$$F_E(\lambda) = g\lambda \quad (2)$$

for a constant  $g \in \mathbb{N}$ .

Algorithms that satisfy these conditions roughly are for example Evolution Strategies in multimodal landscapes. The parameter  $\lambda$  in this case is the population size. There is an interval for  $\lambda$  where a positive success probability less than 1 is possible. This interval, however, is small compared to the population size. If  $\lambda$  exceeds this interval, the success rate remains constant at one. This has been demonstrated for several multimodal test functions in [3]. The validity of the simplification  $F_E(\lambda) = g$  was shown in [4].

### 3 Restart Strategies

To approach the optimal choice of the algorithmic parameter  $\hat{\lambda}$ , which is generally unknown, a restart strategy can be used. Restart strategies (RS) are defined by an infinite sequence

$$\mathbb{R} = (\lambda_0, \lambda_1, \lambda_2, \dots), \quad \lambda_k \in \mathbb{N}, \quad (3)$$

where  $\lambda_k$  is the algorithmic parameter of the  $k$ th run. The  $k$ th run of the RS is denoted by  $\mathbf{R}_k$ .  $\mathbf{R}_k$  is stopped when a local stopping criterion is satisfied. Then, an independent algorithm  $\mathbf{R}_{k+1}$  with parameter  $\lambda_{k+1}$  is executed. This process is repeated until the algorithm is successful.

The theory of restart strategies raises the question of how to choose  $\lambda_k$ . Because of condition (1), it is clear that  $\lambda$  should be increased after each restart. In principle, there are an infinite number of restart strategies. A common choice for  $\lambda_k$  is  $\lambda_k = \lambda_0 2^k$  (see for example [5] or [6]). In this case,  $\lambda$  is increased

multiplicatively after each restart. This type of a restart strategy will be also examined in the following. Instead of base 2, other increasing factors may be used, which leads to the following restart sequence

$$\begin{aligned}\mathbb{R}^\times &= (\lambda_0, \lambda_1, \lambda_2, \dots) \\ \lambda_k &= \lceil \lambda_0 \rho^k \rceil.\end{aligned}\tag{4}$$

$\mathbb{R}^\times$  is called a *strategy type* and  $\rho$  is called *increasing constant*. Because of the assumption that  $\lambda$  increases after each restart, it can be assumed that  $\rho > 1$ .  $\rho$  does not necessarily have to be a natural number, therefore, the amount is rounded up.

For  $\mathbb{R}^\times$ -RS, the rounding of  $\lambda_k$  occurs only once at the end. Alternatively, one can consider the  $\mathbb{R}^*$ -RS where  $\lambda_k$  is determined based on the previous rounded-up values, resulting in

$$\begin{aligned}\mathbb{R}^* &= (\lambda_0, \lambda_1, \lambda_2, \dots) \\ \lambda_k &= \lceil \lambda_{k-1} \rho \rceil = \lceil \lceil \lambda_0 \rho \rceil \rho \rceil \dots \rho \rceil, \quad k \geq 1,\end{aligned}\tag{5}$$

$\mathbb{R}^\times$  and  $\mathbb{R}^*$  are multiplicative strategy types. Another type increases the population size by a constant amount, i.e.,

$$\begin{aligned}\mathbb{R}^+ &= (\lambda_0, \lambda_1, \lambda_2, \dots) \\ \lambda_k &= \lambda_{k-1} + v = \lambda_0 + kv, \quad k \geq 1, v \in \mathbb{N} \setminus \{0\}.\end{aligned}\tag{6}$$

A third type of restart strategies obeys a power law with constant  $\alpha \geq 1$  defined by

$$\begin{aligned}\mathbb{R}^\# &= (\lambda_0, \lambda_1, \lambda_2, \dots) \\ \lambda_k &= \lceil \lambda_0 (k+1)^\alpha \rceil.\end{aligned}\tag{7}$$

To illustrate the impact of the different restart strategies, exemplary runs of an Evolution Strategy with cumulative stepsize adaptation (CSA-ES) [7] on the well-known Rastrigin function are presented in Fig. 1. The exact definition of the CSA-ES and the Rastrigin function is shown in the Appendix. The figure



