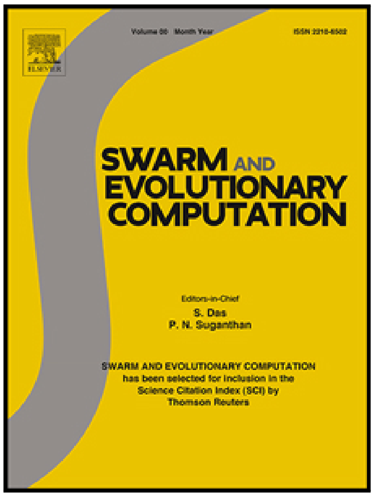
Journal Pre-proof



Error Analysis of Elitist Randomized Search Heuristics

Cong Wang, Yu Chen, Jun He, Chengwang Xie

PII: S2210-6502(21)00036-5
DOI: https://doi.org/10.1016/j.swevo.2021.100875
Reference: SWEVO 100875

To appear in: *Swarm and Evolutionary Computation*

Received date: 28 February 2020
Revised date: 17 February 2021
Accepted date: 23 March 2021

Please cite this article as: Cong Wang, Yu Chen, Jun He, Chengwang Xie, Error Analysis of Elitist Randomized Search Heuristics, *Swarm and Evolutionary Computation* (2021), doi: https://doi.org/10.1016/j.swevo.2021.100875

Error Analysis of Elitist Randomized Search Heuristics[☆,☆☆]

Cong Wang[a], Yu Chen[a,*], Jun He[b], Chengwang Xie[c,*]

[a]*School of Science, Wuhan University of Technology, Wuhan 430070, China*
[b]*School of Science and Technology, Nottingham Trent University, Nottingham NG11 8NS, UK*
[c]*School of Computer and Information Engineering, Nanning Normal University, Nanning 530299, China*

Abstract

For complicated problems that cannot be solved in polynomial first hitting time (FHT)/running time(RT), a remedy is to perform approximate FHT/TH analysis for given approximation ratio. However, approximate FHT/RT analysis of randomized search heuristics (RSHs) is not flexible enough because polynomial FHT/RT is not always available for any given approximation ratio. In this paper, the error analysis, which focuses on estimation of the expected approximation error of RSHs, is proposed to accommodate the requirement of flexible analysis. By diagonalizing one-step transition matrix of the Markov chain model, a tight estimation of the expected approximation error can be obtained via estimation of the multi-step transition matrix. For both uni- and multi-modal problems, error analysis leads to precise estimations of approximation error instead of asymptotic results on fitness values, which demonstrates its competitiveness to FHT/RT analysis as well as the fixed budget analysis.

Keywords: Expected Approximation Error, Fixed-Budget Analysis, Running Time Analysis, Random Local Search, (1+1)EA, Knapsack Problem.

1. Introduction

Randomized search heuristics (RSHs), including evolutionary algorithms (EAs), particle swarm optimization, ant colony optimization, etc., could be employed solving a wide variety of optimization problems. However, their performances are significantly influenced by mathematical characteristics of the investigated problems. Thus, one would compare performances of RSHs on various fitness landscapes prior to design of individualized strategies for complicated optimization problems [1, 2, 3]. Considering that numerical simulation is to some extent approximation of the underlying iteration mechanism, one could design individualized strategies based on the results of theoretical study.

Theoretical analysis of RSHs was usually focused on estimation of the expected first hitting time (FHT) or the expected running time (RT), which quantify the needed evaluation budget to hit the global optimal solutions. A variety of theoretical routines were proposed for estimation of FHT/RT [4, 5, 6, 7, 8, 9], and massive theoretical results have been reported in the past years [10, 11, 12, 13, 14, 15, 16]. Although popular in theoretical study, FHT/RT analysis does not work well when RSHs are not anticipated to locate the global optimal solutions in polynomial FHT/RHT. For this case, a remedy is to investigate the approximate

☆This paper is an extension of our conference paper: C. Wang, Y. Chen, J. He and C. Xie, Estimating approximation errors of elitist evolutionary algorithms, In *International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2019)*, Springer, 2019, pp. 325-340.
☆☆This work was partly supported by the Fundamental Research Funds for the Central Universities (WUT: 2020IB006), the National Nature Science Foundation of China under Grant 61763010, the Guangxi "BAGUI Scholar" Program, and the Science and Technology Major Project of Guangxi under Grant AA18118047.
*Corresponding authors
Email addresses: ychen@whut.edu.cn (Yu Chen), chengwangxie@nnnu.edu.cn (Chengwang Xie)

FHT/RT by taking an approximation set as the hitting destination of consecutive iterations [17, 18, 19, 20, 21, 22, 23, 24, 25].

However, approximate FHT/RT analysis depends on preset values of the approximation ratio, which are sometimes unavailable for achievement of polynomial FHT/RT. For this case, approximate FHT/RT analysis are not flexible enough to accommodate the requirement of theoretical study. Inspired by the fact that numerical performance of EAs is usually evaluated by qualities of solutions obtained with a given budget, Jansen and Zarges proposed to estimate objective values by fixed-budget analysis [26]. Following this theoretical routine, Jansen and Zarges performed a theoretical evaluation of immune-inspired hypermutations [27], and Nallaperuma *et al.* analyzed performances of RSH on the traveling salesperson (TSP) problem [28]. For given iteration budget t, fixed budget analysis generated bound estimations of fitness, which was not general and sometimes invalid for a large t. Moreover, it was performed by analysis tricks depending on properties of the investigated problems. Thus, general analysis frameworks were not easy to be obtained and sometimes only asymptotic results could be obtained.

Similar to the analysis routine of deterministic iteration algorithms, expected approximation error of RSHs can be estimated by evaluating the convergence rate (CR). Due to the stochastic iteration mechanism of RSH, CR was defined as $r^{[t]} = e^{[t]}/e^{[t-1]}$, where $e^{[t]}$ is the expected approximation error at generation t. By restricting the convergence rate under the condition $r^{[t]} \leq \lambda < 1$, Rudolph [29] proved that the sequence $\{e^{[t]}; t = 0, 1, \cdots\}$ converges in mean geometrically fast to 0. However, numerical simulation of $r^{[t]}$ is unstable. Thus, He and Lin [30] propose to investigate the average convergence rate (ACR) of binary-coded RSHs, defined as $R^{[t]} = 1 - \left(e^{[t]}/e^{[0]}\right)^{1/t}$. They estimated the lower bound of $R^{[t]}$ and proved that $R^{[t]}$ converges to an eigenvalue of the transition matrix if the initial population of EA is randomly initialized. Recently, Chen and He [31, 32] performed an ACR analysis for continuous RSHs, which demonstrated a significantly different performance of RSHs for continuous optimization problems.

Starting from $r^{[t]}$ or $R^{[t]}$, it is straightforward to get an exact expression of the approximation error by

$$e^{[t]} = e^{[0]} \prod_{k=1}^{t} r^{[k]} \text{ or } e^{[t]} = e^{[0]}(1 - R^{[t]})^t.$$

He *et al.* [33, 34] performed an unlimited budget analysis to get expected approximation error by estimating one-step convergence rate $r^{[t]}$ for any t. However, tight evaluation of the convergence rate dependent on t is a challenging task, while a general estimation of $r^{[t]}$ could lead to a very loose estimation of the expected approximation error.

An alternative routine to compute the expected approximation error is to estimate the probability distribution via multiplication of transition matrices. He [35] proved if the transition matrix associated with an EA is an upper triangular matrix with distinct diagonal entries, the relative error $e^{[t]}$ could be expressed as

$$e^{[t]} = \sum_{k=1}^{L} c_k \lambda_k^{t-1}, \quad \forall\ t \geq 1, \tag{1}$$

where λ_k are eigenvalues of the transition matrix and c_k are coefficients. In accordance with this idea, He *et al.* [36] proposed to compute c_k and λ_k^{t-1} by estimating t-th power of the transition matrix, and presented several mathematical routines depending on the properties of transition matrices.

As suggested by He *et al.* [35, 36], we investigated performances of random local search (RLS) for the case that the status transition matrices can be computationally diagonalized, and estimated the expected approximation error for arbitrary iteration budget [37]. However, when the bitwise mutation is employed in the (1+1) evolutionary algorithm ((1+1)EA), the transition matrix is a full upper triangular transition matrix, the t-th power of which is theoretically feasible but computationally unavailable. In this study, we try to address this issue by construction of auxiliary Markov models. For a searching process characterized by a full upper triangular matrix (named as an elitist search), we construct an auxiliary bi-diagonal search process that is modelled by a bi-diagonal transition matrix. While the bi-diagonal search converges slowly than the elitist one, expected approximation error of the bi-diagonal search is an upper bound of that of the elitist one. In this way, we get a general framework of error analysis that leads to an exact estimation of approximation error. Rest of this paper is organized as follows. In Section 2, we present the motivation to perform error analysis, and some preliminaries for error analysis are given in Section 3. Then, we get several theorems on the approximation error of RSHs in Section 4, and case studies are performed in Section 5 to demonstrate feasibility and competitiveness of error analysis. Moreover, an instance of the knapsack problem is further investigated in Section 6 to verify applicability of the error analysis on constrained optimization problems. Finally, Section 7 concludes this paper.

2. Motivation to Perform Error Analysis

It is well-known that an algorithm could be efficient when it addresses a problem in polynomial FHT/RT. In view of this idea, a variety of theoretical analyses revealed that RHSs can address some instances efficiently [10, 11, 12, 13, 14, 15, 16]. Meanwhile, it is also demonstrated that some problems cannot be addressed in polynomial FHT/RT. To take the second best, it is expected that an RSH can get approximate optima in polynomial FHT/RT. In this way, approximate FHT/RT analyses are performed for given approximation ratios [17, 18, 19, 20, 21, 22, 23, 24, 25]. However, approximate FHT/RT analysis is not flexible enough to address some cases.

- When short of knowledge about mathematical properties of the investigated problems, for what approximation ratio can an RSH address the investigated problem in polynomial FHT/RT?
- If an approximation ratio cannot be achieved by an RSH in polynomial FHT/RT, what approximate ratio can it attain?
- Furthermore, what approximation ratio/error can be achieved for any given iteration budget?

In numerical experiments, performances of RSHs are usually evaluated by expected fitness values (or approximation errors, if the global optima are known). Inspired by this motivation, Jansen and Zarges proposed to estimate expected objective values by fixed-budget analysis [26]. However, their work did not provide theoretical results for any iteration budget, and the analyzing methods were not general but problem-dependent. In this study, we try to propose a general framework of error analysis to get the expected approximation error for any iteration budget. It is anticipated that results of error analysis can further narrow the gap between theoretical study and algorithm implement.

3. Preliminaries

In this paper, we consider the maximization problem

$$\max \quad f(\mathbf{x}), \quad \mathbf{x} = (x_1, \dots, x_n) \in \{0,1\}^n. \tag{2}$$

Denote its optimal solution and the corresponding objective value as $\mathbf{x}^*$ and f^*, respectively. Then, quality of a solution $\mathbf{x}$ can be evaluated by its approximation error $e(\mathbf{x}) = |f(\mathbf{x}) - f^*|$. In this study, an elitist RSH described in Algorithm 1 is investigated. When the one-bit mutation is employed, it is called a *random local search (RLS)*; if the bitwise mutation is used, it is named as a *(1+1)EA*.

Algorithm 1 Elitist Randomized Search Heuristics

```
counter t = 0;
randomly initialize a solution x_0;
while the stopping criterion is not satisfied do
    generate a new candidate solution y_t from x_t by mutation;
    set individual x_{t+1} = y_t if f(y_t) > f(x_t); otherwise, let x_{t+1} = x_t;
    t = t + 1;
end while
```

The population sequence $\{\mathbf{x}_t, t = 0, 1, \dots\}$ of RLS/(1+1)EA can be modelled as a *Homogeneous Markov Chain (HMC)*. We can classify the solution set into $L+1$ mutually disjoint subset $\mathcal{X}_0, \mathcal{X}_1, \dots, \mathcal{X}_L$, where solutions in $\mathcal{X}_i$ have identical approximation error e_i satisfying

$$0 = e_0 \le e_1 \le \cdots \le e_L. \tag{3}$$

If $\mathbf{x} \in \mathcal{X}_i$, it is *at the status i*. Status 0, consisting of globally optimal solutions, is called the *optimal status*, and other statuses are *non-optimal statuses*. Then, $\{\mathbf{x}_t, t = 0, 1, \dots\}$ is a discrete HMC with $L+1$ available statuses, and the transition probability matrix is $\tilde{\mathbf{R}} = (r_{i,j})_{(L+1)\times(L+1)}$, where

$$r_{i,j} = \Pr\{\mathbf{x}_{t+1} \in \mathcal{X}_i | \mathbf{x}_t \in \mathcal{X}_j\}, \quad i, j = 0, \dots, L.$$

Denote the error vector of available statuses as $\tilde{\mathbf{e}} = (e_0, \dots, e_L)'$ [1]. Initialization of solutions $\mathbf{x_0}$ generates an initial status distribution $\tilde{\mathbf{p}}^{[0]} = (p_0^{[0]}, p_1^{[0]}, \dots, p_L^{[0]})'$. After t generations, we get the status distribution

$$\tilde{\mathbf{p}}^{[t]} = (p_0^{[t]}, p_1^{[t]}, \dots, p_L^{[t]})' = \tilde{\mathbf{R}}^t \tilde{\mathbf{p}}^{[0]},$$

and the expected approximation error can be computed by

$$e^{[t]} = \tilde{\mathbf{e}}' \tilde{\mathbf{R}}^t \tilde{\mathbf{p}}^{[0]}. \tag{4}$$

When the elitist selection is employed, the transition matrix $\tilde{\mathbf{R}}$ is upper triangular, and it can be partitioned as

$$\tilde{\mathbf{R}} = \begin{pmatrix} 1 & \mathbf{r}_0 \\ \mathbf{0} & \mathbf{R} \end{pmatrix}, \tag{5}$$

where $\mathbf{r}_0 = (r_{0,1}, r_{0,2}, \dots, r_{0,L})$, $\mathbf{0} = (0, \dots, 0)'$,

$$\mathbf{R} = \begin{pmatrix} r_{1,1} & \cdots & r_{1,L} \\ & \ddots & \vdots \\ & & r_{L,L} \end{pmatrix}. \tag{6}$$

The following lemma leads to a simplified formula of the expected approximation error.

Lemma 1 *Let $\tilde{\mathbf{e}} = (e_0, e_1, \dots, e_L)$ and $\tilde{\mathbf{v}} = (v_0, v_1, \dots, v_L)'$ be non-negative vectors. If $e_0 = 0$, it holds that*

$$\tilde{\mathbf{e}}' \tilde{\mathbf{R}}^t \tilde{\mathbf{v}} = \mathbf{e}' \mathbf{R}^t \mathbf{v}, \quad , \forall\, t \in \mathbb{Z}^+,$$

where $\mathbf{e} = (e_1, \dots, e_L)$, $\mathbf{v} = (v_1, \dots, v_L)'$, $\tilde{\mathbf{R}}$ and $\mathbf{R}$ confirmed by (5) and (6), respectively.

Proof The proof is trivial since

$$\tilde{\mathbf{e}}' \tilde{\mathbf{R}}^t \tilde{\mathbf{v}} = (0, \mathbf{e}') \begin{pmatrix} 1 & \mathbf{r}_0 \\ \mathbf{0} & \mathbf{R} \end{pmatrix}^t (v_0, \mathbf{v}')' = (0, \mathbf{e}') \begin{pmatrix} 1 & \star \\ \mathbf{0} & \mathbf{R}^t \end{pmatrix} (v_0, \mathbf{v}')' = \mathbf{e}' \mathbf{R}^t \mathbf{v}, \quad , \forall\, t \in \mathbb{Z}^+.$$

□

Let $\mathbf{R}$ be the transition submatrix of non-optimal statuses, and denote the approximation error of non-optimal statuses as

$$\mathbf{e} = (e_1, \dots, e_L)',$$

where $0 < e_i \le e_{i+1}$, $\forall\, i = 1, \dots, L-1$. By Lemma 1 one can get the expected approximation error by substituting $\mathbf{r}$ with the initial distribution of non-optimal statuses

$$\mathbf{p}^{[0]} = (p_1^{[0]}, \dots, p_L^{[0]})'.$$

That is,

$$e^{[t]} = \mathbf{e}' \mathbf{R}^t \mathbf{p}^{[0]}. \tag{7}$$

[1] To avoid confusion with the matrix power, the transpose operation is denoted in this paper by $'$ instead of ^{T}.

4. General Framework of Error Analysis

4.1. Theoretical Routine of Error Analysis

Formula (7) reveal that the key step for evaluation of the expected approximation error is computation of $\mathbf{R}^t$. If $\mathbf{R}$ is diagonal, we know $\mathbf{R}^t$ is a diagonal matrix whose diagonal items are the t-th power of diagonal items of $\mathbf{R}$. If $\mathbf{R}$ is not diagonal, we can get the analytic expression of $\mathbf{R}^t$ by similarity diagonalization.

Definition 1 *A matrix* $\mathbf{R}$ *is* ***diagonalizable****, if there exists an invertible matrix* $\mathbf{P}$ *and a diagonal matrix* $\mathbf{\Lambda} = diag\{\lambda_1, \dots, \lambda_n\}$ *such that*

$$\mathbf{\Lambda} = \mathbf{P}^{-1}\mathbf{R}\mathbf{P}. \tag{8}$$

The following proposition provides a sufficient condition for a matrix to be diagonalizable.

Lemma 2 *[38] An $L \times L$ matrix with L distinct eigenvalues is diagonalizable.*

Because eigenvalues of a upper triangular matrix are its diagonal elements, $\mathbf{R}$ can be diagonalized if it has L distinct diagonal elements [38]. Then, formula (8) implies

$$\mathbf{R} = \mathbf{P}\mathbf{\Lambda}\mathbf{P}^{-1},$$

and it holds

$$\mathbf{R}^t = \left(\mathbf{P}\Lambda\mathbf{P}^{-1}\right)\cdots\left(\mathbf{P}\Lambda\mathbf{P}^{-1}\right) = \mathbf{P}\Lambda^t\mathbf{P}^{-1}.$$

Estimation of expected approximation error is hence reduced to derivation of the transformation matrix $\mathbf{P}$. To address this issue, the generated searching processes are classified into three categories via the transition submatrix of non-optimal statuses.

1. **Diagonal Search**: If the transition submatrix is a diagonal matrix

$$\mathbf{R}_D = diag\{r_{1,1}, r_{2,2}, \dots, r_{L,L}\}, \tag{9}$$

   an RSH generates a *diagonal search*.
2. **Bi-Diagonal Search**: If the transition submatrix is a bi-diagonal matrix

$$\mathbf{R}_{BD} = \begin{pmatrix} r_{1,1} & r_{1,2} & & & \\ & r_{2,2} & r_{2,3} & & \\ & & \ddots & \ddots & \\ & & & r_{L-1,L-1} & r_{L-1,l} \\ & & & & r_{L,L} \end{pmatrix}, \tag{10}$$

   an RSH generates a *bi-diagonal search.*
3. **Elisit Search**: If the transition submatrix is an upper triangular matrix

$$\mathbf{R}_E = \begin{pmatrix} r_{1,1} & r_{1,2} & r_{1,3} & \cdots & r_{1,L} \\ & r_{2,2} & r_{2,3} & \cdots & r_{2,L} \\ & & \ddots & \ddots & \vdots \\ & & & r_{L-1,L-1} & r_{L-1,L} \\ & & & & r_{L,L} \end{pmatrix}, \tag{11}$$

   an RSH generates an *elitist search.*

Definition of status implies that an elitist RSH generates an elitist search. While an RSH generates a diagonal search, computation of the t-th power of the transition submatrix is a trivial task because $\mathbf{R}_D$ is diagonal; For the submatrix $\mathbf{R}_{BD}$ of a bi-diagonal search, we can also derive the analytic forms of the transformation matrix $\mathbf{P}$ and its inverse $\mathbf{P}^{-1}$, and then get the analytic form of the t-th power. However, it is difficult to get the analytic expression of the t-th power for a general upper-triangular matrix. For this case, we would construct an auxiliary bi-diagonal search that converges more slowly than the elitist one, and consequently, an upper bound is available for the expected approximation error of an elitist search.

4.2. Expected Approximation Error of a Diagonal Search

The expected approximation error $e_D^{[t]}$ of a diagonal search is confirmed by the following theorem.

Theorem 1 *For a diagonal search generated by an RSH, it holds*

$$e_D^{[t]} = \sum_{i=1}^{L} r_{i,i}^t e_i p_i^{[0]}.$$

Proof Note that

$$\mathbf{R}_D^t = \left(diag\{r_{1,1}, r_{2,2}, \ldots, r_{L,L}\}\right)^t = diag\{r_{1,1}^t, r_{2,2}^t, \ldots, r_{L,L}^t\}.$$

Then, Lemma 1 implies that

$$e_D^{[t]} = \mathbf{e}'\mathbf{R_D}^t\mathbf{p}^{[0]} = \mathbf{e}'\left(diag\{r_{1,1}^t, r_{2,2}^t, \ldots, r_{L,L}^t\}\right)\mathbf{p}^{[0]} = \sum_{i=1}^{L} r_{i,i}^t e_i p_i^{[0]}.$$

□

4.3. Expected Approximation Error of a Bi-Diagonal Search

Expected approximation error of a bi-diagonal search can be computed via diagonalization of the transition submatrix of non-optimal statuses. If $\mathbf{R}$ is an upper triangular matrix with L distinct eigenvalues, it can be diagonalized as follows.

Lemma 3 *[38] If an $L \times L$ matrix $\mathbf{A}$ has L distinct eigenvalues λ_1, λ_2,...,λ_L, it can be diagonalized as*

$$\mathbf{\Lambda} = \mathbf{P}^{-1}\mathbf{A}\mathbf{P}. \tag{12}$$

Here, $\mathbf{\Lambda} = diag\{\lambda_1, \ldots, \lambda_L\}$, $\mathbf{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_L)$. $\mathbf{p}_i$ is the corresponding eigenvector of λ_i satisfying

$$\mathbf{A}\mathbf{p}_i = \lambda_i \mathbf{p}_i, \quad i = 1, \ldots, L.$$

Recall that equation (12) is equivalent to $\mathbf{A} = \mathbf{P}\mathbf{\Lambda}\mathbf{P}^{-1}$. Then,

$$\mathbf{A}^t = \left(\mathbf{P}\mathbf{\Lambda}\mathbf{P}^{-1}\right)^t = \underbrace{\left(\mathbf{P}\mathbf{\Lambda}\mathbf{P}^{-1}\right)\cdots\left(\mathbf{P}\mathbf{\Lambda}\mathbf{P}^{-1}\right)}_{t} = \mathbf{P}\mathbf{\Lambda}^t\mathbf{P}^{-1}, \tag{13}$$

which can be confirmed by computing the transformation matrix $\mathbf{P}$ and its inverse $\mathbf{P}^{-1}$. Consequently, we get the following result on the analytic expression of $\mathbf{R}_{BD}^t$.

Lemma 4 *If the bi-diagonal matrix $\mathbf{R}_{BD}$ has L distinct diagonal elements, it holds*

$$\mathbf{R}_{BD}^t = \sum_{j=1}^{L} \lambda_j^t \mathbf{p}_j \mathbf{q}_j',$$

where

$$\mathbf{p}_j = \left(\prod_{k=1}^{j-1} \frac{r_{k,k+1}}{r_{j,j} - r_{k,k}}, \prod_{k=2}^{j-1} \frac{r_{k,k+1}}{r_{j,j} - r_{k,k}}, \ldots, \frac{r_{j-1,j}}{r_{j,j} - r_{j-1,j-1}}, 1, 0, \ldots, 0\right)', \tag{14}$$

$$\mathbf{q}_j' = \left(0, \ldots, 0, 1, \frac{r_{j,j+1}}{r_{j,j} - r_{j+1,j+1}}, \prod_{k=j+1}^{j+2} \frac{r_{k-1,k}}{r_{j,j} - r_{k,k}}, \ldots, \prod_{k=j+1}^{L} \frac{r_{k-1,k}}{r_{j,j} - r_{k,k}}\right), \tag{15}$$

$$j = 1 \ldots, L.$$

Proof If $\mathbf{R}_{BD}$ has L distinct diagonal elements, it has L distinct eigenvalues

$$\lambda_i = r_{i,i}, \quad i = 1, \ldots, L.$$

Then, Lemma 3 applies and $\mathbf{R}_{BD}$ can be diagonalized as

$$\mathbf{P}^{-1}\mathbf{R}_{BD}\mathbf{P} = \mathbf{\Lambda} = diag\{\lambda_1, \lambda_2, \ldots, \lambda_L\}, \tag{16}$$

where $\mathbf{P} = (\mathbf{p}_1, \ldots, \mathbf{p}_L)$,

$$\mathbf{R}_{BD}\mathbf{p}_j = \lambda_j \mathbf{p}_j,\ \mathbf{p}_j \neq \mathbf{0}, \quad j = 1, \ldots, L. \tag{17}$$

Denote $\mathbf{p}_j = (p_{1,j}, \ldots, p_{L,j})'$. Equation (17) indicates that

$$\begin{cases} r_{i,i}p_{i,j} + r_{i,i+1}p_{i+1,j} = r_{j,j}p_{i,j}, & i = 1, \ldots, L; \\ r_{L,L}p_{L,j} = r_{j,j}p_{L,j}. \end{cases} \quad j = 1, 2, \ldots, L.$$

Note that $r_{i,i} \neq r_{j,j}$ when $i \neq j$. Thus, for the eigenvalue λ_j we can obtain an corresponding eigenvector $\mathbf{p}_j = (p_{1,j}, \ldots, p_{L,j})'$ confirmed by

$$p_{i,j} = \begin{cases} 0, & \text{if } i > j; \\ 1, & \text{if } i = j; \\ \prod_{k=i}^{j-1} \dfrac{r_{k,k+1}}{r_{j,j} - r_{k,k}}, & \text{if } i < j; \end{cases} \quad j = 1, 2, \ldots, L.$$

That is,

$$\mathbf{p}_j = \left(\prod_{k=1}^{j-1} \frac{r_{k,k+1}}{r_{j,j} - r_{k,k}}, \prod_{k=2}^{j-1} \frac{r_{k,k+1}}{r_{j,j} - r_{k,k}}, \ldots, \frac{r_{j-1,j}}{r_{j,j} - r_{j-1,j-1}}, 1, 0, \ldots, 0 \right)', \quad j = 1 \ldots, L.$$

Denote $\mathbf{Q} = \mathbf{P}^{-1} = (\mathbf{q}_1, \ldots, \mathbf{q}_L)'$, where $\mathbf{q}_j = (q_{1,j}, \ldots, q_{L,j})'$. Because $\mathbf{Q}$ is the inverse matrix of $\mathbf{P}$, it is upper triangular, and its diagonal elements are inverses of the corresponding diagonal elements of $\mathbf{P}$. Thus,

$$q_{j,j} = 1, \quad j = 1, \ldots, L. \tag{18}$$

By equation (16) we know

$$(\mathbf{R}'_{BD}\mathbf{q}_1, \ldots, \mathbf{R}'_{BD}\mathbf{q}_L)' = \mathbf{Q}\mathbf{R}_{BD} = \mathbf{\Lambda}\mathbf{Q} = (\lambda_1\mathbf{q}_1, \ldots, \lambda_L\mathbf{q}_L)',$$

which menas

$$r_{j-1,j}q_{i,j-1} + r_{j,j}q_{i,j} = r_{i,i}q_{i,j}, \quad i = 1, \ldots, L,\ j = 1, \ldots, L.$$

Combining it with equation (18), we know

$$\begin{cases} q_{i,j} = 0; & i = 1, \ldots, j-1; \\ q_{i,j} = 1; & i = j; \\ q_{i,j} = \prod_{k=i+1}^{j} \frac{r_{k-1,k}}{r_{i,i} - r_{k,k}}; & i = j+1, \ldots, L, \end{cases} \quad j = 1, \ldots, L.$$

Then, rows of $\mathbf{Q}$ are depicted by

$$\mathbf{q}'_j = \left(0, \ldots, 0, 1, \frac{r_{j,j+1}}{r_{j,j} - r_{j+1,j+1}}, \prod_{k=j+1}^{j+2} \frac{r_{k-1,k}}{r_{j,j} - r_{k,k}}, \ldots, \prod_{k=j+1}^{n} \frac{r_{k-1,k}}{r_{j,j} - r_{k,k}} \right), j = 1 \ldots, L.$$

Finally, from (13) we conclude that

$$\mathbf{R}_{BD}^t = \mathbf{P}\mathbf{\Lambda}^t\mathbf{P}^{-1} = \mathbf{P}\mathbf{\Lambda}^t\mathbf{Q} = (\mathbf{p}_1, \ldots, \mathbf{p}_L) diag\{\lambda_1^t, \ldots, \lambda_L^t\}(\mathbf{q}_1, \ldots, \mathbf{q}_L)' = \sum_{j=1}^{L} \lambda_j^t \mathbf{p}_j \mathbf{q}_j'.$$

□

Now, we get the theorem on the expected approximation error of a bi-diagonal search.

Theorem 2 *The expected approximation error of a bi-diagonal search is*

$$e_{BD}^{[t]} = \sum_{j=1}^{L} \lambda_j^t \left(\mathbf{e}'\mathbf{p}_j\right) \left(\mathbf{q}_j'\mathbf{p}^{[0]}\right),$$

where $\mathbf{p}_j$ *and* $\mathbf{q}_j$ *are confirmed by (14) and (15), respectively.*

Proof Applying Lemmas 1 and 4, we know

$$e_{BD}^{[t]} = \mathbf{e}'\mathbf{R}_{BD}^t\mathbf{p}^{[0]} = \mathbf{e}'\left(\sum_{j=1}^{L} \lambda_j^t \mathbf{p}_j \mathbf{q}_j'\right)\mathbf{p}^{[0]} = \sum_{j=1}^{L} \lambda_j^t \left(\mathbf{e}'\mathbf{p}_j\right)\left(\mathbf{q}_j'\mathbf{p}^{[0]}\right).$$

□

4.4. Expected Approximation Error of an Elitist Search

Although feasible for the bi-diagonal matrix, it is difficult to get a general result for the exact expression of t-th power of upper triangular matrices. So, we would like to estimate not the precise expression but an upper bound of the approximation error of an elitist search. This idea could be realized by constructing an auxiliary bi-diagonal search that converges more slowly than the elitist one. Construction of the auxiliary search is based on the sufficient condition presented in the following lemma.

Lemma 5 *[36] Provided that transition matrices* $\tilde{\mathbf{R}} = (r_{i,j})_{(L+1)\times(L+1)}$ *and* $\tilde{\mathbf{S}} = (s_{i,j})_{(L+1)\times(L+1)}$ *are upper triangular. If*

$$s_{j,j} \geq r_{j,j}, \qquad \forall\ j, \tag{19}$$

$$\textstyle\sum_{l=0}^{i-1}(r_{l,j} - s_{l,j}) \geq 0, \qquad \forall\ i < j, \tag{20}$$

$$\textstyle\sum_{l=0}^{i}(s_{l,j-1} - s_{l,j}) \geq 0, \qquad \forall\ i < j-1, \tag{21}$$

it holds

$$\mathbf{T}\tilde{\mathbf{R}}^t \leq \mathbf{T}\tilde{\mathbf{S}}^t, \quad \forall\ t \in \mathbb{Z}^+,$$

where

$$T = \begin{pmatrix} 1 & \ldots & 1 \\ & \ddots & \vdots \\ & & 1 \end{pmatrix}.$$

Construct an auxiliary search characterized by $\tilde{\mathbf{S}}$, and partitioned it as

$$\tilde{\mathbf{S}} = \begin{pmatrix} s_{0,0} & \mathbf{s}_0 \\ \mathbf{0} & \mathbf{S} \end{pmatrix}, \tag{22}$$

where $\mathbf{s}_0 = (s_{0,1}, \ldots, x_{0,L})$,

$$\mathbf{S} = \begin{pmatrix} s_{1,1} & \ldots & s_{1,L} \\ & \ddots & \vdots \\ & & s_{L,L} \end{pmatrix}. \tag{23}$$

We get the following theorem that can be utilized to estimate upper bounds of expected approximation errors.

Theorem 3 *Denote the approximation error vector as* $\tilde{\mathbf{e}} = (e_0, \mathbf{e}')' = (e_0, e_1, \ldots, e_L)'$*, and let* $\tilde{\mathbf{v}} = (v_0, \mathbf{v}')' = (v_0, v_1, \ldots, v_L)'$*, where* $v_i > 0, i = 0, 1, \ldots, L$*. If transition matrices* $\tilde{\mathbf{R}}$ *and* $\tilde{\mathbf{S}}$ *satisfy conditions (19)-(21), it holds*

$$\mathbf{e}'\mathbf{R}^t\mathbf{v} \leq \mathbf{e}'\mathbf{S}^t\mathbf{v},$$

where $\mathbf{R}$ *and* $\mathbf{S}$ *are the transition submatrices confirmed by (6) and (23), respectively.*

Proof Since

$$\begin{aligned}
\tilde{\mathbf{e}}'\tilde{\mathbf{R}}^t\tilde{\mathbf{v}} - \tilde{\mathbf{e}}'\tilde{\mathbf{S}}^t\tilde{\mathbf{v}} &= \tilde{\mathbf{e}}'\left(\tilde{\mathbf{R}}^t - \tilde{\mathbf{S}}^t\right)\tilde{\mathbf{r}} \\
&= (e_0, e_1 - e_0, \ldots, e_L - e_{L-1})\mathbf{T}\left(\tilde{\mathbf{R}}^t - \tilde{\mathbf{S}}^t\right)\tilde{\mathbf{v}} \\
&\leq (e_0, e_1 - e_0, \ldots, e_L - e_{L-1})\left(\mathbf{T}\tilde{\mathbf{R}}^t - \mathbf{T}\tilde{\mathbf{S}}^t\right)\tilde{\mathbf{v}}.
\end{aligned}$$

Lemma 5 implies that $\mathbf{T}\tilde{\mathbf{R}}^t - \mathbf{T}\tilde{\mathbf{S}}^t$ is a negative matrix. Recall that the approximation vector satisfies

$$e_0 = 0; e_i \leq e_{i+1}, \quad i = 0, \ldots, L-1.$$

Then, we get a nonnegative vector

$$\Delta\tilde{\mathbf{e}} = (e_0, e_1 - e_0, \ldots, e_L - e_{L-1})',$$

and it holds

$$\tilde{\mathbf{e}}' = (\Delta\tilde{\mathbf{e}})'\mathbf{T}.$$

Consequently,

$$\tilde{\mathbf{e}}'\tilde{\mathbf{R}}^t\tilde{\mathbf{v}} = (\Delta\tilde{\mathbf{e}})'\mathbf{T}\tilde{\mathbf{R}}^t\tilde{\mathbf{v}} \leq (\Delta\tilde{\mathbf{e}})'\mathbf{T}\tilde{\mathbf{S}}^t\tilde{\mathbf{v}} = \tilde{\mathbf{e}}'\tilde{\mathbf{S}}^t\tilde{\mathbf{v}}.$$

Applying Lemma 1 we conclude that $\mathbf{e}'\mathbf{R}^t\mathbf{v} \leq \mathbf{e}'\mathbf{S}^t\mathbf{v}$. □

Substituting $\mathbf{v}$ in Theorem 3 by the initial distribution vector $\mathbf{q}^{[0]}$ of non-optimal statuses, we can get a general method for upper bound estimation of expected approximation errors. To make this estimation practical, one can construct an auxiliary search that is bi-diagonal.

5. Case Study

To validate feasibility of the error analysis routines proposed in Section 4, several case studies are performed for the RLS and the (1+1)EA.