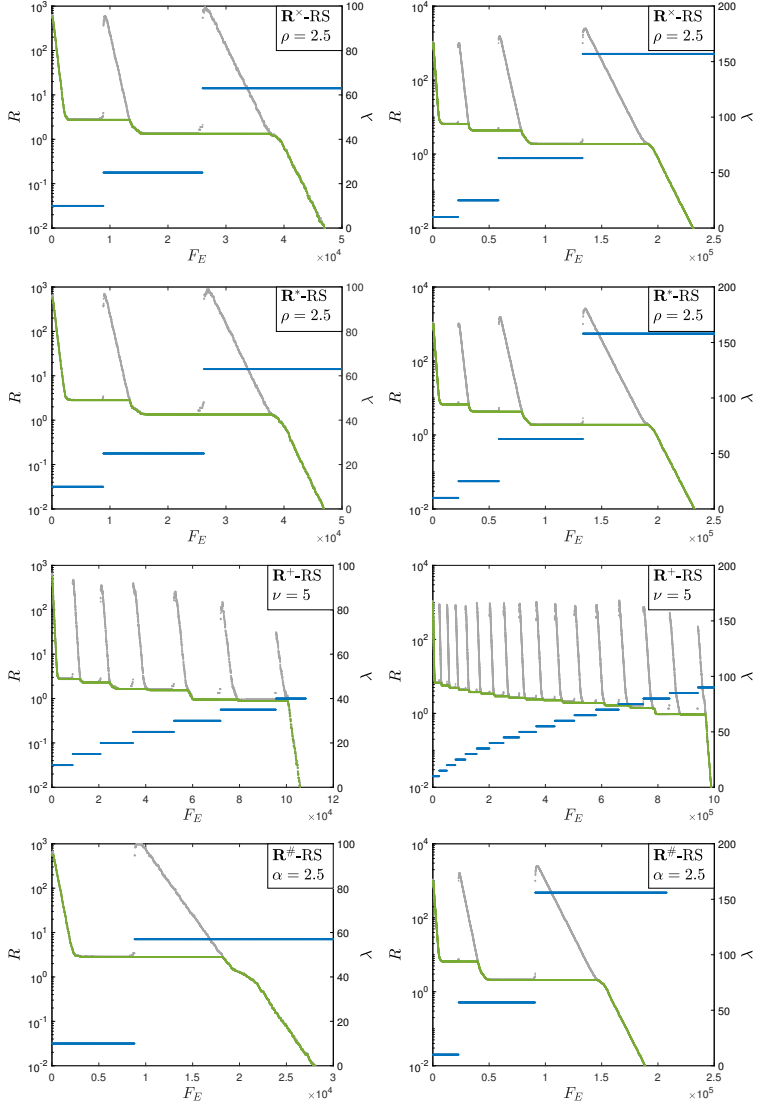


Figure 1:  $R$ -dynamics of the  $\mathbb{R}$ -RS (3) applied to the CSA-ES (Alg. 2) in the Rastrigin landscape (18) with  $\vartheta = 0.5$ ,  $N = 30$  (left plots) and  $N = 100$  (right plots). All experiments use  $\lambda_0 = 10$  as start value.  $R$  denotes the distance to the global optimizer.  $F_E$  denotes the number of function evaluations. Gray markers show the median of  $R$ , green markers show the median of the best  $R$ -value achieved up to  $F_E$ . Blue markers show the median of  $\lambda$ . The median was taken over 51 independent runs.

depicts the  $R$ -dynamics of restart strategies as a function of the number of function evaluations. The total number of function evaluations was calculated by aggregating the data across all restarts. The variable  $R$  represents the distance to the global optimizer and is depicted by the gray markers. The green markers indicate the best value of  $R$  that has been attained so far. Both are represented by the left y-axis. The blue markers show the population size  $\lambda$ , which is represented by the right y-axis. The  $\lambda$ -values represent a step function. At the jumps of this step function, a new restart begins with a larger value of  $\lambda$ . In the associated  $R$ -dynamics, one observes that a smaller distance to the optimum can be achieved after a certain number of function evaluations following the restart. This results in the emergence of kinks in the  $R$ -dynamics. The left plot employs  $N = 30$ , while the right plots use  $N = 100$ . For larger dimensions, a larger value of  $\hat{\lambda}$  is necessary to achieve success. This is visible in Fig. 1 where the number of function evaluations and the number of restarts are larger in the right figure.