Problem 1 ***(OneMax)***

$$\max f(\mathbf{x}) = \sum_{i=1}^{n} x_i, \quad \mathbf{x} = (x_1, \ldots, x_n) \in \{0,1\}^n.$$

Problem 2 ***(Peak)***

$$\max f(\mathbf{x}) = \prod_{i=1}^{n} x_i, \quad \mathbf{x} = (x_1, \ldots, x_n) \in \{0,1\}^n.$$

Problem 3 ***(Deceptive Problem)***

$$\max f(\mathbf{x}) = \begin{cases} \sum_{i=1}^{n} x_i, & \text{if } \sum_{i=1}^{n} x_i > n-1, \\ n - 1 - \sum_{i=1}^{n} x_i, & \text{otherwise.} \end{cases} \quad \mathbf{x} = (x_1, \ldots, x_n) \in \{0,1\}^n.$$

5.1. The OneMax Problem

Objective value of the OneMax problem is total amount of 1-bits in the bit-string $\mathbf{x}$, and the approximation error is number of 0-bits. Thus, the approximation error vector is

$$\tilde{\mathbf{e}} = (e_0, \mathbf{e}')' = (0, 1, 2, \ldots, n)'. \tag{24}$$

The solution space can be divided into $n+1$ statuses labeled by approximation errors. Random initialization generates the initial distribution denoted as

$$\tilde{\mathbf{p}}^{[0]} = (p_0^{[0]}, \mathbf{p}'^{[0]})' = (C_n^1/2^n, C_n^2/2^n, \ldots, C_n^n/2^n)'. \tag{25}$$

Then, the expected approximation error of RLS is given by the following theorem.

Theorem 4 *The expected approximation error of RLS for the OneMax problem is* $\frac{n}{2}\left(1-\frac{1}{n}\right)^t$.

Proof Combining the one-bit mutation with the elitist selection, RLS transfers from status j to $j-1$ with probability j/n, and keeps its status unchanged with probability $1 = j/n$. Thus, application of RLS on the OneMax problem generates a bi-diagonal search. The transition submatrix of non-optimal statuses is

$$\mathbf{R} = (r_{i,j})_{n\times n} = \begin{pmatrix} 1-1/n & 2/n & & & \\ & 1-2/n & 3/n & & \\ & & \ddots & \ddots & \\ & & & 1/n & 1 \\ & & & & 0 \end{pmatrix}. \tag{26}$$

Then, Theorem 2 implies that

$$e^{[t]} = \mathbf{e}'\mathbf{R}^t\mathbf{p}^{[0]} = \mathbf{e}'\left(\sum_{j=1}^{n}\lambda_j^t\mathbf{p}_j\mathbf{q}_j'\right)\mathbf{p}^{[0]} = \sum_{j=1}^{n}\lambda_j^t\left(\mathbf{e}'\mathbf{p}_j\right)\left(\mathbf{q}_j'\mathbf{p}^{[0]}\right), \tag{27}$$

where $\lambda_j = r_{j,j}$, $\mathbf{p}_j$ and $\mathbf{q}_j$ are defined by (14) and (15), respectively. By computation in Appendix A, we get the expression of $\mathbf{e}'\mathbf{p}_j$ and $\mathbf{q}_j'\mathbf{p}^{[0]}$ presented by (A.3) and (A.4). Substituting (24), (25), (A.3) and (A.4) to (27), we know

$$e^{[t]} = \sum_{j=1}^{n}\lambda_j^t\left(\mathbf{e}'\mathbf{p}_j\right)\left(\mathbf{q}_j'\mathbf{p}^{[0]}\right) = \lambda_1^t(\mathbf{e}'\mathbf{p}_1)(\mathbf{q}_1'\mathbf{p}^{[0]}) = \frac{n}{2}\left(1-\frac{1}{n}\right)^t.$$

□

The same results about performance of RLS on the OneMax problem have also been reported in [26, 33]. Jansen and Zarges get this results by the law of total probability [26], and He *et al.* get it with the help of the constant convergence rate [33].

Since the local search strategy of RLS generates a bi-diagonal search with a simple transition matrix, we can get the exact expression of expected approximation error. While the (1+1)EA is investigated, the analyzing process presented in [26] could be complicated. However, estimation of expected approximation error for the (1+1)EA can be obtained by application of the analysis framework proposed in Section 4.

Theorem 5 *The expected approximation error of (1+1)EA for the OneMax problem satisfies*

$$e^{[t]} \le \frac{n}{2}\left(1-\frac{1}{ne}\right)^t.$$

Proof Since the bitwise mutation can locate any solution with a positive probability, the (1+1)EA applied to OneMax generates an elitist search with $r_{i,j} > 0, \forall\, 0 \le i \le j \le L$. Thus, we estimate the upper bound of its approximation error by constructing an auxiliary bi-diagonal search.

The auxiliary bi-diagonal search is obtained by considering a special case that only one '0' in $\mathbf{x}$ is flip to '1'. While $\mathbf{x}$ is at the status j, it leads to decrease of approximation error and the status transition from j to $j-1$, and the probability is $\frac{j}{n}\left(1-\frac{1}{n}\right)^{n-1}$. Then, we get a lower bound of the transition probability $r_{j-1,j}$:

$$r_{j-1,j} \geq \frac{j}{n}\left(1-\frac{1}{n}\right)^{n-1}, \quad j=1,\dots,n.$$

For the elitist search generated by (1+1)EA, we get an auxiliary bi-diagonal search with probability transition matrix

$$\begin{aligned}
&\tilde{\mathbf{S}} = (s_{i,j})_{i,j=0,\dots,n} \\
&= \begin{pmatrix}
1 & \frac{1}{n}(1-\frac{1}{n})^{n-1} & & & & \\
 & 1-\frac{1}{n}(1-\frac{1}{n})^{n-1} & \frac{2}{n}(1-\frac{1}{n})^{n-1} & & & \\
 & & 1-\frac{2}{n}(1-\frac{1}{n})^{n-1} & \frac{3}{n}(1-\frac{1}{n})^{n-1} & & \\
 & & & \ddots & \ddots & \\
 & & & & 1-\frac{n-1}{n}(1-\frac{1}{n})^{n-1} & (1-\frac{1}{n})^{n-1} \\
 & & & & & 1-(1-\frac{1}{n})^{n-1}
\end{pmatrix}.
\end{aligned}$$

It is trivial to verify that $\tilde{\mathbf{R}}$ and $\tilde{\mathbf{S}}$ satisfy conditions (19)-(21), and the result of Theorem 3 holds. Combining Theorems 2 and 3 we know

$$e^{[t]} \leq \tilde{\mathbf{e}}'\tilde{\mathbf{S}}^t\tilde{\mathbf{p}}^{[0]} = \mathbf{e}'\mathbf{S}^t\mathbf{p}^{[0]} = \sum_{j=1}^{n} \lambda_j^t \left(\mathbf{e}'\mathbf{p}_j\right)\left(\mathbf{q}_j'\mathbf{p}^{[0]}\right), \tag{28}$$

where

$$\lambda_j = 1 - \frac{j}{n}\left(1-\frac{1}{n}\right)^{n-1}, \tag{29}$$

$$\mathbf{p}_j = \left(\prod_{k=1}^{j-1}\frac{s_{k,k+1}}{s_{j,j}-s_{k,k}}, \prod_{k=2}^{j-1}\frac{s_{k,k+1}}{s_{j,j}-s_{k,k}}, \dots, \frac{s_{j-1,j}}{s_{j,j}-s_{j-1,j-1}}, 1, 0, \dots, 0\right)', \tag{30}$$

$$\mathbf{q}_j' = \left(0, \dots, 0, 1, \frac{s_{j,j+1}}{s_{j,j}-s_{j+1,j+1}}, \prod_{k=j+1}^{j+2}\frac{s_{k-1,k}}{s_{j,j}-s_{k,k}}, \dots, \prod_{k=j+1}^{n}\frac{s_{k-1,k}}{s_{j,j}-s_{k,k}}\right), \tag{31}$$

$j = 1\dots,n$. Similar to computation of $\mathbf{p}_j$ and $\mathbf{q}_j'$ in Appendix A, we know that the values of $\mathbf{p}_j$ and $\mathbf{q}_j$ defined by (30) and (31) are confirmed by (A.1) and (A.2), respectively. Thus, the expression of $\mathbf{e}'\mathbf{p}_j$ and $\mathbf{q}_j'\mathbf{p}^{[0]}$ are presented by (A.3) and (A.4), too. Submitting (24), (25), (29), (A.3) and (A.4) to (28) we get the results that

$$e^{[t]} \leq \sum_{j=1}^{n} \lambda_j^t \left(\mathbf{e}'\mathbf{p}_j\right)\left(\mathbf{q}_j'\mathbf{p}^{[0]}\right) = \lambda_1^t(\mathbf{e}'\mathbf{p}_1)(\mathbf{q}_1'\mathbf{p}^{[0]}) = \frac{n}{2}\left(1-\frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1}\right)^t \leq \frac{n}{2}\left(1-\frac{1}{ne}\right)^t.$$

□

To demonstrate how tight the estimated upper bound is, we perform numerical comparison between the simulation results and the estimated bound, where the simulated approximation errors for the 10-, 20-,...,90-D OneMax problems are averaged for 1000 independent runs. As illustrated in Figure 1, the estimated upper bound is very tight for low-dimensional OneMax problems, and the difference between simulation results and the estimated upper bound increases slowly with increase of the problem dimension.

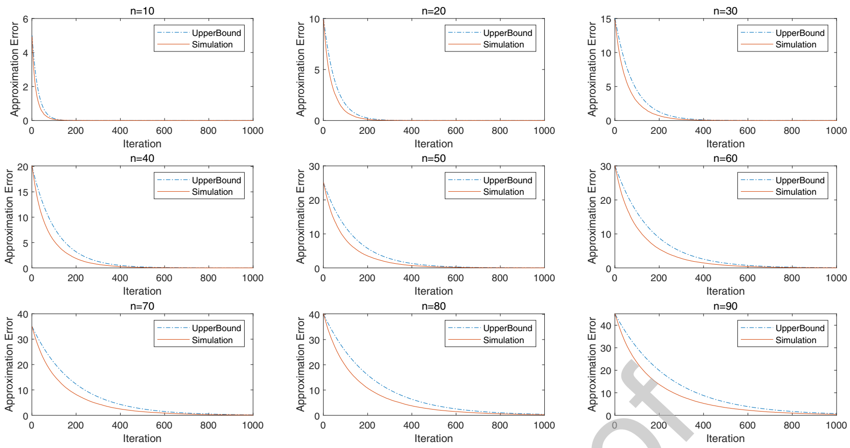


Figure 1: Comparison between the estimated upper bound and simulation results on expected approximation error of (1+1)EA solving the 10-, 20-,...,90-D OneMax problems.

5.2. The Peak Problem

Landscape of the Peak problem consists of a summit value 1 at $\mathbf{x}^* = (1, \ldots, 1)$ and a platform of value 0 for all other solutions. By defining the status index i as the total amount of 0-bits in a solution $\mathbf{x}$, we know $\tilde{\mathbf{e}} = (0, 1, \ldots, 1)'$. Correspondingly, the initial distribution is $\tilde{\mathbf{p}}^{[0]} = (C_n^0/2^n, C_n^1/2^n, C_n^2/2^n, \ldots, C_n^n/2^n)'$.

Theorem 6 *For RLS on the Peak problem,*

$$e^{[t]} = 1 - \frac{n+1}{2^n} + \frac{n}{2^n}\left(1 - \frac{1}{n}\right)^t.$$

Proof When the RLS is applied to the Peak problem, the one-bit mutation generates status transitions with probability

$$r_{i,j} = \begin{cases} 1/n, & i = 0, j = 1; \\ 1 - 1/n, & i = j = 1, \\ 1; & i = j \neq 1; \\ 0, & \text{otherwise.} \end{cases}$$

Thus, we get the diagonal transition submatrix of non-optimal statuses:

$$\mathbf{R} = diag\left(1 - \frac{1}{n}, 1, \ldots, 1\right).$$

Applying Theorem 1 we know that

$$e^{[t]} = \mathbf{e}'\mathbf{R}^t\mathbf{p}^{[0]} = \sum_{i=1}^{n} e_i r_{i,i}^t p_i = \left(1 - \frac{1}{n}\right)^t \frac{C_n^1}{2^n} + \sum_{i=2}^{n} \frac{C_n^i}{2^n} = 1 - \frac{n+1}{2^n} + \frac{n}{2^n}\left(1 - \frac{1}{n}\right)^t.$$

□

Since the error vector of non-optimal statuses is $\mathbf{e} = (1, \ldots, 1)'$, the obtained expected approximation error is equal to the probability to stay at non-optimal statuses. The optimal solution is achievable if and only if the initial solution is located at statuses 0 and 1. On the contrary, it cannot jump out of the fitness platform if the initial solution is not located adjacent to the global optimal solution. Thus, the probability to stay at non-optimal statuses would not converge to zero when $t \to \infty$, and its global convergence to the optimal solution cannot be guaranteed.

Theorem 7 *For (1+1)EA applied to the Peak problem,*

$$e^{[t]} = \sum_{i=1}^{n} \left[1 - \left(\frac{1}{n-1}\right)^i \left(1 - \frac{1}{n}\right)^n\right]^t \frac{C_n^i}{2^n}.$$

Proof When the (1+1)EA is employed to solve the Peak problem, the transition probability

$$r_{i,j} = \begin{cases} (1/n)^j (1-1/n)^{n-j}, & i = 0, j \neq 0; \\ 1, & i = j = 0, \\ 1 - (1/n)^j (1-1/n)^{n-j}; & i = j \neq 1; \\ 0, & \text{otherwise.} \end{cases}$$

Then, it holds that

$$\mathbf{R} = diag\left(1 - \frac{1}{n}\left(1 - \frac{1}{n}\right)^{n-1}, 1 - \left(\frac{1}{n}\right)^2 \left(1 - \frac{1}{n}\right)^{n-2}, \ldots, 1 - \left(\frac{1}{n}\right)^{n-1}\left(1 - \frac{1}{n}\right), 1 - \left(\frac{1}{n}\right)^n\right).$$

Applying Theorem 1, we know

$$e^{[t]} = \sum_{i=1}^{n} e_i r_{i,i}^t p_i = \sum_{i=1}^{n} \left[1 - \left(\frac{1}{n}\right)^i \left(1 - \frac{1}{n}\right)^{n-i}\right]^t \frac{C_n^i}{2^n} = \sum_{i=1}^{n} \left[1 - \left(\frac{1}{n-1}\right)^i \left(1 - \frac{1}{n}\right)^n\right]^t \frac{C_n^i}{2^n}.$$

□

Investigation on this case shows that error analysis can also work well when landscapes of problems have a platform. Because the transition submatrix is diagonal, computation of the expected approximation error is a trivial task that can be implemented easily.

5.3. The Deceptive Problem

According to definition of the Deceptive problem, we can get the following mapping from the total amount of 1-bits to the fitness and approximation error of $\mathbf{x}$.

$$\begin{array}{lccccc} |\mathbf{x}|: & 0 & 1 & \cdots & n-1 & n \\ & \updownarrow & \updownarrow & \cdots & \updownarrow & \downarrow \\ f(\mathbf{x}): & n-1 & n-2 & \cdots & 0 & n \\ & \updownarrow & \updownarrow & \cdots & \updownarrow & \updownarrow \\ e(\mathbf{x}): & 1 & 2 & \cdots & n & 0 \end{array} \tag{32}$$

Then, the feasible solution set could be divided into $n+1$ subsets, where $|\mathbf{x}| = 0$ corresponds to the locally optimal state. The initial approximation error and the initial distribution are presented as follows.

$$\tilde{\mathbf{e}} = (0, \mathbf{e}')' = (0, 1, 2, \ldots, n)', \tag{33}$$

$$\tilde{\mathbf{p}}^{[0]} = \left(\frac{C_n^n}{2^n}, \mathbf{p}'^{[0]}\right)' = \left(\frac{C_n^n}{2^n}, \frac{C_n^0}{2^n}, \frac{C_n^1}{2^n}, \ldots, \frac{C_n^{n-1}}{2^n}\right)'. \tag{34}$$

Because RLS employing a local search cannot escape from absorbing region of the local optimal solution, the expected approximation error cannot converge to 0 when the random initialization is employed. Accordingly, we get the following results on expected approximation error.

Theorem 8 *For RLS applied to the Deceptive problem,*

$$e^{[t]} = \left(1 - \frac{1}{2^{n-1}}\right) + \left(\frac{n}{2} - \frac{n}{2^{n-1}}\right)\left(1 - \frac{1}{n}\right)^t.$$

Proof For RLS applied to the Deceptive problem, we have

$$\mathbf{R} = (r_{i,j})_{n\times n} \begin{pmatrix} 1 & 1/n & & & \\ & 1-1/n & 2/n & & \\ & & \ddots & \ddots & \\ & & & 2/n & (n-1)/n \\ & & & & 0 \end{pmatrix}. \tag{35}$$

Then, Theorem 2 implies that

$$e^{[t]} = \mathbf{e}'\mathbf{R}^t\mathbf{p}^{[0]} = \sum_{j=1}^{n} \lambda_j^t \left(\mathbf{e}'\mathbf{p}_j\right)\left(\mathbf{q}_j'\mathbf{p}^{[0]}\right), \tag{36}$$

where $\lambda_j = r_{j,j}$, $\mathbf{p}_j$ and $\mathbf{q}_j$ are defined by (14) and (15), respectively. According to the derivation in Appendix B, we know the analytic expressions of $\mathbf{e}\mathbf{p}_j$ and $\mathbf{q}_j'\mathbf{p}^{[0]}$ are confirmed by (B.1), (B.2) and (B.3). Thus,

$$e^{[t]} = \lambda_1^t(\mathbf{e}'\mathbf{p}_1)(\mathbf{q}_1'\mathbf{p}^{[0]}) + \lambda_2^t(\mathbf{e}'\mathbf{p}_2)(\mathbf{q}_2'\mathbf{p}^{[0]}) = \left(1-\frac{1}{2^{n-1}}\right) + \left(\frac{n}{2}-\frac{n}{2^{n-1}}\right)\left(1-\frac{1}{n}\right)^t.$$

□

Furthermore, we would like to investigate the convergence performance of (1+1)EA on the Deceptive problem. To estimate the expected approximation error of (1+1)EA on the Deceptive problem, we need the result presented in the following lemma.

Lemma 6 *Consider a Markov chain model of Algorithm 1 whose transition matrix can be partitioned as*

$$\tilde{\mathbf{R}} = \begin{pmatrix} \hat{\mathbf{R}} & \hat{\mathbf{r}}^{[1]} \\ 0 & r_{L,L} \end{pmatrix}. \tag{37}$$

Correspondingly, denote

$$\tilde{\mathbf{e}} = (\hat{\mathbf{e}}', e_L)', \quad \tilde{\mathbf{p}}^{[0]} = (\hat{\mathbf{p}}^{[0]\prime}, p_L^{[0]})'.$$

Then, it holds for the expected approximation error that

$$e^{[t]} = \hat{\mathbf{e}}'\hat{\mathbf{R}}^t\hat{\mathbf{p}}^{[0]} + p_L^{[0]}\sum_{k=0}^{t-1} r_{L,L}^k \hat{\mathbf{e}}'\hat{\mathbf{R}}^{t-1-k}\hat{\mathbf{r}}^{[1]} + p_L^{[0]} e_L r_{L,L}^t.$$

Proof By (37), we know

$$e^{[t]} = \tilde{\mathbf{e}}'\tilde{\mathbf{R}}^t\tilde{\mathbf{p}}^{[0]} = (\hat{\mathbf{e}}', e_L)\begin{pmatrix} \hat{\mathbf{R}} & \hat{\mathbf{r}}^{[1]} \\ 0 & r_{L,L} \end{pmatrix}^t (\hat{\mathbf{p}}^{[0]\prime}, p_L^{[0]})' = (\hat{\mathbf{e}}', e_L)\begin{pmatrix} \hat{\mathbf{R}}^t & \hat{\mathbf{r}}^{[t]} \\ 0 & r_{L,L}^t \end{pmatrix}(\hat{\mathbf{p}}^{[0]\prime}, p_L^{[0]})',$$

where $\hat{\mathbf{r}}^{[t]} = \sum_{k=0}^{t-1} r_{L,L}^k \hat{\mathbf{R}}^{t-1-k}\hat{\mathbf{r}}^{[1]}$. Thus,

$$e^{[t]} = \hat{\mathbf{e}}'\hat{\mathbf{R}}^t\hat{\mathbf{p}}^{[0]} + p_L^{[0]}(\hat{\mathbf{e}}'\hat{\mathbf{r}}^{[t]} + e_L r_{L,L}^t) = \hat{\mathbf{e}}'\hat{\mathbf{R}}^t\hat{\mathbf{p}}^{[0]} + p_L^{[0]}\sum_{k=0}^{t-1} r_{L,L}^k \hat{\mathbf{e}}'\hat{\mathbf{R}}^{t-1-k}\hat{\mathbf{r}}^{[1]} + p_L^{[0]} e_L r_{L,L}^t.$$

□

Then, we can get an estimation about upper bound of the expected approximation error.

Theorem 9 *The expected approximation error of (1+1)EA for the Deceptive problem is bounded by*

$$e^{[t]} \le \left(1-\frac{n+1}{2^n}+\frac{en^2}{2^n}\right)\left[1-\left(\frac{1}{n}\right)^n\right]^t + \left(\frac{n}{2}-\frac{n}{2^n}+\frac{en^2}{2^n}\right)\left[1-\frac{1}{en}\right]^t + \frac{n^2}{2^n}\left[1-\frac{1}{e}\right]^t.$$

Proof Partition the transition matrix as

$$\tilde{\mathbf{R}} = \begin{pmatrix} \hat{\mathbf{R}} & \hat{\mathbf{r}}^{[1]} \\ 0 & r_{n,n} \end{pmatrix}, \tag{38}$$

where $\hat{\mathbf{R}} = (r_{i,j})_{i,j=0,1,\dots,n}$, $\hat{\mathbf{r}}^{[1]} = (r_{0,n}, r_{1,n}, \dots, r_{n-1,n})'$. Denote

$$\hat{\mathbf{e}} = (e_0, \dots, e_{n-1})' = (0, \dots, n-1)',$$
$$\hat{\mathbf{p}}^{[0]} = \left(p_0^{[0]}, \dots, p_{n-1}^{[0]}\right)' = \left(\frac{C_n^n}{2^n}, \frac{C_n^0}{2^n}, \frac{C_n^1}{2^n}, \dots, \frac{C_n^{n-2}}{2^n}\right)'.$$

Then, Lemma 6 implies that

$$e^{[t]} = \hat{\mathbf{e}}'\hat{\mathbf{R}}^t\hat{\mathbf{p}}^{[0]} + p_n^{[0]}(\hat{\mathbf{e}}'\hat{\mathbf{r}}^{[t]} + e_n r_{n,n}^t) = \hat{\mathbf{e}}'\hat{\mathbf{R}}^t\hat{\mathbf{p}}^{[0]} + p_n^{[0]}\sum_{k=0}^{t-1} r_{n,n}^k \hat{\mathbf{e}}'\hat{\mathbf{R}}^{t-1-k}\hat{\mathbf{r}}^{[1]} + p_n^{[0]} e_n r_{n,n}^t. \tag{39}$$

While the bitwise mutation is implemented, probability to flip j bits is $\left(\frac{1}{n}\right)^j\left(1-\frac{1}{n}\right)^{n-j}$, and that to flip one of j bits is $C_j^1\left(\frac{1}{n}\right)\left(1-\frac{1}{n}\right)^{n-1}$. The mapping illustrated in (32) indicates that

$$r_{0,j} = \left(\frac{1}{n}\right)^{n+1-j}\left(1-\frac{1}{n}\right)^{j-1}, \quad j = 1, \dots, n-1,$$
$$r_{j-1,j} \ge \frac{j-1}{n}\left(1-\frac{1}{n}\right)^{n-1}, \quad j = 2, \dots, n-1.$$

For the searching process characterized by $\hat{\mathbf{R}}$, we construct an auxiliary bi-diagonal search characterized by the transition matrix

$$\hat{\mathbf{S}} = \begin{pmatrix} 1 & (\frac{1}{n})^n & (\frac{1}{n})^n & \cdots & (\frac{1}{n})^n \\ & 1-(\frac{1}{n})^n & \frac{1}{n}(1-\frac{1}{n})^{n-1} & & \\ & & 1-(\frac{1}{n})^n - \frac{1}{n}(1-\frac{1}{n})^{n-1} & \ddots & \\ & & & \ddots & \frac{n-2}{n}(1-\frac{1}{n})^{n-1} \\ & & & & 1-(\frac{1}{n})^n - \frac{n-2}{n}(1-\frac{1}{n})^{n-1} \end{pmatrix}. \tag{40}$$

It is trivial to check that $\hat{\mathbf{R}}$ and $\hat{\mathbf{S}}$ satisfied conditions (19)-(21). Then, by Theorem 3 we know

$$\hat{\mathbf{e}}'\hat{\mathbf{R}}^t\hat{\mathbf{p}}^{[0]} \le \hat{\mathbf{e}}'\hat{\mathbf{S}}^t\hat{\mathbf{p}}^{[0]}, \tag{41}$$
$$\hat{\mathbf{e}}'\hat{\mathbf{R}}^{t-1-k}\hat{\mathbf{r}}^{[1]} \le \hat{\mathbf{e}}'\hat{\mathbf{S}}^{t-1-k}\hat{\mathbf{r}}^{[1]}. \tag{42}$$

Furthermore, denote

$$\check{\mathbf{R}} = (r_{i,j})_{i,j=1,\dots,n-1},$$
$$\check{\mathbf{e}} = (1, \dots, n-1)', \tag{43}$$
$$\check{\mathbf{p}}^{[0]} = \left(\frac{C_n^0}{2^n}, \frac{C_n^1}{2^n}, \dots, \frac{C_n^{n-2}}{2^n}\right)', \tag{44}$$
$$\check{\mathbf{r}}^{[1]} = (r_{1,n}, \dots, r_{n-1,n})', \tag{45}$$

and let

$$\check{\mathbf{S}} = (s_{i,j})_{i,j=1,\ldots,n-1}$$
$$= \begin{pmatrix} 1-(\frac{1}{n})^n & \frac{1}{n}(1-\frac{1}{n})^{n-1} & & \\ & 1-(\frac{1}{n})^n - \frac{1}{n}(1-\frac{1}{n})^{n-1} & \ddots & \\ & & \ddots & \frac{n-2}{n}(1-\frac{1}{n})^{n-1} \\ & & & 1-(\frac{1}{n})^n - \frac{n-2}{n}(1-\frac{1}{n})^{n-1} \end{pmatrix}. \tag{46}$$

By equation (39), we know

$$\begin{aligned} e^{[t]} &= \hat{\mathbf{e}}'\hat{\mathbf{R}}^t\hat{\mathbf{p}}^{[0]} + p_n^{[0]}\sum_{k=0}^{t-1} r_{n,n}^k \hat{\mathbf{e}}'\hat{\mathbf{R}}^{t-1-k}\hat{\mathbf{r}}^{[1]} + p_n^{[0]}e_n r_{n,n}^t \\ &\le \hat{\mathbf{e}}'\hat{\mathbf{S}}^t\hat{\mathbf{p}}^{[0]} + p_n^{[0]}\sum_{k=0}^{t-1} r_{n,n}^k \hat{\mathbf{e}}'\hat{\mathbf{S}}^{t-1-k}\hat{\mathbf{r}}^{[1]} + p_n^{[0]}e_n r_{n,n}^t \quad \text{(by (41) and (42))} \\ &= \check{\mathbf{e}}'\check{\mathbf{S}}^t\check{\mathbf{p}}^{[0]} + p_n^{[0]}\sum_{k=0}^{t-1} r_{n,n}^k \check{\mathbf{e}}'\check{\mathbf{S}}^{t-1-k}\check{\mathbf{r}}^{[1]} + p_n^{[0]}e_n r_{n,n}^t \quad \text{(by Lemma 1)} \\ &= \check{\mathbf{e}}'\left(\sum_{j=1}^{n-1}\lambda_j^t\check{\mathbf{p}}_j\check{\mathbf{q}}_j'\right)\check{\mathbf{p}}^{[0]} + p_n^{[0]}\sum_{k=0}^{t-1} r_{n,n}^k \check{\mathbf{e}}'\left(\sum_{j=1}^{n-1}\lambda_j^{t-1-k}\check{\mathbf{p}}_j\check{\mathbf{q}}_j'\right)\check{\mathbf{r}}^{[1]} + p_n^{[0]}e_n r_{n,n}^t \quad \text{(by Lemma 4)} \\ &= \sum_{j=1}^{n-1}\lambda_j^t(\check{\mathbf{e}}'\check{\mathbf{p}}_j)(\check{\mathbf{q}}_j'\check{\mathbf{p}}^{[0]}) + p_n^{[0]}\sum_{k=0}^{t-1} r_{n,n}^k \sum_{j=1}^{n-1}\lambda_j^{t-1-k}(\check{\mathbf{e}}'\check{\mathbf{p}}_j)(\check{\mathbf{q}}_j'\check{\mathbf{r}}^{[1]}) + p_n^{[0]}e_n r_{n,n}^t, \end{aligned} \tag{47}$$

where

$$\lambda_j = s_{j,j} = 1 - \left(\frac{1}{n}\right)^n - \frac{j-1}{n}\left(1-\frac{1}{n}\right)^{n-1}, \tag{48}$$

$$\check{\mathbf{p}}_j = \left(\prod_{k=1}^{j-1}\frac{s_{k,k+1}}{s_{j,j}-s_{k,k}}, \prod_{k=2}^{j-1}\frac{s_{k,k+1}}{s_{j,j}-s_{k,k}}, \ldots, \frac{s_{j-1,j}}{s_{j,j}-s_{j-1,j-1}}, 1, 0, \ldots, 0\right)', \tag{49}$$

$$\check{\mathbf{q}}_j' = \left(0, \ldots, 0, 1, \frac{s_{j,j+1}}{s_{j,j}-s_{j+1,j+1}}, \prod_{k=j+1}^{j+2}\frac{s_{k-1,k}}{s_{j,j}-s_{k,k}}, \ldots, \prod_{k=j+1}^{n-1}\frac{s_{k-1,k}}{s_{j,j}-s_{k,k}}\right), \tag{50}$$

$j = 1\ldots, n-1$. According to the derivation in Appendix C, we know the analytic expressions of $\check{\mathbf{e}}\check{\mathbf{p}}_j$ and $\check{\mathbf{q}}_j'\check{\mathbf{p}}^{[0]}$ are confirmed by (C.1), (C.2) and (C.3). Substituting (48), (C.1), (C.2) and (C.3) to (47) we know

$$\begin{aligned} e^{[t]} &\le \sum_{j=1}^{2}\lambda_j^t(\check{\mathbf{e}}'\check{\mathbf{p}}_j)(\check{\mathbf{q}}_j'\check{\mathbf{p}}^{[0]}) + p_n^{[0]}\sum_{j=1}^{2}\sum_{k=0}^{t-1} r_{n,n}^k\lambda_j^{t-1-k}(\check{\mathbf{e}}'\check{\mathbf{p}}_j)(\check{\mathbf{q}}_j'\check{\mathbf{r}}^{[1]}) + p_n^{[0]}e_n r_{n,n}^t \\ &\le \sum_{j=1}^{2}\lambda_j^t(\check{\mathbf{e}}'\check{\mathbf{p}}_j)(\check{\mathbf{q}}_j'\check{\mathbf{p}}^{[0]}) + p_n^{[0]}\left[\left(1-\frac{n+1}{n}\left(1-\frac{1}{n}\right)^{n-1}\right)\left(\frac{\lambda_1^t}{\frac{1}{n}(1-\frac{1}{n})^{n-1}} + \frac{\lambda_2^t}{\frac{1}{n}(1-\frac{1}{n})^{n-1}}\right) + e_n r_{n,n}^t\right] \\ &\le \left(1-\frac{n+1}{2^n}\right)\lambda_1^t + \left(\frac{n}{2}-\frac{n}{2^n}\right)\lambda_2^t + \frac{en^2}{2^n}(\lambda_1^t+\lambda_2^t) + \frac{n^2}{2^n}r_{n,n}^t \\ &\le \left(1-\frac{n+1}{2^n}+\frac{en^2}{2^n}\right)\left[1-\left(\frac{1}{n}\right)^n\right]^t + \left(\frac{n}{2}-\frac{n}{2^n}+\frac{en^2}{2^n}\right)\left[1-\frac{1}{en}\right]^t + \frac{n^2}{2^n}\left[1-\frac{1}{e}\right]^t. \end{aligned}$$

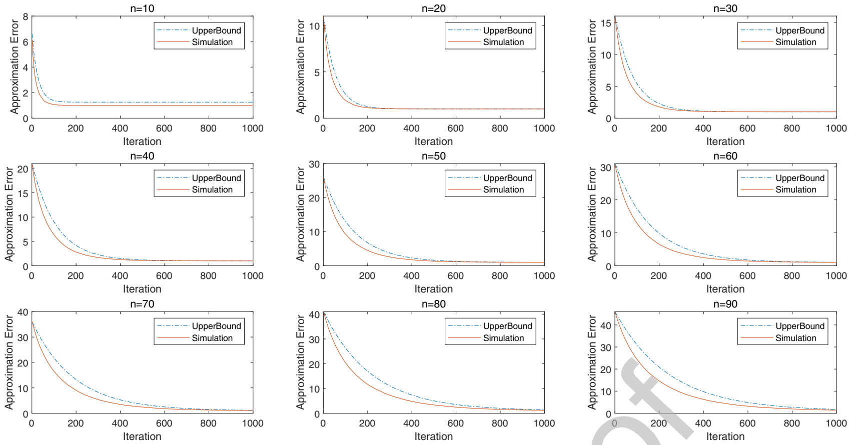


Figure 2: Comparison between the estimated upper bound and simulation results on expected approximation error of (1+1)EA solving the 10-, 20-,..., 90-D Deceptive problems.

Table 1: Running Time (RT) and Approximation Error (AE) of Randomized Search Heuristics on Investigated Problems

Index	Algorithm	Problem		
		OneMax	Peak	Deceptive
RT	RLS	$n\log n - 0.1159\ldots n \pm o(n)$[39]	∞	∞
	(1+1)EA	$(1-o(1))en\log n$[40]	$\Omega\left((n/2)^n\right)$[5]	$\Theta(n^n)$[41]
AE	RLS	$\frac{n}{2}\left(1-\frac{1}{n}\right)^t$	$1-\frac{n+1}{2^n}+\frac{n}{2^n}\left(1-\frac{1}{n}\right)^t$	$\left(1-\frac{1}{2^{n-1}}\right)+\left(\frac{n}{2}-\frac{n}{2^{n-1}}\right)\left(1-\frac{1}{n}\right)^t$
	(1+1)EA	$\frac{n}{2}\left(1-\frac{1}{en}\right)^t$	$\sum_{i=1}^{n}\left[1-\left(\frac{1}{n-1}\right)^i\left(1-\frac{1}{n}\right)^n\right]^t\frac{C_n^i}{2^n}$	$\left(1-\frac{n+1}{2^n}+\frac{en^2}{2^n}\right)\left[1-\left(\frac{1}{n}\right)^n\right]^t+$ $\left(\frac{n}{2}-\frac{n}{2^n}+\frac{en^2}{2^n}\right)\left[1-\frac{1}{en}\right]^t+\frac{n^2}{2^n}\left[1-\frac{1}{e}\right]^t$

□

Tightness of the estimated upper bound is again evaluated by comparison with simulation results illustrated in Figure 2. It is demonstrated that the error analysis method generates a tight upper bound for the approximation error of (1+1)EA solving the deceptive problem.