A comparative analysis of the various strategy types reveals that the  $\mathbb{R}^+$ -RS requires significantly more restarts in comparison to the other types. Moreover, the number of function evaluations required to achieve success is considerably larger. However, this could be compensated for by choosing a larger value for  $v$ . The multiplicative strategy types  $\mathbb{R}^*$  and  $\mathbb{R}^\times$  exhibit no significant differences. However, it should be noted that the value of  $\lambda$  is larger for the  $\mathbb{R}^*$ -RS due to the additional rounding. This effect becomes more pronounced the more restarts are required. Interestingly, the strategy type  $\mathbb{R}^\#$  clearly has the best performance in this example.

It is not obvious whether one strategy type is better than the other. It is also not clear how to choose the increasing constants  $v$ ,  $\rho$ , and  $\alpha$  for any given strategy type. There is no optimality criterion up until now. In the following sections, the influence of the increasing constants on the number of function evaluations is investigated experimentally.

## 4 The Loss Function

When choosing the increasing constant, it is important to avoid choosing values that are too small as this will result in many restarts being necessary. Conversely, if the increasing constant is set to a very large value,  $\lambda$  will also become very large after just a few restarts. This causes  $\lambda$  to be much larger than necessary, requiring more function evaluations than necessary. The loss  $\Delta F_E$  of a restart strategy is defined by<sup>1</sup>

$$\Delta F_E(\hat{\lambda}, \rho) := \sum_{k=0}^{\hat{k}(\hat{\lambda})} F_E(\lambda_k) - F_E(\hat{\lambda}) = \left( \sum_{k=0}^{\hat{k}(\hat{\lambda})} \lambda_k - \hat{\lambda} \right) g, \quad (8)$$

where the second equality follows from condition (2).  $\hat{k}(\hat{\lambda})$  denotes the minimum number of restarts required to obtain a  $\lambda$  larger than or equal to  $\hat{\lambda}$ , i.e.,

$$\hat{k}(\hat{\lambda}) := \arg \min \{k | \lambda_k \geq \hat{\lambda}\}. \quad (9)$$

Because  $g$  is constant it can be dropped in the following considerations. Therefore, the reduced loss function

$$\mathbf{L}(\hat{\lambda}, \rho) := \sum_{k=0}^{\hat{k}(\hat{\lambda})} \lambda_k - \hat{\lambda}. \quad (10)$$

will be used in the following. (10) can be calculated numerically using Alg. 1. The update of  $\lambda$  depends on the specific strategy type.

---

### Algorithm 1.: Numerical Calculation of the Loss Function (10)

---

```

1: Initialize ( $\lambda = \lambda_0, F_E = \lambda_0, k = 0$ )
2: while  $\lambda < \hat{\lambda}$  do
3:    $k = k + 1$ 
4:    $\lambda = r(\lambda)$  ▷ update  $\lambda$ , depending on strategy type
5:    $F_E = F_E + \lambda$ 
6:  $\mathbf{L} = F_E - \hat{\lambda}$ 

```

---

<sup>1</sup>  $\rho$  is used as a substitute to indicate the dependency of  $\Delta F_E$  on the respective increasing constants.