5.4. Competitiveness of Error Analysis to RT Estimation

Although there are a variety of publications regarding RT of RSHs for the OneMax problem, the Peak problem and the Deceptive problem, we only collect in Tab. 1 the state-of-the-art results about RLS and (1+1)EA. As is well-known, RLS and (1+1)EA can address the OneMax problem in RT of $\Theta(n\log n)$, and RTs of (1+1)EA rise exponentially with increase of the problem size of the Peak problem and the Deceptive problem. Because RLS cannot converge globally to the optimal solutions of the Peak problem and the Deceptive problem, the expected RTs are equal to infinity.

In principle, the result of error analysis is incomparable to that of RT analysis, because they are performed based on evaluation metrics that focus on different aspects of RSHs. However, error analysis leads to concrete estimation of approximation error, which could be much more informative. Recall that error analysis generates concrete estimation of the expected approximation error. There is no doubt that we can transform them into simple asymptotic results to validate the connection between approximation error and the problem dimension. Furthermore, estimation of the success probability can also been obtained via the concrete estimation of approximation error.

5.4.1. Asymptotic Analysis of Approximation Error

Considering that the polynomial iteration budget is available, we set $t = an^k(a > 0, k \in \mathbb{Z}^+)$. Then, the approximation error of RLS on the OneMax problem is $e^{[t]} = \frac{n}{2}\left(1-\frac{1}{n}\right)^{an^k}$, and it holds that $e^{[t]} < \frac{n}{2}\left(2^a\right)^{-n^{k-1}}, \quad n \geq 2$. That is, the asymptotic expected approximation error is $\mathcal{O}(nc^{n^{k-1}})$ for some constant $c \in (0,1)$. Similarly, the asymptotic expected approximation error of (1+1)EA for the OneMax problem is $\mathcal{O}(nc^{n^{k-1}})$ with the iteration budget $t = an^k$, $(a > 0, k \in \mathbb{Z}^+)$, where $c \in (0,1)$.

For the (1+1)EA applied to the Peak problem, a general result on the lower bound of $e^{[t]}$ can be obtained by considering that

$$\left[1-\left(\frac{1}{n-1}\right)^i\left(1-\frac{1}{n}\right)^n\right]^t \geq \left[1-\frac{1}{n-1}\left(1-\frac{1}{n}\right)^n\right]^t \geq \left[1-\frac{1}{2(n-1)}\right]^t.$$

Then, if the iteration budget is $t = an^k$, $(a > 0, k \in \mathbb{Z}^+)$, the asymptotic approximation error is $\Omega(c^{n^{k-1}})$, where $c \in (0,1)$.

5.4.2. Estimation of the Hitting Probability

Based on the analytic expression of the expected approximation error, a rough estimation of the probability to hit a given fitness level can be obtained via the Markov inequality [42]

$$\Pr\{e(t) \geq c\} \leq \frac{e^{[t]}}{c}, \tag{51}$$

where $e(t) = e(\mathbf{x}_t)$.

Take as an example the case that the (1+1)EA is applied to the OneMax problem. Considering that the expected RT is $(1-o(1))en\log n$, we set $t = aen\log n (a > 0)$. Formula (51) implies that

$$\Pr\{e(aen\log n) \geq 1\} \leq \frac{n}{2}\left(1-\frac{1}{en}\right)^{aen\log n} \leq \frac{1}{2n^{ae-1-\alpha(n)}},$$

where $\alpha(n) = e - \left(1+\frac{1}{en-1}\right)^{en-1}$. Then, from the fact that $\lim_{n\to\infty}\alpha(n) = 0$ we conclude that the probability to get the optimal solution of the OneMax problem is $1-\mathcal{O}\left(\frac{1}{n^{ae-1}}\right)$.

Furthermore, we can get concrete estimation of the success probability. Because

$$\alpha(n) = e - \left(1+\frac{1}{en-1}\right)^{en-1} < e-2,$$

it holds that

$$\Pr\{e(aen\log n) \geq 1\} \leq \frac{1}{2n^{ae-1-\alpha(n)}} < \frac{1}{2n^{(a-1+1/e)e}},$$

which converges to zero while $1 - a < \frac{1}{e}$. Thus, (1+1)EA would get the global optimal solution of the OneMax problem with an overwhelming probability if one sets $t > \left(1-\frac{1}{e}\right)en\log n$. This result presents a concrete threshold value of the iteration budget, which is more instructive for implement of RSHs than the results of RT analysis.

6. Error Analysis of the Knapsack Problem

In this section, we demonstrate the feasibility of error analysis for constrained optimization problems by investigating the knapsack problem

$$\begin{aligned} \max \quad & f(\mathbf{x}) = \sum_{i=1}^n p_i x_i, \\ s.t. \quad & \sum_{i=1}^n w_i x_i \leq W,\ x_i \in \{0,1\}, i = 1,\dots,n. \end{aligned} \tag{52}$$

Table 2: Parameters of the knapsack problem ($\alpha \in (0,1)$, $\alpha n \in \mathbb{Z}^+$).

$\mathbf{x}$	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_3$
	x_1	$x_2, \dots, x_{\alpha n}$	$x_{\alpha n+1}, \dots, x_n$
Item i	1	$2, \dots, \alpha n$	$\alpha n + 1, \dots, n$
Profit p_i	n	1	$\frac{1}{n}$
Weight w_i	n	$\frac{1}{\alpha n}$	n
P-W ratio $\frac{p_i}{w_i}$	1	αn	$\frac{1}{n^2}$
Capacity W	n		

An instance of the knapsack problem with parameters listed in Tab. 2 was proposed by He *et. al* [43], who showed that an $(N+1)$EA cannot obtain an α-approximate solution in polynomial FHT/RT. Global optimal solution of the investigated knapsack problem is $\mathbf{x}_g^* = (1, 0, \dots, 0)$ with $f(\mathbf{x}_g^*) = n$, and $\mathbf{x}_l^* = (0, 1, \dots, 1, 0, \dots 0)$ is the local optimal solution with $f(\mathbf{x}_l^*) = \alpha n - 1$. Since the penalty method would introduce an penalty parameter that functions on the fitness value of solutions, we employ a ratio-greedy repair mechanism to transform infeasible solutions into feasible ones. If an infeasible solution is generated, we sort all items according to the profit-to-weight(P-W) ratio, and items with small P-W ratio are successively removed from the knapsack until a feasible solution is achieved.

Denote a solution of the knapsack problem as $\mathbf{x} = (x_1, \dots, x_n)$, where $x_i \in \{0, 1\}$, $i = 1, \dots, n$. If item i is put into the knapsack, the corresponding binary variable x_i is set as '1'. According to the P-W ratios of items(variables), we can partition the solution vector $\mathbf{x}$ into three sub-vectors.

- $\mathbf{x}_1 = (x_1)$, which corresponds to the first item with the P-W ratio 1. The solution $\mathbf{x}_g^* = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) = (1, \mathbf{0}, \mathbf{0})$ represents the best packing solution that only contains the first item.
- $\mathbf{x}_2 = (x_2, \dots, x_{\alpha n})$, where variables correspond to the $2^{th} - \alpha n^{th}$ items with the P-W ratio αn. The sum total of weights of all these items is $1 - \frac{1}{\alpha n}$, and the total profit is $\alpha n - 1$. That is, $\mathbf{x}_l^* = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) = (0, \mathbf{1}, \mathbf{0})$ is a local optimal solution with $f(\mathbf{x}_l^*) = \alpha n - 1$.
- $\mathbf{x}_3 = (x_{\alpha n+1}, \dots, x_n)$, variables in which correspond to the last $n - \alpha n$ items with the P-W ratio $\frac{1}{n^2}$.

Recall that all items corresponding to $\mathbf{x}_1$ and $\mathbf{x}_3$ have a weight n, and the total weight of items in $\mathbf{x}_2$ is $1 - \frac{1}{\alpha n}$. A solution $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ is feasible if and only if at most one of sub-vectors $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_3$ is non-zero[2].

- If $|\mathbf{x}_1| = |\mathbf{x}_2| = |\mathbf{x}_3| = 0$, $\mathbf{x}$ represents an empty knapsack with $f(\mathbf{x}) = 0$.
- If $|\mathbf{x}_1| = 1$, $|\mathbf{x}_2| = |\mathbf{x}_3| = 0$, we get the global optimal solution $\mathbf{x}_g^*$ with $f(\mathbf{x}_g^*) = n$.
- If $|\mathbf{x}_2| \geq 1$, $|\mathbf{x}_1| = |\mathbf{x}_3| = 0$, we have $f(\mathbf{x}) = |\mathbf{x}_2|$.
- If $|\mathbf{x}_3| = 1$, $|\mathbf{x}_1| = |\mathbf{x}_2| = 0$, we have $f(\mathbf{x}) = \frac{1}{n}$.

If a generated solution $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$ contains more than one non-zero sub-vectors, it is infeasible and the ratio-greedy strategy is triggered to generate a feasible one. According to the values of P-W ratio, the repair strategy would first remove items represented by $\mathbf{x}_3$, then flip variables in $\mathbf{x}_1$ to zero. If an infeasible solution is in the form of $\mathbf{x} = (0, \mathbf{0}, \mathbf{x}_3)$ with $|\mathbf{x}_3| \geq 2$, the repair strategy would randomly delete redundant items until only one variable in $\mathbf{x}_3$ is '1'.

Accordingly, we get $\alpha n + 2$ statuses of feasible solutions labelled as $0, 1, \dots, \alpha n + 1$. We get the fitness vector

$$\tilde{\mathbf{f}} = (f_0, f_1, \dots, f_{\alpha n+1})' = \left(n, \alpha n - 1, \dots, 1, \frac{1}{n}, 0\right)',$$

[2]A vector $\mathbf{y}$ is non-zero means there exists at least one non-zero component in it, i.e., $|\mathbf{y}| > 0$.

and the corresponding error vector is

$$\tilde{\mathbf{e}} = (e_0, \mathbf{e}) = (e_0, \ldots, e_{\alpha n+1})' = \left(0, n-(\alpha n-1), \ldots, n-1, n-\frac{1}{n}, n\right)'. \tag{53}$$

6.1. Random Initialization and Initial Probability Distribution

Random initialization generates the initial distribution as follows.

1. If $\mathbf{x}_1 = 1$ and $|\mathbf{x}_2| = 0$, the global solution $\mathbf{x}^* = (1, 0, \ldots, 0)$ is obtained. Then, the initial solution is located at status 0 with probability $p_0^{[0]} = \frac{1}{2^{\alpha n}}$.
2. If $\alpha n - 1 \geq |\mathbf{x}_2| \geq 1$, it generates feasible solutions $\mathbf{x} = (\mathbf{0}, \mathbf{x_2}, \mathbf{0})$, no matter what the sub-vectors $\mathbf{x}_1$ and $\mathbf{x}_3$ are. For this case, a feasible solution with $f(\mathbf{x}) = |\mathbf{x}_2| = i$ is generated with probability $C_{\alpha n-1}^{i}(\frac{1}{2})^i(\frac{1}{2})^{\alpha n-1-i}$, $i = \alpha n-1, \ldots, 1$. Thus, we get the sub-vector of approximation error

   $$\mathbf{e}_1 = (e_1, \ldots, e_{\alpha n-1}) = (n-(\alpha n-1), n-(\alpha n-2), \ldots, n-1)',$$

   and the corresponding distribution sub-vector is

   $$\mathbf{p}_1^{[0]} = (p_1^{[0]}, p_2^{[0]}, \ldots, p_{\alpha n-1}^{[0]})' = \left(C_{\alpha n-1}^{\alpha n-1}\left(\frac{1}{2}\right)^{\alpha n-1}, C_{\alpha n-1}^{\alpha n-2}\left(\frac{1}{2}\right)^{\alpha n-1}, \ldots, C_{\alpha n-1}^{1}\left(\frac{1}{2}\right)^{\alpha n-1}\right)'.$$

3. If $|\mathbf{x}_1| = 0$, $|\mathbf{x}_2| = 0$ and $|\mathbf{x}_3| \geq 1$, the ratio-greedy strategy randomly delete redundant items represented by $\mathbf{x}_3$ until only one is remained. Consequently, the approximation error of feasible solutions is $n - \frac{1}{n}$, and we get the corresponding probability

   $$p_{\alpha n}^{[0]} = \sum_{i=1}^{n-\alpha n} C_{n-\alpha n}^{i}\left(\frac{1}{2}\right)^i\left(\frac{1}{2}\right)^{n-i} = \left(\frac{1}{2}\right)^{\alpha n} - \frac{1}{2^n}.$$

4. If $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) = (0, \ldots, 0)$, it generates a solution whose approximation error is n, and the corresponding probability is $p_{\alpha n+1}^{[0]} = \frac{1}{2^n}$.

In conclusion, we get the initial status distribution

$$\tilde{\mathbf{p}}^{[0]} = (p_0^{[0]}, \mathbf{p}_1^{[0]'}, p_{\alpha n+1}^{[0]})' = \left(\frac{1}{2^{\alpha n}}, C_{\alpha n-1}^{\alpha n-1}\left(\frac{1}{2}\right)^{\alpha n-1}, \ldots, C_{\alpha n-1}^{1}\left(\frac{1}{2}\right)^{\alpha n-1}, \left(\frac{1}{2}\right)^{\alpha n} - \frac{1}{2^n}, \frac{1}{2^n}\right)'.$$

For non-optimal statuses,

$$\mathbf{p}^{[0]} = (\mathbf{p}_1^{[0]'}, p_{\alpha n+1}^{[0]})' = \left(C_{\alpha n-1}^{\alpha n-1}\left(\frac{1}{2}\right)^{\alpha n-1}, \ldots, C_{\alpha n-1}^{1}\left(\frac{1}{2}\right)^{\alpha n-1}, \left(\frac{1}{2}\right)^{\alpha n} - \frac{1}{2^n}, \frac{1}{2^n}\right)'. \tag{54}$$

6.2. Expected Approximation Error of RSHs

6.2.1. Expected Approximation Error of RLS

Assisted by the ratio-greedy repair strategy, RLS generates status transitions detailed as follows.

1. **Status transition from** $\alpha n + 1$ **to** 0**,** $\alpha n - 1$ **or** αn. When the one-bit mutation is performed on $\mathbf{x} = (0, \ldots, 0)$, it generates a solution $\mathbf{y}$ with better fitness, which is accepted by the elitist selection. Consequently, the solution status transfers from $\alpha n+1$ to 0, $\alpha n-1$ or αn. The corresponding transition probabilities $p_{\mathbf{x}\to\mathbf{y}}$ are detailed in Tab. 3 as **Case 1**.
2. **Status transition from** αn **to** 0 **or** $\alpha n - 1$. While the one-bit mutation is performed on a feasible solution $\mathbf{x} = (0, \mathbf{0}, \mathbf{x}_3)$ with $|\mathbf{x}_3| = 1$, any flip of $\mathbf{x}_3$ from '0' to '1' generates an infeasible solution, and the greedy-repair strategy would convert it to another solution at status αn, which would not result in status transition. If one bit in $\mathbf{x}_1$ or $\mathbf{x}_2$ is flipped to '1', the repair strategy will keep it and flip the '1' in $\mathbf{x}_3$ to '0', which results in the status transition from αn to 0 or $\alpha n - 1$. This case is labeled in Tab. 3 as **Case 2**.

Table 3: Status transitions and the corresponding probabilities generated by RLS.

		Case 1			**Case 2**		**Case 3**
$\mathbf{x}$	Status	$\alpha n+1$			αn		$k,(k=2,\dots,\alpha n-1)$
	$f(\mathbf{x})$	0			$\frac{1}{n}$		$\alpha n-k,(k=\alpha n-1,\dots,2)$
	$e(\mathbf{x})$	n			$n-\frac{1}{n}$		$(1-\alpha)n+k$
$\mathbf{y}$	Status	αn	$\alpha n-1$	0	$\alpha n-1$	0	$k-1$
	$f(\mathbf{y})$	$\frac{1}{n}$	1	n	1	n	$\alpha n-k+1$
	$e(\mathbf{y})$	$n-\frac{1}{n}$	$n-1$	0	$n-1$	0	$(1-\alpha)n+k-1$
$p_{\mathbf{x}\to\mathbf{y}}$		$\frac{n-\alpha n}{n}$	$\frac{\alpha n-1}{n}$	$\frac{1}{n}$	$\frac{\alpha n-1}{n}$	$\frac{1}{n}$	$\frac{k-1}{n}$

3. **Status transition from k to $k-1$, $k=2,\dots,\alpha n-1$.** While the present solution is at status k, $k=2,\dots,\alpha n-1$, the corresponding solution is $\mathbf{x}=(0,\mathbf{x}_2,\mathbf{0})$ with $|\mathbf{x}_2|=\alpha n-k$. Then, a candidate solution is accepted if and only if it is generated by flipping another '0' in $\mathbf{x}_2$ to '1'. The status transition is denoted as **Case 3** in Tab. 3.
4. If the present solution is at status 1, the one-bit mutation cannot generate a better solution any more. Then, the iteration process would stagnate at the local optima.

According to the results presented in Tab. 3, we get the transition matrix

$$\tilde{\mathbf{R}}=\begin{pmatrix}\hat{\mathbf{R}} & \hat{\mathbf{r}}^{[1]}\\ \mathbf{0} & 0\end{pmatrix}, \tag{55}$$

where

$$\hat{\mathbf{r}}^{[1]}=(r_{0,\alpha n+1},\dots,r_{n,\alpha n+1})'=(1/n,0,\dots,0,\alpha-1/n,1-\alpha)', \tag{56}$$

$$\hat{\mathbf{R}}=\begin{pmatrix}1 & \check{\mathbf{r}}\\ 0 & \check{\mathbf{R}}\end{pmatrix}=\begin{pmatrix}1 & 0 & 0 & \cdots & 0 & 0 & \frac{1}{n}\\ & 1 & \frac{1}{n} & \cdots & 0 & 0 & 0\\ & & 1-\frac{1}{n} & \ddots & \vdots & \vdots & \vdots\\ & & & \ddots & \vdots & \vdots & \vdots\\ & & & & \alpha-\frac{3}{n} & 0 & 0\\ & & & & 1-(\alpha-\frac{3}{n}) & \alpha-\frac{2}{n} & 0\\ & & & & & 1-(\alpha-\frac{2}{n}) & \alpha-\frac{1}{n}\\ & & & & & & 1-\alpha\end{pmatrix}. \tag{57}$$

Then, we can get tight upper and lower bounds for the expected approximation error of RLS.