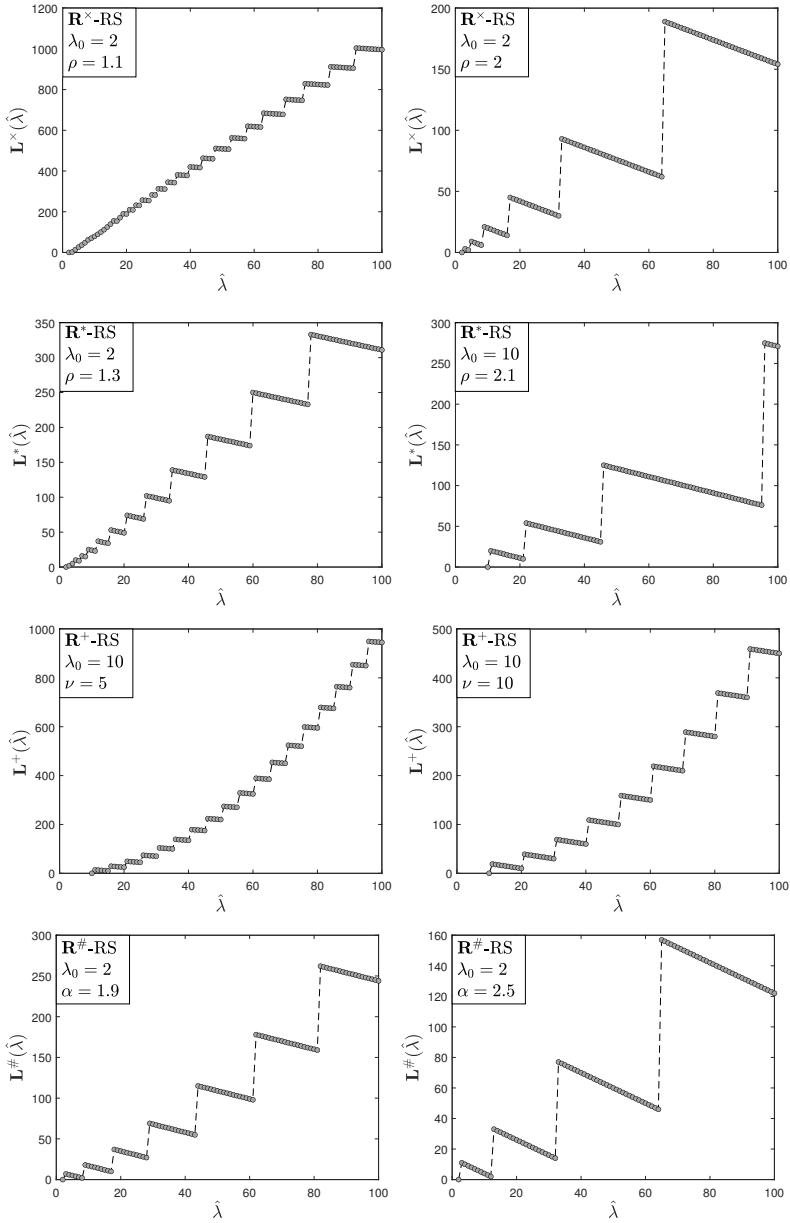


Figure 2: Loss function (10) determined numerically using Alg. 1.

Figure 2 shows loss functions as a function of  $\hat{\lambda}$  for a fixed value of the increasing constant. The markers represent the numerical solution for the loss function computed with Alg. 1 and

$$r(\lambda) = \lceil \lambda_0 \rho^k \rceil \quad \text{for } \mathbb{R}^\times \quad (11)$$

$$r(\lambda) = \lceil \lambda \rho \rceil \quad \text{for } \mathbb{R}^* \quad (12)$$

$$r(\lambda) = \lambda + v \quad \text{for } \mathbb{R}^+ \quad (13)$$

$$r(\lambda) = \lceil \lambda_0 (k+1)^\alpha \rceil \quad \text{for } \mathbb{R}^\#, \quad (14)$$

where  $k$  is the number of restarts. For all strategy types, the loss function jumps. These jumps occur at the values  $\lambda_k + 1$  where the cost of an additional restart is added. Between these jumps, the loss decreases linearly with  $\hat{\lambda}$ . The left and right plots show the same strategy type with different increasing constants. The frequency of the jumps is larger for smaller increasing constants. The magnitude of the loss varies considerably for the different increasing constants.