Theorem 10 *For RLS on the Knapsack problem, the expected approximation error is bounded by*

$$\begin{aligned}&(n-\alpha n+1)\left[1-\frac{1}{2^{\alpha n-1}}-\frac{1}{2^n}\right]+\frac{\alpha n-2}{4}\left(1-\frac{1}{n}\right)^{t-1}+\frac{n+1}{\alpha n}\frac{1}{2^{\alpha n}}(1-\alpha)^t\\ &\le e^{[t]}\le(n-\alpha n+1)\left[1-\frac{1}{2^{\alpha n}}+\frac{1}{2^n}\right]+\frac{\alpha n-1}{2}\left(1-\frac{1}{n}\right)^{t-1}+\frac{n+1}{\alpha n}\frac{1}{2^{\alpha n}}(1-\alpha)^t.\end{aligned}$$

Proof By Lemma 6, we know

$$e^{[t]}=\hat{\mathbf{e}}'\hat{\mathbf{R}}^t\hat{\mathbf{p}}^{[0]}+p^{[0]}_{\alpha n+1}\hat{\mathbf{e}}'\hat{\mathbf{R}}^{t-1}\hat{\mathbf{r}}^{[1]},$$

where $\hat{\mathbf{R}}$ is given by equation (57),

$$\hat{\mathbf{e}} = (e_0, \dots, e_{\alpha n})' = \left(0, n - (\alpha n - 1), \dots, n - 1, n - \frac{1}{n}\right)', \tag{58}$$

$$\hat{\mathbf{p}}^{[0]} = (p_0^{[0]}, \dots, p_{\alpha n}^{[0]})' = \left(\frac{1}{2^{\alpha n}}, C_{\alpha n-1}^{\alpha n-1}\left(\frac{1}{2}\right)^{\alpha n-1}, \dots, C_{\alpha n-1}^{1}\left(\frac{1}{2}\right)^{\alpha n-1}, \left(\frac{1}{2}\right)^{\alpha n} - \frac{1}{2^n}\right)', \tag{59}$$

Then, Lemma 4 and Theorem 2 imply that

$$\begin{aligned} e^{[t]} &= \check{\mathbf{e}}' \left(\sum_{j=1}^{\alpha n} \lambda_j^t \check{\mathbf{p}}_j \check{\mathbf{q}}_j'\right) \check{\mathbf{p}}^{[0]} + p_{\alpha n+1}^{[0]} \check{\mathbf{e}}' \left(\sum_{j=1}^{\alpha n} \lambda_j^{t-1} \check{\mathbf{p}}_j \check{\mathbf{q}}_j'\right) \check{\mathbf{r}}^{[0]} \\ &= \sum_{j=1}^{\alpha n} \lambda_j^t \left(\check{\mathbf{e}}' \check{\mathbf{p}}_j\right) \left(\check{\mathbf{q}}_j' \check{\mathbf{p}}^{[0]}\right) + p_{\alpha n+1}^{[0]} \sum_{j=1}^{\alpha n} \lambda_j^{t-1} \left(\check{\mathbf{e}}' \check{\mathbf{p}}_j\right) \left(\check{\mathbf{q}}_j' \check{\mathbf{r}}^{[1]}\right), \end{aligned} \tag{60}$$

where

$$\lambda_j = r_{j,j} = \begin{cases} 1 - \frac{j-1}{n}, & j = 1, \dots, \alpha n - 1, \\ 1 - \frac{j}{n}, & j = \alpha n, \end{cases} \tag{61}$$

,

$$\check{\mathbf{e}} = (e_1, \dots, e_{\alpha n})' = \left(n - (\alpha n - 1), \dots, n - 1, n - \frac{1}{n}\right)', \tag{62}$$

$$\check{\mathbf{p}}^{[0]} = (p_1^{[0]'}, \dots, p_{\alpha n}^{[0]})' = \left(C_{\alpha n-1}^{\alpha n-1}\left(\frac{1}{2}\right)^{\alpha n-1}, \dots, C_{\alpha n-1}^{1}\left(\frac{1}{2}\right)^{\alpha n-1}, \left(\frac{1}{2}\right)^{\alpha n} - \frac{1}{2^n}\right)', \tag{63}$$

$$\check{\mathbf{r}}^{[1]} = (r_{1,\alpha n+1}, \dots, r_{n,\alpha n+1})' = (0, \dots, 0, \alpha - 1/n, 1 - \alpha)', \tag{64}$$

$$\mathbf{p}_j = \left(\prod_{k=1}^{j-1} \frac{r_{k,k+1}}{r_{j,j} - r_{k,k}}, \prod_{k=2}^{j-1} \frac{r_{k,k+1}}{r_{j,j} - r_{k,k}}, \dots, \frac{r_{j-1,j}}{r_{j,j} - r_{j-1,j-1}}, 1, 0, \dots, 0\right)', \tag{65}$$

$$\mathbf{q}_j' = \left(0, \dots, 0, 1, \frac{r_{j,j+1}}{r_{j,j} - r_{j+1,j+1}}, \prod_{k=j+1}^{j+2} \frac{r_{k-1,k}}{r_{j,j} - r_{k,k}}, \dots, \prod_{k=j+1}^{\alpha n} \frac{r_{k-1,k}}{r_{j,j} - r_{k,k}}\right), \quad j = 1 \dots, \alpha n. \tag{66}$$

By derivation presented in Appendix D, we get the expressions of components in (60). Substituting (61), (D.1), (D.2), (D.3), (D.4), (D.5) and (D.6) to (60), we know that

$$\begin{aligned} e^{[t]} =& \lambda_1^{t-1} \left(\check{\mathbf{e}}' \check{\mathbf{p}}_1\right) \left(\lambda_1 \check{\mathbf{q}}_1' \check{\mathbf{p}}^{[0]} + p_{\alpha n+1}^{[0]} \check{\mathbf{q}}_1' \check{\mathbf{r}}^{[1]}\right) + \lambda_2^{t-1} \left(\check{\mathbf{e}}' \check{\mathbf{p}}_2\right) \left(\lambda_2 \check{\mathbf{q}}_2' \check{\mathbf{p}}^{[0]} + p_{\alpha n+1}^{[0]} \check{\mathbf{q}}_2' \check{\mathbf{r}}^{[1]}\right) \\ &+ \lambda_{\alpha n}^{t-1} \left(\check{\mathbf{e}}' \check{\mathbf{p}}_{\alpha n}\right) \left(\lambda_{\alpha n} \check{\mathbf{q}}_{\alpha n}' \check{\mathbf{p}}^{[0]} + p_{\alpha n+1}^{[0]} \check{\mathbf{q}}_{\alpha n}' \check{\mathbf{r}}^{[1]}\right), \end{aligned} \tag{67}$$

which is bounded by

$$\begin{aligned} &(n - \alpha n + 1)\left[1 - \frac{1}{2^{\alpha n-1}} - \frac{1}{2^n}\right] + \frac{\alpha n - 2}{4}\left(1 - \frac{1}{n}\right)^{t-1} + \frac{n+1}{\alpha n}\frac{1}{2^{\alpha n}}(1-\alpha)^t \\ &\le e^{[t]} \le (n - \alpha n + 1)\left[1 - \frac{1}{2^{\alpha n}} + \frac{1}{2^n}\right] + \frac{\alpha n - 1}{2}\left(1 - \frac{1}{n}\right)^{t-1} + \frac{n+1}{\alpha n}\frac{1}{2^{\alpha n}}(1-\alpha)^t. \end{aligned}$$

□

Note that formula (67) presents a general expression dependent on any initial distribution. Because the Knapsack problem has a local absorbing region where individuals cannot jump out, the first eigenvalue λ_1 is sure to be equal to 1, and (67) implies that $e^{[t]}$ converges to 0 if and only if $(\check{\mathbf{e}}'\check{\mathbf{p}}_1)\left(\lambda_1\check{\mathbf{q}}_1'\check{\mathbf{p}}^{[0]} + p_{\alpha n+1}^{[0]}\check{\mathbf{q}}_1'\check{\mathbf{r}}^{[1]}\right)$ equals to 0, which is further equivalent to the statement that both $\check{\mathbf{p}}^{[0]}$ and $p_{\alpha n+1}^{[0]}$ are equal to zero. This statement means that any initial strategy that does not generate the global optimal solution with probability 1 cannot guarantee convergence of the RLS to the global optimal solution.

6.2.2. Expected Approximation Error of (1+1)EA

Denote $\tilde{p}_{i,j}$ as the probability to transfer from status j to status i. When the bitwise mutation is employed, $\tilde{p}_{i,j}$ is estimated as follows.

- While status j transfers to status 0, $j = 1, \ldots, \alpha n + 1$,

$$\tilde{p}_{0,j} = \begin{cases} \left(\frac{1}{n}\right)^{\alpha n+1-j}\left(1-\frac{1}{n}\right)^{j-1}, & j = 1, \ldots, \alpha n - 1, \\ \frac{1}{n}\left(1-\frac{1}{n}\right)^{\alpha n-1}, & j = \alpha n, \\ \frac{1}{n}\left(1-\frac{1}{n}\right)^{\alpha n-1}, & , j = \alpha n + 1. \end{cases} \tag{68}$$

- The probability to transfer from status j to status i $(1 \leq i < j)$ is

$$\tilde{p}_{i,j} \geq \begin{cases} C_{j-1}^{j-i}\left(\frac{1}{n}\right)^{j-i}\left(1-\frac{1}{n}\right)^{(\alpha n-1)-(j-i)}, & j = 1, \ldots, \alpha n - 1, \\ C_{\alpha n-1}^{\alpha n-i}\left(\frac{1}{n}\right)^{\alpha n-i}\left(1-\frac{1}{n}\right)^{i-1}, & j = \alpha n, \alpha n + 1, 1 \leq i \leq \alpha n - 1, \\ \left(1-\frac{1}{n}\right)^{\alpha n} - \left(1-\frac{1}{n}\right)^{n}, & , j = \alpha n + 1, i = \alpha n. \end{cases} \tag{69}$$

Note that we get identical lower bounds of $\tilde{p}_{i,\alpha n}$ and $\tilde{p}_{i,\alpha n+1}$ for $1 \leq i \leq \alpha n - 1$. Because the difference between $e_{\alpha n}$ and $e_{\alpha n+1}$ is $\frac{1}{n}$, an infinitesimal that could be ignored, we combine statuses αn and $\alpha n + 1$ together as the αn-th status. Taking n as the approximation error of the newly defined status αn, we get the error vector

$$\tilde{\mathbf{e}}_R = (e_0, \ldots, e_{\alpha n-1}, e_{\alpha n+1})' = (0, n - (\alpha n - 1), \ldots, n - 1, n)'. \tag{70}$$

Correspondingly, the initial distribution is

$$\tilde{\mathbf{p}}_R^{[0]} = (p_0^{[0]}, \ldots, p_{\alpha n-1}^{[0]}, p_{\alpha n}^{[0]} + p_{\alpha n+1}^{[0]})' = \left(\frac{1}{2^{\alpha n}}, C_{\alpha n-1}^{\alpha n-1}\left(\frac{1}{2}\right)^{\alpha n-1}, \ldots, C_{\alpha n-1}^{1}\left(\frac{1}{2}\right)^{\alpha n-1}, \left(\frac{1}{2}\right)^{\alpha n}\right)'. \tag{71}$$

Let $\tilde{\mathbf{R}} = (r_{i,j})_{i,j=0,1,\ldots,\alpha n}$ be the transition matrix regarding the redefined individual statuses. Because approximation error of the combined status is taken as the bigger one of two combined statuses, the approximation error is bounded from above by

$$e^{[t]} = \tilde{\mathbf{e}}'\tilde{\mathbf{P}}^t\tilde{\mathbf{p}}^{[0]} \leq \tilde{\mathbf{e}}_R'\tilde{\mathbf{R}}^t\tilde{\mathbf{p}}_R^{[0]}. \tag{72}$$

In the following, we get the upper bound of $e^{[t]}$ by estimating $\tilde{\mathbf{e}}_R'\tilde{\mathbf{R}}^t\tilde{\mathbf{p}}_R^{[0]}$.

Theorem 11 *The expected approximation error of (1+1)EA for the Knapsack problem is bounded by*

$$
\begin{aligned}
e^{[t]} \le& \left[(1-\alpha)n+1+\tfrac{1}{\alpha(1-\alpha)2^{\alpha n-1}}\left(1-\left(1-\tfrac{1}{n}\right)^{\alpha n-1}\right)\right]\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}\right]^t \\
&+\left\{\tfrac{(2^{\alpha n-2}-1)(\alpha n-1)}{2^{\alpha n-1}}+\tfrac{\alpha n-1}{2^{\alpha n-2}\alpha(1-\alpha)}\left[1-\left(1-\tfrac{1}{n}\right)^{\alpha n-1}\right]\right\}\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}-\tfrac{1}{n}\left(1-\tfrac{1}{n}\right)^{\alpha n-2}\right]^t \\
&+\left(n-\tfrac{1}{n}\right)\tfrac{1}{2^{\alpha n}}\left(1-\tfrac{1}{n}\right)^{\alpha n t}.
\end{aligned}
$$

Proof Partition the transition matrix as

$$
\tilde{\mathbf{R}} = \begin{pmatrix} \hat{\mathbf{R}} & \hat{\mathbf{r}}^{[1]} \\ 0 & r_{\alpha n,\alpha n} \end{pmatrix} \tag{73}
$$

where $\hat{\mathbf{R}} = (r_{i,j})_{\alpha n\times\alpha n}, i,j=0,1,\ldots,\alpha n-1$, $\hat{\mathbf{r}}^{[1]} = (r_{0,\alpha n}, r_{1,\alpha n},\ldots,r_{\alpha n-1,\alpha n})'$. When the present status is αn, the status would keep unchanged if the first αn bits are not flipped from '0' to '1'. That is,

$$
r_{\alpha n,\alpha n} = \left(1-\frac{1}{n}\right)^{\alpha n} \tag{74}
$$

By substituting (70) and (71) to (72), Lemma 6 implies that

$$
e^{[t]} \le \hat{\mathbf{e}}_R'\hat{\mathbf{R}}^t\hat{\mathbf{p}}_R^{[0]} + (p_{\alpha n}^{[0]}+p_{\alpha n+1}^{[0]})\sum_{k=0}^{t-1} r_{\alpha n,\alpha n}^k \hat{\mathbf{e}}_R'\hat{\mathbf{R}}^{t-1-k}\hat{\mathbf{r}}^{[1]} + (p_{\alpha n}^{[0]}+p_{\alpha n+1}^{[0]})e_{\alpha n+1}r_{\alpha n,\alpha n}^t, \tag{75}
$$

where

$$
\hat{\mathbf{e}}_R = (e_0,\ldots,e_{\alpha n-1})' = (0, n-(\alpha n-1),\ldots,n-1)',
$$

$$
\hat{\mathbf{p}}_R^{[0]} = (p_0^{[0]},\ldots,p_{\alpha n-1}^{[0]})' = \left(\frac{1}{2^{\alpha n}}, C_{\alpha n-1}^{\alpha n-1}\left(\frac{1}{2}\right)^{\alpha n-1},\ldots,C_{\alpha n-1}^{1}\left(\frac{1}{2}\right)^{\alpha n-1}\right)'.
$$

Furthermore, denote

$$
\mathbf{e}_R = (e_1,\ldots,e_{\alpha n-1})' = (n-(\alpha n-1),\ldots,n-1)', \tag{76}
$$

$$
\mathbf{p}_R^{[0]} = (p_1^{[0]},\ldots,p_{\alpha n-1}^{[0]})' = \left(C_{\alpha n-1}^{\alpha n-1}\left(\frac{1}{2}\right)^{\alpha n-1},\ldots,C_{\alpha n-1}^{1}\left(\frac{1}{2}\right)^{\alpha n-1}\right)', \tag{77}
$$

$$
\mathbf{r}^{[1]} = (r_{1,\alpha n},\ldots,r_{\alpha n-1,\alpha n})', \tag{78}
$$

and construct an auxiliary transition matrix

$$
\hat{\mathbf{S}} = (s_{i,j})_{\alpha n\times\alpha n} = \begin{pmatrix} 1 & \mathbf{s} \\ 0 & \mathbf{S} \end{pmatrix}, \quad i,j=0,1,\ldots,\alpha n-1, \tag{79}
$$

where

$$
\mathbf{s} = \left(\left(\frac{1}{n}\right)^{\alpha n}, \left(\frac{1}{n}\right)^{\alpha n},\cdots,\left(\frac{1}{n}\right)^{\alpha n}\right),
$$

$$
\mathbf{S} = (s_{i,j})_{i,j=1,\ldots,\alpha n-1}
$$

$$
\begin{pmatrix}
1-\left(\frac{1}{n}\right)^{\alpha n} & \frac{1}{n}\left(1-\frac{1}{n}\right)^{\alpha n-2} & \cdots & 0 & 0 \\
 & 1-\left(\frac{1}{n}\right)^{\alpha n}-\frac{1}{n}\left(1-\frac{1}{n}\right)^{\alpha n-2} & \cdots & 0 & 0 \\
 & & \ddots & \vdots & \vdots \\
 & & & \frac{\alpha n-3}{n}\left(1-\frac{1}{n}\right)^{\alpha n-2} & 0 \\
 & & & 1-\left(\frac{1}{n}\right)^{\alpha n}-\frac{\alpha n-3}{n}\left(1-\frac{1}{n}\right)^{\alpha n-2} & \frac{\alpha n-2}{n}\left(1-\frac{1}{n}\right)^{\alpha n-2} \\
 & & & & 1-\left(\frac{1}{n}\right)^{\alpha n}-\frac{\alpha n-2}{n}\left(1-\frac{1}{n}\right)^{\alpha n-2}
\end{pmatrix}. \tag{80}
$$