It can also be seen from Fig. 2 that for a fixed value of the increasing constant, the loss is unbounded of  $\hat{\lambda}$ . The larger  $\hat{\lambda}$  is, the larger the loss can be expected. Therefore, to further characterize the restart effort, it is useful to introduce the relative loss. It measures the loss relative to  $\hat{\lambda}$ . The relative loss is defined by

$$\ell(\hat{\lambda}, \rho) := \frac{\mathbf{L}(\hat{\lambda}, \rho)}{\hat{\lambda}}. \quad (15)$$

The relative loss function is shown in Fig. 3 for all strategy types. This function also has the characteristic jumps and decreases in  $\hat{\lambda}$  between these jumps. The top plots represent the  $\mathbb{R}^+$ -RS and the  $\mathbb{R}^\#$ -RS. For both cases and for sufficiently large  $\hat{\lambda}$ , the relative loss is observed to be smaller for larger increasing constants. In all cases, the relative loss goes to infinity with  $\hat{\lambda}$ . In contrast, for the multiplicative strategy types  $\mathbb{R}^\times$  and  $\mathbb{R}^*$ , represented by the bottom plots, this is not the case. The relative loss function has an upper bound. The figures demonstrate that the relative loss functions for the  $\mathbb{R}^\times$ -RS and  $\mathbb{R}^*$ -RS are nearly identical, which suggests that both strategies can be considered interchangeable.

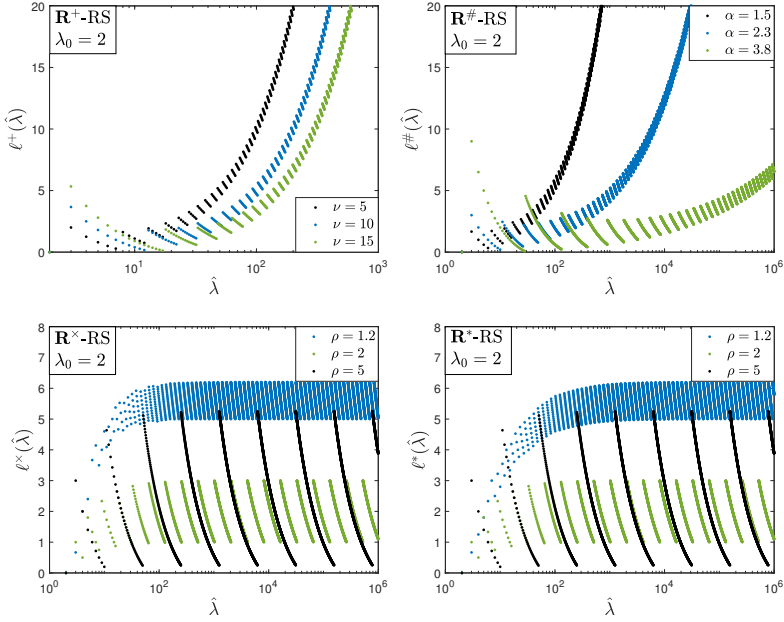


Figure 3: Relative loss function (15) determined numerically using Alg. 1.

If the relative loss has an upper bound  $\ell_{\text{up}}(\hat{\lambda}, \rho)$ , the strategy is classified as *bounded*. Consequently, the  $\mathbb{R}^\times$ -RS and the  $\mathbb{R}^*$ -RS are bounded restart strategies. In contrast, the  $\mathbb{R}^+$ -RS and the  $\mathbb{R}^\#$ -RS are not bounded.

The upper bound of the multiplicative restart strategies depends strongly on  $\rho$ . The local maxima of the relative loss functions approach a constant value as  $\hat{\lambda}$  increases. The value

$$\overline{\ell_{\text{up}}}(\rho) := \lim_{\hat{\lambda} \rightarrow \infty} \ell_{\text{up}}(\hat{\lambda}, \rho) \quad (16)$$

is called the *asymptotic upper bound* of the relative loss function. This upper bound is dependent on the increasing constant. The objective is to find the value of  $\rho$  that minimizes the asymptotic upper bound. This value is called the optimal choice of the increasing constant and is independent of  $\hat{\lambda}$ . This is explored further for the  $\mathbb{R}^\times$ -RS in Fig. 4.

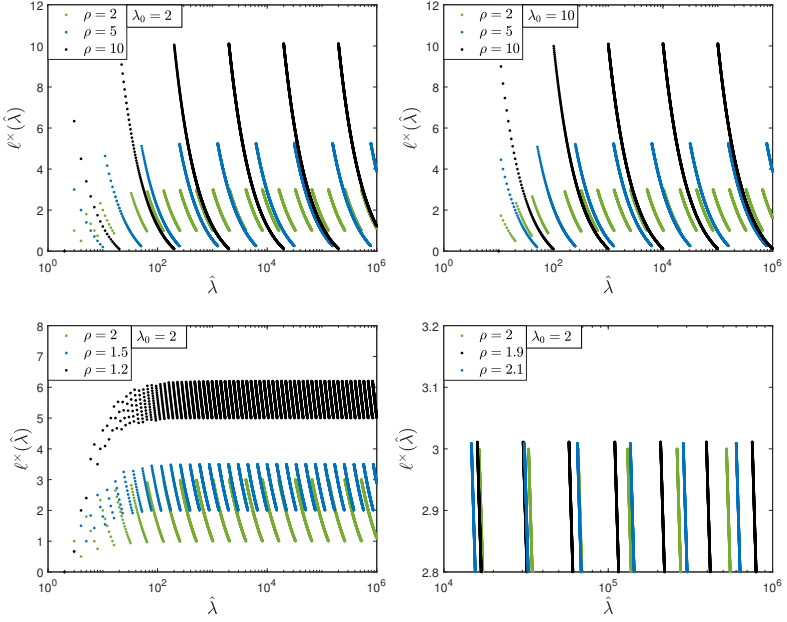


Figure 4: Relative loss function (15) for the  $\mathbb{R}^\times$ -RS determined numerically using Alg. 1.

The upper left and upper right plots illustrate the relative loss function for identical values of  $\rho$ , but for different start values  $\lambda_0$ . The left plot uses  $\lambda_0 = 2$  and the right plot  $\lambda_0 = 10$ , respectively. The asymptotic upper bounds are identical in both plots, indicating that they are independent of  $\lambda_0$ . As illustrated in the upper plots of Fig. 4, the asymptotic upper bound of the relative loss function is larger for larger values of  $\rho$ . Conversely, the lower left plot shows that the asymptotic upper bound is smaller for larger  $\rho$ . In both cases, the upper bound attains its minimal value when  $\rho = 2$ . The lower right plot shows the relative loss where  $\rho$  has values closed at 2. Again, the loss is minimal when  $\rho$  is equal to 2. This leads to the hypothesis that  $\hat{\rho} = 2$  is the optimal choice of  $\rho$ . The asymptotic upper bound of the relative loss function in this case is 3. Based on these observations, the following theorem can be formulated:

**Theorem:** The asymptotic upper bound  $\overline{\ell_{\text{up}}^{\times}}(\rho)$  (16) of the relative loss for the  $\mathbb{R}^{\times}$ -RS is minimal for  $\hat{\rho} = 2$ . Furthermore, it holds

$$\overline{\ell_{\text{up}}^{\times}}(\hat{\rho}) = 3. \quad (17)$$

The proof will be given in a forthcoming paper.

In addition to the asymptotic upper bound, one may also define an asymptotic lower bound. As illustrated in the upper plots of Fig. 3, the lower bound of the relative loss for the  $\mathbb{R}^+$ -RS and the  $\mathbb{R}^{\#}$ -RS also tend to infinity. In the case of the  $\mathbb{R}^{\times}$ -RS, the asymptotic lower bound is observed to be smaller, the larger  $\rho$  is. This is visible in the left plots of Fig. 4 and indicates that there is no optimal choice of  $\rho$  w.r.t. the asymptotic lower bound.