It is trivial to check that matrices $\hat{\mathbf{R}}$ and $\hat{\mathbf{S}}$ satisfy conditions (19)-(21). By applying Theorem 3 and Lemma 1, (75) implies that

$$\begin{aligned} e^{[t]} &\le \hat{\mathbf{e}}_R'\hat{\mathbf{S}}^t\hat{\mathbf{p}}_R^{[0]} + (p_{\alpha n}^{[0]} + p_{\alpha n+1}^{[0]})\sum_{k=0}^{t-1} r_{\alpha n,\alpha n}^k \hat{\mathbf{e}}_R'\hat{\mathbf{S}}^{t-1-k}\hat{\mathbf{r}}^{[1]} + (p_{\alpha n}^{[0]} + p_{\alpha n+1}^{[0]})e_{\alpha n+1}r_{\alpha n,\alpha n}^t \\ &= \mathbf{e}_R'\mathbf{S}^t\mathbf{p}_R^{[0]} + (p_{\alpha n}^{[0]} + p_{\alpha n+1}^{[0]})\sum_{k=0}^{t-1} r_{\alpha n,\alpha n}^k \mathbf{e}_R'\mathbf{S}^{t-1-k}\mathbf{r}^{[1]} + (p_{\alpha n}^{[0]} + p_{\alpha n+1}^{[0]})e_{\alpha n+1}r_{\alpha n,\alpha n}^t \end{aligned} \tag{81}$$

Furthermore, by applying Lemma 4 we conclude that

$$\begin{aligned} e^{[t]} \le &\sum_{j=1}^{\alpha n-1} \lambda_j^t (\mathbf{e}_R'\mathbf{p}_j)(\mathbf{q}_j'\mathbf{p}_R^{[0]}) + (p_{\alpha n}^{[0]} + p_{\alpha n+1}^{[0]})\sum_{k=0}^{t-1} r_{\alpha n,\alpha n}^k \sum_{j=1}^{\alpha n-1} \lambda_j^{t-1-k}(\mathbf{e}_R'\mathbf{p}_j)(\mathbf{q}_j'\mathbf{r}^{[1]}) \\ &+ (p_{\alpha n}^{[0]} + p_{\alpha n+1}^{[0]})e_{\alpha n+1}r_{\alpha n,\alpha n}^t, \end{aligned} \tag{82}$$

where

$$\lambda_j = s_{j,j} = 1 - \left(\frac{1}{n}\right)^{\alpha n} - \frac{j-1}{n}\left(1 - \frac{1}{n}\right)^{\alpha n-2}, \tag{83}$$

$$\mathbf{p}_j = \left(\prod_{k=1}^{j-1}\frac{s_{k,k+1}}{s_{j,j}-s_{k,k}}, \prod_{k=2}^{j-1}\frac{s_{k,k+1}}{s_{j,j}-s_{k,k}}, \dots, \frac{s_{j-1,j}}{s_{j,j}-s_{j-1,j-1}}, 1, 0, \dots, 0\right)', \tag{84}$$

$$\mathbf{q}_j' = \left(0, \dots, 0, 1, \frac{s_{j,j+1}}{s_{j,j}-s_{j+1,j+1}}, \prod_{k=j+1}^{j+2}\frac{s_{k-1,k}}{s_{j,j}-s_{k,k}}, \dots, \prod_{k=j+1}^{n}\frac{s_{k-1,k}}{s_{j,j}-s_{k,k}}\right), \tag{85}$$

$j = 1\dots,\alpha n-1$. According to the derivation presented in Appendix E, we know the expression of $\mathbf{e}_R'\mathbf{p}_j$, $\mathbf{q}_j'\mathbf{p}_R^{[0]}$ and $\mathbf{q}_j'\mathbf{r}^{[1]}$ are confirmed by (E.1), (E.2) and (E.3). Substituting (74), (83), (E.1), (E.2) and (E.3) to (82), we conclude that

$$\begin{aligned} e^{[t]} \le\ &(n-\alpha n+1)\left(1-\tfrac{1}{2^{\alpha n-1}}\right)\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}\right]^t + (\alpha n-1)\left(\tfrac{1}{2}-\tfrac{1}{2^{\alpha n-1}}\right)\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}-\tfrac{1}{n}\left(1-\tfrac{1}{n}\right)^{\alpha n-2}\right]^t \\ &+ \tfrac{1}{2^{\alpha n}}\left[1-\left(1-\tfrac{1}{n}\right)^{\alpha n-1}\right]\Big\{\textstyle\sum_{k=0}^{t-1}\left[\left(1-\tfrac{1}{n}\right)^{\alpha n}\right]^k\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}\right]^{t-1-k} \\ &+(\alpha n-1)\textstyle\sum_{k=0}^{t-1}\left[\left(1-\tfrac{1}{n}\right)^{\alpha n}\right]^k\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}-\tfrac{1}{n}\left(1-\tfrac{1}{n}\right)^{\alpha n-2}\right]^{t-1-k}\Big\} + \left(n-\tfrac{1}{n}\right)\tfrac{1}{2^{\alpha n}}\left(1-\tfrac{1}{n}\right)^{\alpha n t} \\ \le\ &(n-\alpha n+1)\left(1-\tfrac{1}{2^{\alpha n-1}}\right)\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}\right]^t + (\alpha n-1)\left(\tfrac{1}{2}-\tfrac{1}{2^{\alpha n-1}}\right)\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}-\tfrac{1}{n}\left(1-\tfrac{1}{n}\right)^{\alpha n-2}\right]^t \\ &+ \tfrac{1}{2^{\alpha n}}\left[1-\left(1-\tfrac{1}{n}\right)^{\alpha n-1}\right]\left\{\tfrac{2}{\alpha(1-\alpha)}\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}\right]^t + \tfrac{4(\alpha n-1)}{\alpha(1-\alpha)}\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}-\tfrac{1}{n}\left(1-\tfrac{1}{n}\right)^{\alpha n-2}\right]^t\right\} \\ &+ \left(n-\tfrac{1}{n}\right)\tfrac{1}{2^{\alpha n}}\left(1-\tfrac{1}{n}\right)^{\alpha n t} \\ \le\ &\left[(1-\alpha)n+1+\tfrac{1}{\alpha(1-\alpha)2^{\alpha n-1}}\left(1-\left(1-\tfrac{1}{n}\right)^{\alpha n-1}\right)\right]\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}\right]^t \\ &+ \left\{\tfrac{(2^{\alpha n-2}-1)(\alpha n-1)}{2^{\alpha n-1}} + \tfrac{\alpha n-1}{2^{\alpha n-2}\alpha(1-\alpha)}\left[1-\left(1-\tfrac{1}{n}\right)^{\alpha n-1}\right]\right\}\left[1-\left(\tfrac{1}{n}\right)^{\alpha n}-\tfrac{1}{n}\left(1-\tfrac{1}{n}\right)^{\alpha n-2}\right]^t \\ &+ \left(n-\tfrac{1}{n}\right)\tfrac{1}{2^{\alpha n}}\left(1-\tfrac{1}{n}\right)^{\alpha n t}. \end{aligned}$$

□

As illustrated in Figure 3, it is showed that the estimated upper bound for the Knapsack problem is tight, too. Although global exploration ability can be achieved by the bitwise mutation, the (1+1)EA always

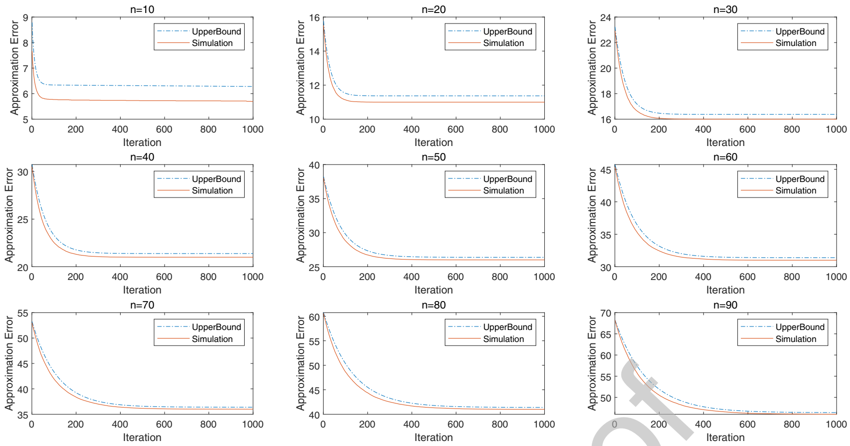


Figure 3: Comparison between the estimated upper bound and simulation results on expected approximation error of (1+1)EA solving the 10-, 20-,...,90-D Knapsack problems($\alpha = 0.5$).

focuses on local exploitation. Then, to jump from it to the global optimal solution is a difficult task because the transition probability is $\left(\frac{1}{n}\right)^{\alpha n}$. As a consequence, the upper bound of $e^{[t]}$ is dominated by the first item, which is of the order $\left[1 - \left(\frac{1}{n}\right)^{\alpha n}\right]^t$.

Generally, the expected approximation error converges very slowly to 0 as $t \to \infty$. However, we can excavate more information from the results of error analysis. For instance, consider a scalable case that α varies with n such that $\alpha n = k$, $1 \le k \le n$. Then, the asymptotic expected approximation error would be $\mathcal{O}(n)$ while the iteration budget is $t = an^k$. Furthermore, for given iteration budget $t = an^l$ with $l > k$, the asymptotic expected approximation error is $\mathcal{O}(nc^{n^{l-k}})$, $0 < c < 1$.

7. Conclusions and Discussions

In order to bridge the gap between theories and applications of RSH, this paper is dedicated to analyze elitist RSH by estimating the expected approximation error. According to the distribution of non-zero elements in the transition matrix of Markov chain, searching processes of elitist RSH are classified into three categories, and we propose a general framework for estimation of approximation error, named as the error analysis.

The error analysis can be applied to an RSH that is modeled by an upper triangular transition matrix. By computing the t-th power of the transition probability matrix, we can obtain general results on expected approximation error regarding any iteration t. Tricks of error analysis are definition of statues, diagonalization of upper triangular matrices and multiplication of block matrices. With help of these mathematical techniques, the error analysis can be applied easily to elitist RHS for uni- and multi-modal problems, which demonstrates the universality of error analysis. Meanwhile, the obtained results are concrete expressions of approximation error, which is much more precise than the asymptotic results of fixed-budget analysis.

Analysis of population-based EAs in the framework of error analysis is feasible if one can address how the transition probability is influenced by population size. For the (1+λ) EA that generates multiple offsprings by one parent, it is feasible to estimate improvement of transition probability, and the challenge lies in computation of the t-th power of transition matrix. However, to analyze (N+N)EA we must overcome the difficulties in both estimation of transition probability and computation of the t-th power. There are some other open questions in error analysis, including construction of Markov chain model, design of auxiliary searches, and computation of combinatorics, etc. Moreover, this study is based on the precondition that the transition matrix is diagonalizable. Thus, our future work would focus on analysis of RSHs whose transition matrix are not diagonalizable.

Appendix

Appendix A. Computation of $\mathbf{e}'\mathbf{p}_j$ and $\mathbf{q'}_j\mathbf{p}^{[0]}$ in Proof of Theorem 4

Denote $\mathbf{p}_j = (p_{1,j}, \ldots, p_{n,j})'$, $\mathbf{q}_j = (q_{1,j}, \ldots, q_{n,j})'$. By Lemma 4 we get the following results on $p_{i,j}$.

1. If $i > j$, $p_{i,j} = 0$;
2. if $i = j$, $p_{i,j} = 1$;
3. if $i < j$, equation (26) implies that

$$p_{i,j} = \prod_{k=i}^{j-1} \frac{r_{k,k+1}}{r_{j,j} - r_{k,k}} = \prod_{k=i}^{j-1} \frac{\frac{k+1}{n}}{\frac{k}{n} - \frac{j}{n}} = (-1)^{j-i} C_j^i, \quad , i = 1, \ldots, j-1.$$

Meanwhile, we can get the values of $q_{i,j}$:

1. If $i < j$, $q_{i,j} = 0$;
2. if $i = j$, $q_{i,j} = 1$;
3. if $i > j$, equation (26) implies that

$$q_{i,j} = \prod_{k=j+1}^{i} \frac{r_{k-1,k}}{r_{j,j} - r_{k,k}} = \prod_{k=j+1}^{i} \frac{\frac{k}{n}}{\frac{k}{n} - \frac{j}{n}} = C_i^j, \quad , i = j+1, \ldots, n.$$

In summary,

$$\mathbf{p}_j = \left((-1)^{j-1} C_j^1, (-1)^{j-2} C_j^2, \ldots, (-1)^1 C_j^{j-1}, 1, 0, \ldots, 0 \right)', \tag{A.1}$$

$$\mathbf{q}_j' = \left(0, \ldots, 0, 1, C_{j+1}^j, C_{j+2}^j, \ldots, C_n^j \right), \tag{A.2}$$

$$j = 1, \ldots, n.$$

Combing (24) and (A.1) we know

$$\mathbf{e}'\mathbf{p}_j = \sum_{i=1}^{j} i(-1)^{j-i} C_j^i = \sum_{i=1}^{j} (-1)^{j-i} C_j^1 C_{j-1}^{i-1} = \begin{cases} 1, & \text{if } j = 1; \\ 0, & \text{if } 2 \le j \le n. \end{cases} \tag{A.3}$$

Moreover, (25) and (A.2) imply that

$$\mathbf{q}_1'\mathbf{p}^{[0]} = \sum_{i=1}^{n} C_i^1 \frac{C_n^i}{2^n} = \frac{1}{2^n} \sum_{i=1}^{n} C_n^1 C_{n-1}^{i-1} = \frac{n}{2}. \tag{A.4}$$

Appendix B. Computation of $\mathbf{e}'\mathbf{p}_j$ and $\mathbf{q}_j'\mathbf{p}^{[0]}$ in Proof of Theorem 8

Denote $\mathbf{p}_j = (p_{1,j}, \ldots, p_{n,j})'$, $\mathbf{q}_j = (q_{1,j}, \ldots, q_{n,j})'$. Applying Lemma 4 we get the following results.

1. If $i > j$, $p_{i,j} = 0$;
2. if $i = j$, $p_{i,j} = 1$;

3. if $i < j$, equation (35) implies that

$$p_{i,j} = \prod_{k=i}^{j-1} \frac{r_{k,k+1}}{r_{j,j} - r_{k,k}}$$
$$= \begin{cases} \prod_{k=i}^{j-1} \frac{\frac{k}{n}}{\frac{k-1}{n} - \frac{j-1}{n}} = (-1)^{j-i} C_{j-1}^{i-1}, & i = 1, \ldots, j-1, j = 1, \ldots, n-1; \\ \prod_{k=i}^{j-1} \frac{\frac{k}{n}}{0 - (1 - \frac{k-1}{n})} = (-1)^{j-i} C_{j-1}^{i-1} \frac{1}{n+1-i}, & i = 1, \ldots, j-1, j = n. \end{cases}$$

Meanwhile, values of $q_{i,j}$ are as follows.

1. If $i < j$, $q_{i,j} = 0$;
2. if $i = j$, $q_{i,j} = 1$;
3. if $i > j$, equation (35) implies that

$$q_{i,j} = \prod_{k=j+1}^{i} \frac{r_{k-1,k}}{r_{j,j} - r_{k,k}}$$
$$= \begin{cases} \prod_{k=j+1}^{i} \frac{\frac{k-1}{n}}{\frac{k-1}{n} - \frac{j-1}{n}} = C_{i-1}^{j-1}, & i = j+1, \ldots, n-1; \\ \prod_{k=j+1}^{i-1} \frac{\frac{k-1}{n}}{\frac{k-1}{n} - \frac{j-1}{n}} \frac{n-1}{n-j+1} = C_{i-1}^{j-1} \frac{n-j}{n-j+1}, & i = n. \end{cases}$$

That is,

$$\mathbf{p}_j = \begin{cases} \left((-1)^{j-1} C_{j-1}^{0}, (-1)^{j-2} C_{j-1}^{1}, \ldots, -C_{j-1}^{j-2}, 1, 0, \ldots, 0\right)^T, & j < n \\ \left((-1)^{n-1} C_{n-1}^{1-1}/C_{n+1-1}^{1}, (-1)^{n-2} C_{n-1}^{2-1}/C_{n+1-2}^{1}, \ldots, -C_{n-1}^{(n-1)-1}/C_{n+1-(n-1)}^{1}, 1\right)^T, & j = n. \end{cases}$$

$$\mathbf{q}'_j = \left(0, \ldots, 0, 1, C_j^{j-1}, C_{j+1}^{j-1}, \ldots, C_{n-2}^{j-1}, C_{n-1}^{j-1} \frac{n-j}{n+1-j}\right), \quad j = 1, \ldots, n.$$

By equation (33), we know

$$\mathbf{ep}_j = \sum_{i=1}^{j} (-1)^{j-i} C_{j-1}^{i-1} i = \frac{d}{dx} \left(\sum_{i=1}^{j} (-1)^{j-i} C_{j-1}^{i-1} x^i \right) \Bigg|_{x=1}$$
$$= \frac{d}{dx} \left(x \sum_{k=0}^{j-1} (-1)^{j-1-k} C_{j-1}^{k} x^k \right) \Bigg|_{x=1} = \frac{d}{dx} \left(x(x-1)^{j-1} \right) \Big|_{x=1}$$
$$= \begin{cases} 1, & j = 1, 2, \\ 0, & j = 3, \ldots, n-1, \end{cases} \tag{B.1}$$

$$\mathbf{ep}_n = \sum_{i=1}^{n} (-1)^{n-i} i C_{n-1}^{i-1}/(n+1-i) = (n+1) \sum_{i=1}^{n} (-1)^{n-i} C_{n-1}^{i-1}/(n+1-i) - \sum_{i=1}^{n} (-1)^{n-i} C_{n-1}^{i-1}$$
$$= (n+1) \left[\left(\sum_{i=1}^{n} (-1)^{n-i} C_{n-1}^{i-1} \int_0^x x^{n-i} dx \right) \right]_{x=1} = (n+1) \left[\int_0^x x^{n-1} \left(\sum_{k=0}^{n-1} (-1)^{n-1-k} C_{n-1}^{k} x^{-k} \right) dx \right]_{x=1}$$
$$= (n+1) \left[\int_0^x (1-x)^{n-1} dx \right]_{x=1} = \frac{n+1}{n}. \tag{B.2}$$