## 5 Conclusion and Future Work

This paper investigated restart strategies (RS) applied to algorithms whose success depends on an algorithmic parameter  $\lambda$ . The optimal choice of this algorithmic parameter for RS is a question that has not been investigated so far. To this end, the loss function was introduced, which measures the number of function evaluations of an RS compared to the number of function evaluations of the optimal strategy. Even more important is the relative loss, i.e., the loss relative to the optimal  $\lambda$ . To estimate and compare the relative loss for different restart strategies, the set of all restart strategies was divided into parameter-dependent subsets called strategy types. The goal was to find strategy types whose relative loss functions are upper bounded. In this case, it is possible to minimize the upper bound according to the parameter of the strategy type.

To gain further insight, the relative loss function was determined experimentally for different strategy types. In the case of the  $\mathbb{R}^+$ -RS, where the same amount  $v$  is added after each restart, the relative loss function is unbounded. The same is true for the  $\mathbb{R}^{\#}$ , where the parameters are determined according to a power law. For the multiplicative strategy type  $\mathbb{R}^{\times}$ -RS, where the algorithmic parameter is multiplied by an increasing constant  $\rho$  after each restart, it was demonstrated that the relative loss function is bounded. The same is true for the  $\mathbb{R}^*$ -RS,



which is also a multiplicative strategy type. In contrast to the  $\mathbb{R}^\times$ -RS the value of  $\lambda_k$  for this strategy type is derived from the previous rounded values. In investigating this strategy type, it was shown that there exists an optimal choice of  $\rho$ , which minimizes the asymptotic upper bound of the relative loss function. The optimal choice for the increasing constant is  $\hat{\rho} = 2$ . When  $\rho = \hat{\rho}$  is chosen, the maximum relative loss w.r.t.  $\hat{\lambda}$  is 3. These results are independent of the start value  $\lambda_0$ . It was also demonstrated that there is no  $\rho$  that minimizes the asymptotic lower bound of the relative loss function. All of these results have been demonstrated experimentally. The analytical details and proofs will be presented in a forthcoming paper.

## Appendix

**CSA-Evolution Strategy:** The experiments described in Section 3 are conducted using the  $(\mu/\mu_I, \lambda)$ -CSA-ES with  $c = 1/\sqrt{N}$ . This Evolution Strategy (ES) is outlined in detail in Alg. 2. In each generation, the population comprises  $\lambda$  offspring individuals. The offspring individuals are generated with mutation strength  $\sigma$  by the use of isotropic Gaussian mutations (Lines 4 and 5). In the selection step, the fitness value of each offspring individual is calculated (Line 6). Subsequently, the individuals are sorted in accordance with their fitness values (Line 7). Only the  $\mu$  individuals with the best fitness survive and serve as the parents for the subsequent generation. The ratio between parents and offspring is denoted by  $\vartheta := \mu/\lambda$  and is referred to as the truncation ratio. The initial state of the next generation is the result of recombination, i.e., the centroid of all parents (Line 9). The subscript  $m; \lambda$  denotes the  $m$ th best individual from  $\lambda$  offspring. The mutation strength  $\sigma$  is adapted for each new generation through cumulative step-size adaptation (CSA) (Lines 10 - 12).  $c$  is called the cumulation time parameter.

**Test Function:** The Rastrigin function for an  $N$ -dimensional search vector  $\mathbf{y} = (y_1, \dots, y_N)$  is given by

$$F(\mathbf{y}) = \sum_{i=1}^N [y_i^2 + A(1 - \cos(\alpha y_i))] \quad (18)$$

---

**Algorithm 2.: The  $(\mu/\mu_l, \lambda)$ -CSA Evolution Strategy**

---

```
1: Initialize  $(\mathbf{y}^{(0)}, \boldsymbol{\sigma}^{(0)}, \boldsymbol{\sigma}_{\text{stop}}, \mathbf{s} = \mathbf{1}, g = 0)$ 
2: repeat
3:   for  $l = 1$  to  $\lambda$  do
4:      $\tilde{\mathbf{z}}_l = (\mathcal{N}(0, 1), \dots, \mathcal{N}(0, 1))$  ▷ search direction
5:      $\tilde{\mathbf{y}}_l = \mathbf{y}^{(g)} + \boldsymbol{\sigma}^{(g)} \tilde{\mathbf{z}}_l$  ▷ mutate  $\mathbf{y}$ 
6:      $\tilde{F}_l = F(\tilde{\mathbf{y}}_l)$  ▷ evaluate offspring
7:   Sort Individuals  $\tilde{\mathbf{y}}$  Ascendingly w.r.t. Fitness  $\tilde{F}$ 
8:    $g = g + 1$ 
9:    $\mathbf{y}^{(g)} = \frac{1}{\mu} \sum_{m=1}^{\mu} \tilde{\mathbf{y}}_{m;\lambda}$  ▷ recombine the  $\mu$  best  $\tilde{\mathbf{y}}$ 
10:   $\mathbf{z}^{(g)} = \frac{1}{\mu} \sum_{m=1}^{\mu} \tilde{\mathbf{z}}_{m;\lambda}$  ▷ recombine the  $\mu$  best  $\tilde{\mathbf{z}}$ 
11:   $\mathbf{s} = (1 - c)\mathbf{s} + \sqrt{\mu c (2 - c)} \mathbf{z}^{(g)}$  ▷ update  $\mathbf{s}$ -path
12:   $\boldsymbol{\sigma}^{(g)} = \boldsymbol{\sigma}^{(g-1)} \exp\left(\frac{\|\mathbf{s}\|^2 - N}{2DN}\right)$  ▷ update  $\boldsymbol{\sigma}$ 
13: until  $\boldsymbol{\sigma}^{(g)} < \boldsymbol{\sigma}_{\text{stop}}$ 
```

---

where the parameter  $A$  denotes the oscillation amplitude and  $\alpha$  denotes the frequency. The experiments in Section 3 were executed with  $A = 1$  and  $\alpha = 2\pi$ . The global optimizer is located at  $\hat{\mathbf{y}} = \mathbf{0}$ . The Rastrigin function is a highly multimodal function. The number of local minima increases exponentially with the search space dimensionality  $N$ . In the case of the experiments the number of local minima is  $7^{30} - 1$  for  $N = 30$  and  $7^{100} - 1$  for  $N = 100$ .

## References

- [1] M. Luby, A. Sinclair, and D. Zuckerman: “Optimal speedup of Las Vegas algorithms”. In: *[1993] The 2nd Israel Symposium on Theory and Computing Systems.*, 1993, p. 128–133. doi: 10.1109/TEVC.2024.3419931.
- [2] J.-H., Lorenz: “Restart Strategies in a Continuous Setting”. In: *Theory of Computing Systems.*, Vol. 65. 2021, p. 1143–1164. doi: 10.1007/s00224-021-10041-0. url: <https://doi.org/10.1007/s00224-021-10041-0>
- [3] L. Schönerberger, and H.-G. Beyer: “On a Population Sizing Model for Evolution Strategies in Multimodal Landscapes”. In: *IEEE Transactions on Evolutionary Computation.* 2024. doi: 10.1109/TEVC.2024.3419931.

- [4] A. Omeradzic, and H.-G. Beyer: “Self-Adaptation of Multi-Recombinant Evolution Strategies on the Highly Multimodal Rastrigin Function”. In: *IEEE Transactions on Evolutionary Computation*. 2024. doi: 10.1109/TEVC.2024.3400857.
- [5] A. Auger, and N. Hansen: “A Restart CMA Evolution Strategy with Increasing Population Size”. In: *IEEE Transactions on Evolutionary Computation*. Vol. 2. 2005, p. 1769–1776 doi: 10.1109/CEC.2005.1554902.
- [6] T. Jansen: “On the Analysis of Dynamic Restart Strategies for Evolutionary Algorithms”. In: *Parallel Problem Solving from Nature — PPSN VII*. Springer Berlin Heidelberg, 2002, p. 33–43.
- [7] N. Hansen: “*Verallgemeinerte individuelle Schrittweitenregelung in der Evolutionsstrategie*”. Doctoral thesis. Technical University of Berlin, 1998.