By equation (34), we know

$$\mathbf{q}_j'\mathbf{p}^{[0]} = \sum_{i=j}^{n-1} C_{i-1}^{j-1}\frac{C_n^{i-1}}{2^n} + C_{n-1}^{j-1}\frac{n-j}{n+1-j}\frac{C_n^{n-1}}{2^n}$$
$$= \frac{1}{2^n}\left(\sum_{i=j}^{n} C_n^{j-1}C_{n-j+1}^{i-j} - C_n^{j-1}\right) = \frac{C_n^{j-1}}{2^n}\left(2^{n-j+1}-2\right). \quad \text{(B.3)}$$

Appendix C. Estimation of $\check{\mathbf{e}}'\check{\mathbf{p}}_j$, $\check{\mathbf{q}}_j'\check{\mathbf{p}}^{[0]}$ and $\check{\mathbf{q}}_j'\check{\mathbf{r}}^{[1]}$ in Proof of Theorem 9

Similar to computation in Appendix B, by (40) we know

$$\check{\mathbf{p}}_j = \left((-1)^{j-1}C_{j-1}^0, (-1)^{j-2}C_{j-1}^1, \dots, -C_{j-1}^{j-2}, 1, 0, \dots, 0\right)', j = 1, \dots, n-1$$
$$\check{\mathbf{q}}_j' = \left(0, \dots, 0, 1, C_j^{j-1}, C_{j+1}^{j-1}, \dots, C_{n-2}^{j-1}\right), \quad j = 1, \dots, n-1.$$

Then, equation (43) implies that

$$\check{\mathbf{e}}\check{\mathbf{p}}_j = \sum_{i=1}^{j}(-1)^{j-i}C_{j-1}^{i-1}i = \begin{cases} 1, & j = 1, 2, \\ 0, & j = 3, \dots, n-1. \end{cases} \quad \text{(C.1)}$$

Moreover, equation (44) implies

$$\check{\mathbf{q}}_j'\check{\mathbf{p}}^{[0]} = \sum_{i=j}^{n-1} C_{i-1}^{j-1}\frac{C_n^{i-1}}{2^n} = \frac{C_n^{j-1}}{2^n}(2^{n-j+1} - (n-j+2)), \quad \text{(C.2)}$$

and by equation (45) we know

$$\check{\mathbf{q}}_j'\check{\mathbf{r}}^{[1]} = \sum_{i=j}^{n-1} C_{i-1}^{j-1}r_{i,n} \le C_{n-1}^{j-1}(1 - r_{0,n} - r_{n,n})$$
$$= C_{n-1}^{j-1}\left[1 - \frac{1}{n}\left(1-\frac{1}{n}\right)^{n-1} - \left(\left(1-\frac{1}{n}\right)^n + C_{n-1}^1\left(\frac{1}{n}\right)^2\left(1-\frac{1}{n}\right)^{n-2}\right)\right]$$
$$= C_{n-1}^{j-1}\left(1 - \frac{n+1}{n}\left(1-\frac{1}{n}\right)^{n-1}\right). \quad \text{(C.3)}$$

Appendix D. Computation of $\check{\mathbf{e}}'\check{\mathbf{p}}_j$ and $\check{\mathbf{q}}_j'\check{\mathbf{p}}^{[0]}$ in Proof of Theorem 10

Derivation of $\check{\mathbf{e}}\check{\mathbf{p}}_j$ and $\check{\mathbf{q}}_j\check{\mathbf{p}}^{[0]}$ is similar to that in Appendix B. By (57) we know

$$\check{\mathbf{p}}_j = \begin{cases} \left((-1)^{j-1}C_{j-1}^0, (-1)^{j-2}C_{j-1}^1, \dots, -C_{j-1}^{j-2}, 1, 0, \dots, 0\right)^T, & j < \alpha n \\ \left((-1)^{j-1}C_{j-1}^{1-1}/C_{j+1-1}^1, (-1)^{j-2}C_{j-1}^{2-1}/C_{j+1-2}^1, \dots, -C_{j-1}^{(j-1)-1}/C_{j+1-(j-1)}^1, 1\right)^T, & j = \alpha n, \end{cases}$$
$$\check{\mathbf{q}}_j' = \left(0, \dots, 0, 1, C_j^{j-1}, C_{j+1}^{j-1}, \dots, C_{\alpha n-2}^{j-1}, C_{\alpha n-1}^{j-1}\frac{\alpha n - j}{\alpha n + 1 - j}\right), \quad j = 1, \dots, \alpha n.$$

Then, equation (62) implies that

$$\check{\mathbf{e}}\check{\mathbf{p}}_j = \sum_{i=1}^{j}(-1)^{j-i}C_{j-1}^{i-1}(n-(\alpha n-i)) = (1-\alpha)n\sum_{i=1}^{j}(-1)^{j-i}C_{j-1}^{i-1} + \sum_{i=1}^{j}(-1)^{j-i}C_{j-1}^{i-1}i$$

$$= \begin{cases} n-\alpha n+1, & j=1, \\ 1, & j=2, \\ 0, & j=3,\dots,\alpha n-1; \end{cases} \tag{D.1}$$

$$\check{\mathbf{e}}\check{\mathbf{p}}_{\alpha n} = \sum_{i=1}^{\alpha n}(-1)^{\alpha n-i}(n-(\alpha n-i))C_{\alpha n-1}^{i-1}/(\alpha n+1-i)$$

$$= (n+1)\sum_{i=1}^{\alpha n}(-1)^{\alpha n-i}C_{\alpha n-1}^{i-1}/(\alpha n+1-i) - \sum_{i=1}^{\alpha n}(-1)^{\alpha n-i}C_{\alpha n-1}^{i-1} = \frac{n+1}{\alpha n}. \tag{D.2}$$

By equation (63), we know that

$$\check{\mathbf{q}}_j'\check{\mathbf{p}}^{[0]} = \sum_{i=j}^{\alpha n-1} C_{i-1}^{j-1}\frac{C_{\alpha n-1}^{i-1}}{2^{\alpha n-1}} + C_{\alpha n-1}^{j-1}\frac{\alpha n-j}{\alpha n+1-j}\left(\frac{1}{2^{\alpha n}}-\frac{1}{2^n}\right)$$

$$= C_{\alpha n-1}^{j-1}\left[\frac{1}{2^{j-1}}-\frac{1}{2^{\alpha n-1}}+\frac{\alpha n-j}{\alpha n+1-j}\left(\frac{1}{2^{\alpha n}}-\frac{1}{2^n}\right)\right]. \quad j=1,\dots,\alpha n-1, \tag{D.3}$$

$$\check{\mathbf{q}}_{\alpha n}'\check{\mathbf{p}}^{[0]} = \frac{1}{2^{\alpha n}}-\frac{1}{2^n}. \tag{D.4}$$

Moreover, equation (64) implies that

$$\left(1-\frac{1}{n}\right)C_{n-2}^{j-1}\frac{\alpha n-j}{\alpha n+1-j} \le \check{\mathbf{q}}_j'\check{\mathbf{r}}^{[1]} \le \left(1-\frac{1}{n}\right)C_{n-1}^{j-1}, \quad j<\alpha n, \tag{D.5}$$

$$\check{\mathbf{q}}_{\alpha n}'\check{\mathbf{r}}^{[1]} = 1-\alpha. \tag{D.6}$$

Appendix E. Computation of $\mathbf{e}_R'\mathbf{p}_j$, $\mathbf{q}_j'\mathbf{p}_R^{[0]}$ and $\mathbf{q}_j'\mathbf{r}^{[1]}$ in Proof of Theorem 11

Similar computation in Appendix A, by (84) and (85) we know

$$\mathbf{p}_j = \left((-1)^{j-1}C_{j-1}^0, (-1)^{j-2}C_{j-1}^1, \dots, -C_{j-1}^{j-2}, 1, 0, \dots, 0\right)',$$

$$\mathbf{q}_j' = \left(0,\dots,0,1,C_j^{j-1},C_{j+1}^{j-1},\dots,C_{\alpha n-2}^{j-1}\right), \quad j=1,\dots,\alpha n-1.$$

Combining them with (76), (77) and (78), we know

$$\mathbf{e}_R'\mathbf{p}_j = \sum_{i=1}^{j}(-1)^{j-i}C_{j-1}^{i-1}(n-(\alpha n-i)) = \begin{cases} n-\alpha n+1, & j=1; \\ 1, & j=2; \\ 0, & j=3,\dots,\alpha n-1, \end{cases} \tag{E.1}$$

$$\mathbf{q}_j'\mathbf{p}_R^{[0]} = \sum_{k=j}^{\alpha n-1} C_{k-1}^{j-1}C_{\alpha n-1}^{k-1}\frac{1}{2^{\alpha n-1}} = C_{\alpha n-1}^{j-1}\left[\frac{1}{2^{j-1}}-\frac{1}{2^{\alpha n-1}}\right], \tag{E.2}$$

$$\mathbf{q}_j'\mathbf{r}^{[1]} = \sum_{i=j}^{\alpha n-1} C_{i-1}^{j-1}r_{i,\alpha n} \le \sum_{i=1}^{\alpha n-1} C_{i-1}^{j-1}r_{i,\alpha n} \le C_{\alpha n-1}^{j-1}(1-r_{0,\alpha n}-r_{\alpha n,\alpha n})$$

$$= C_{\alpha n-1}^{j-1}\left[1-\left(1-\frac{1}{n}\right)^{\alpha n-1}\right]. \tag{E.3}$$

References

[1] S. Yang, K. Li, W. Li, W. Chen, Y. Chen, Dynamic fitness landscape analysis on differential evolution algorithm, in: International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2016), Springer, 2016, pp. 179–184.

[2] S. Li, W. Gong, X. Yan, C. Hu, D. Bai, L. Wang, L. Gao, Parameter extraction of photovoltaic models using an improved teaching-learning-based optimization, Energy Conversion and Management 186 (2019) 293–305.

[3] W. Gong, Y. Wang, Z. Cai, L. Wang, Finding multiple roots of nonlinear equation systems via a repulsion-based adaptive differential evolution, IEEE Transactions on Systems, Man, and Cybernetics: Systems 50 (4) (2020) 1499–1513.

[4] J. He, X. Yao, Drift analysis and average time complexity of evolutionary algorithms, Artificial intelligence 127 (1) (2001) 57–85.

[5] S. Droste, T. Jansen, I. Wegener, On the analysis of the (1+ 1) evolutionary algorithm, Theoretical Computer Science 276 (1-2) (2002) 51–81.

[6] J. He, X. Yao, Towards an analytic framework for analysing the computation time of evolutionary algorithms, Artificial Intelligence 145 (1-2) (2003) 59–97.

[7] B. Doerr, D. Johannsen, C. Winzen, Multiplicative drift analysis, Algorithmica 64 (4) (2012) 673–697.

[8] Y. Yu, C. Qian, Z.-H. Zhou, Switch analysis for running time analysis of evolutionary algorithms, IEEE Transactions on Evolutionary Computation 19 (6) (2014) 777–792.

[9] H. Huang, J. Su, Y. Zhang, Z. Hao, An experimental method to estimate running time of evolutionary algorithms for continuous optimization, IEEE Transactions on Evolutionary Computation 24 (2) (2020) 275–289.

[10] P. Oliveto, J. He, X. Yao, Time complexity of evolutionary algorithms for combinatorial optimization: A decade of results, International Journal of Automation and Computing 4 (3) (2007) 281–293.

[11] Y. Zhou, Runtime analysis of an ant colony optimization algorithm for tsp instances, IEEE Transactions on Evolutionary Computation 13 (5) (2009) 1083–1092.

[12] H. Huang, C.-G. Wu, Z.-F. Hao, A pheromone-rate-based analysis on the convergence time of aco algorithm, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 39 (4) (2009) 910–923.

[13] T. Friedrich, J. He, N. Hebbinghaus, F. Neumann, C. Witt, Approximating covering problems by randomized search heuristics using multi-objective models, Evolutionary Computation 18 (4) (2010) 617–633.

[14] A. M. Sutton, F. Neumann, S. Nallaperuma, Parameterized runtime analyses of evolutionary algorithms for the planar euclidean traveling salesperson problem, Evolutionary Computation 22 (4) (2014) 595–628.

[15] B. Doerr, F. Neumann, A. M. Sutton, Time complexity analysis of evolutionary algorithms on random satisfiable k-cnf formulas, Algorithmica 78 (2) (2017) 561–586.

[16] C. Qian, J.-C. Shi, K. Tang, Z.-H. Zhou, Constrained monotone k-submodular function maximization using multiobjective evolutionary algorithms with theoretical guarantee, IEEE Transactions on Evolutionary Computation 22 (4) (2017) 595–608.

[17] Y. Chen, X. Zou, J. He, Drift conditions for estimating the first hitting times of evolutionary algorithms, International Journal of Computer Mathematics 88 (1) (2011) 37–50.

[18] Y. Yu, X. Yao, Z.-H. Zhou, On the approximation ability of evolutionary optimization with application to minimum set cover, Artificial Intelligence (180-181) (2012) 20–33.

[19] X. Lai, Y. Zhou, J. He, J. Zhang, Performance analysis of evolutionary algorithms for the minimum label spanning tree problem, IEEE Transactions on Evolutionary Computation 18 (6) (2014) 860–872.

[20] H. Huang, W. Xu, Y. Zhang, Z. Lin, Z. Hao, Runtime analysis for continuous (1+ 1) evolutionary algorithm based on average gain model, Scientia Sinica Informationis 44 (6) (2014) 811–824.

[21] Y. Zhou, X. Lai, K. Li, Approximation and parameterized runtime analysis of evolutionary algorithms for the maximum cut problem, IEEE transactions on cybernetics 45 (8) (2015) 1491–1498.

[22] X. Xia, Y. Zhou, X. Lai, On the analysis of the (1+ 1) evolutionary algorithm for the maximum leaf spanning tree problem, International Journal of Computer Mathematics 92 (10) (2015) 2023–2035.

[23] X. Xia, Y. Zhou, Approximation performance of the (1+ 1) evolutionary algorithm for the minimum degree spanning tree problem, in: Bio-Inspired Computing-Theories and Applications (BIC-TA 2015), Springer, 2015, pp. 505–512.

[24] X. Peng, Y. Zhou, G. Xu, Approximation performance of ant colony optimization for the tsp (1, 2) problem, International Journal of Computer Mathematics 93 (10) (2016) 1683–1694.

[25] Y. Zhang, H. Huang, Z. Hao, G. Hu, First hitting time analysis of continuous evolutionary algorithms based on average gain, Cluster Computing 19 (3) (2016) 1323–1332.

[26] T. Jansen, C. Zarges, Performance analysis of randomised search heuristics operating with a fixed budget, Theoretical Computer Science 545 (2014) 39–58.

[27] T. Jansen, C. Zarges, Reevaluating immune-inspired hypermutations using the fixed budget perspective, IEEE Transactions on Evolutionary Computation 18 (5) (2014) 674–688.

[28] S. Nallaperuma, F. Neumann, D. Sudholt, Expected fitness gains of randomized search heuristics for the traveling salesperson problem, Evolutionary computation 25 (4) (2017) 673–705.

[29] G. Rudolph, Convergence rates of evolutionary algorithms for a class of convex objective functions, Control and Cybernetics 26 (1997) 375–390.

[30] J. He, G. Lin, Average convergence rate of evolutionary algorithms, IEEE Transactions on Evolutionary Computation 20 (2) (2016) 316–321.

[31] Y. Chen, J. He, Average convergence rate of evolutionary algorithms ii: Continuous optimisation, in: International Symposium on Intelligence Computation and Applications, Springer, 2019, pp. 31–45.

[32] Y. Chen, J. He, Average convergence rate of evolutionary algorithms in continuous optimization, Information Sciencesdoi: https://doi.org/10.1016/j.ins.2020.12.076.
[33] J. He, T. Jansen, C. Zarges, Unlimited budget analysis, in: Proceedings of the Genetic and Evolutionary Computation Conference Companion, ACM, 2019, pp. 427–428.
[34] J. He, T. Jansen, C. Zarges, Unlimited budget analysis of randomised search heuristics, arXiv preprint arXiv:1909.03342.
[35] J. He, An analytic expression of relative approximation error for a class of evolutionary algorithms, in: Proceedings of 2016 IEEE Congress on Evolutionary Computation (CEC 2016), 2016, pp. 4366–4373.
[36] J. He, Y. Chen, Y. Zhou, A theoretical framework of approximation error analysis of evolutionary algorithms, arXiv preprint arXiv:1810.11532.
[37] C. Wang, Y. Chen, J. He, C. Xie, Estimating approximation errors of elitist evolutionary algorithms, in: International Conference on Bio-Inspired Computing: Theories and Applications, Springer, 2019, pp. 325–340.
[38] D. C. Lay, Linear algebra and its applications, Addison Wesley Boston, 2003.
[39] C. Witt, Fitness levels with tail bounds for the analysis of randomized search heuristics, Information Processing Letters 114 (1) (2014) 38 – 41.
[40] B. Doerr, M. Fouz, C. Witt, Quasirandom evolutionary algorithms, in: Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation, GECCO '10, Association for Computing Machinery, New York, NY, USA, 2010, p. 14571464.
[41] J. He, T. Chen, X. Yao, On the easiest and hardest fitness functions, IEEE Transactions on Evolutionary Computation 19 (2) (2015) 295–305.
[42] Çinlar Erhan, Probability and Stochasitics, Springer, 2011.
[43] J. He, B. Mitavskiy, Y. Zhou, A theoretical assessment of solution quality in evolutionary algorithms for the knapsack problem, in: 2014 IEEE Congress on Evolutionary Computation (CEC 2014), IEEE, 2014, pp. 141–148.

Cong Wang: Theoretical analysis of case studies **Yu Chen**: Design of research, draft preparation. **Ju He**: Supervision. **Chengwang Xie**: Writing- Reviewing and Editing,

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: