

---

# Expected Fitness Gains of Randomized Search Heuristics for the Traveling Salesperson Problem

**Samadhi Nallaperuma**

samadhi.nallaperuma@adelaide.edu.au

Optimisation and Logistics, School of Computer Science, The University of Adelaide, Adelaide, SA 5005, Australia

**Frank Neumann**

frank.neumann@adelaide.edu.au

Optimisation and Logistics, School of Computer Science, The University of Adelaide, Adelaide, SA 5005, Australia

**Dirk Sudholt**

d.sudholt@sheffield.ac.uk

Department of Computer Science, The University of Sheffield, Sheffield, S10 2TN, United Kingdom

---

## Abstract

Randomized search heuristics are frequently applied to NP-hard combinatorial optimization problems. The runtime analysis of randomized search heuristics has contributed tremendously to their theoretical understanding. Recently, randomized search heuristics have been examined regarding their achievable progress within a fixed time budget. We follow this approach and present a fixed budget analysis for an NP-hard combinatorial optimization problem. We consider the well-known Traveling Salesperson problem (TSP) and analyze the fitness increase that randomized search heuristics are able to achieve within a given fixed time budget. In particular, we analyze Manhattan and Euclidean TSP instances and Randomized Local Search (RLS), (1+1) EA and (1+ $\lambda$ ) EA algorithms for the TSP in a smoothed complexity setting and derive the lower bounds of the expected fitness gain for a specified number of generations.

## Keywords

Traveling Salesperson Problem, fitness gain, fixed-budget analysis, runtime analysis theory.

## 1 Introduction

Randomized search heuristics (RSH) such as randomized local search, evolutionary algorithms and ant colony optimization have become very popular in recent years to solve hard combinatorial optimization problems such as the Traveling salesperson problem [2, 10, 20, 24, 29].

Regarding RSH as classical randomized algorithms [21], a lot of progress has been made in recent years on their theoretical understanding [1, 11]. Initially, most of the studies were focused on simple example functions. Gradually, the analysis on combinatorial optimization problems was also established. We refer the textbook by Neumann and Witt [23] and the survey article by Oliveto et al. [25] for a comprehensive presentation on the runtime analysis of randomized search heuristics for problems from combinatorial optimization.

All these studies on analyzing the time complexity were based on a single perspective, the expected optimization time. There were slight variations on this considering the number of generations or fitness evaluations. Jansen and Zarges [12] and Zhou et al. [31] pointed out that there is a gap between the empirical results and the theoretical results obtained on the optimization time. Theoretical research most often yields asymptotic results on finding the global optimum while practitioners concern more about achieving some good result within a reasonable time budget. Furthermore, it is beneficial to know how much progress an algorithm can make given some additional time budget. Experimental studies on this topic have been carried out in the domain of algorithm engineering [27] and the term fixed budget runtime analysis has been introduced by Jansen and Zarges [13, 15].

So far, fixed budget analysis has been conducted for very simple test functions, on which the considered randomized search heuristics such as randomized local search and the (1+1) EA follow a typical search trajectory [6, 15] with high probability. This implies that on these functions the development of the best fitness over time forms an almost deterministic curve that describes the algorithm's typical behavior. Given such a strong relation between fitness and time, and the availability of upper and lower tail bounds, it is then possible to derive tight upper and lower bounds on the expected fitness increase over any given period of time. More recent studies in this line of research include comparisons of fixed budget results obtained by evolutionary algorithms to other RSH such as artificial immune systems [5, 16], fixed budget analysis in dynamic optimization [14] and introduction of more general methods of analysis [18].

The goal of obtaining tight upper and lower bounds is only feasible for functions where randomized search heuristics show a typical search trajectory and tail bounds are available to bound deviations from this trajectory. This usually does not apply to hard combinatorial problems like the Traveling Salesperson Problem, and currently no fixed-budget analysis is available for such problems. We argue that for these problems fixed budget results can be obtained by relaxing the above goal towards only considering *lower bounds* on the expected fitness gain. Lower bounds can be determined based on the expected minimum improvement made in a generation. In this manner there is no requirement for obtaining tail bounds, which drastically widens the scope of problems that can be tackled with this approach. Even though lower bounds on the expected fitness gain may not be tight, they provide proven guarantees on the progress made by a RSH. The aim of this approach is to establish guarantees on the expected fitness gain for various kinds of RSH, hence providing guidance for choosing, designing, and tuning RSH such that they find high-fitness solutions in a short time.

This study provides a starting point for fixed-budget analysis of randomized search heuristics for combinatorial optimization problems. We consider randomized local search (RLS) and (1+1) Evolutionary Algorithm ((1+1) EA) on the famous Traveling Salesperson Problem (TSP). The dynamics of evolutionary algorithms on the TSP have been studied empirically [2, 24], but the authors are not aware of any rigorous theoretical works studying the fitness of evolutionary algorithms on this problem.

More specifically, here we analyze Manhattan and Euclidean TSP instances in the setting of smoothed complexity [28]. Smoothed analysis provides a generic framework to analyze algorithms like 2-Opt for TSP with the capability to interpolate between average and worst case analysis. This analysis was first proposed by Spielman and Teng [28] focusing on the simplex algorithm to explain the discrepancy between its exponential worst case runtime and the fast performance in practice. The probabilistic model proposed by Englert et al. [8] is a reminiscence of the original smoothed analysis

model. Later, these results were refined by Manthey and Veestra [19]. Here, we will adhere to the initial analysis by Englert et al. [8] as our major focus is on transferring these results to a fixed budget analysis of RSH. Then we further generalize our results for  $(1+\lambda)$  RLS and  $(1+\lambda)$  EA as well.

We build on the analysis of Englert et al. [9] for 2-Opt which allows to get bounds on the expected progress of a 2-Opt operation in the smoothed setting. We consider 2-dimensional metric TSP instances. First, we obtain fixed budget results based on the minimum improvement that RLS and  $(1+1)$  EA can make in one generation. We further improve these results, following [9], by analyzing a sequence of consecutive 2-Opt steps together to identify linked pairs. Interestingly, considering only single improving steps gives a constant lower bound on the progress achievable in each of the  $t$  generations whereas the analysis of a sequence of consecutive 2-Opt steps gives a larger expected progress per step if  $t$  is large. The analysis follows in a similar way for the population based algorithms also. Additionally, the analysis of population-based algorithms shows that there is a fitness gain improved by at least  $\Omega(\min(\lambda, n^2))$  times of the lower bound of the fitness gain for RLS or  $(1+1)$  EA for both single step and consecutive step cases.

This article extends its conference version [22] by the investigations of the expected fitness gains for population-based algorithms. Furthermore, it extends the conference article by the analysis regarding the achieved approximation ratio given a fixed time budget and experimental analysis on sample algorithm runs.

The organization of the paper is as follows. Section 2 describes problem context and the considered algorithms. Section 3 and Section 4 contain the analysis for Manhattan and Euclidean instances respectively. Section 5 extends the results for population-based algorithms. Section 6 presents experimental results on sample algorithm runs. Finally, Section 7 concludes with highlights and future directions.

## 2 Preliminaries

Generally, an instance of the TSP consists of a set  $V = \{v_1, \dots, v_n\}$  of  $n$  vertices (depending on the context, synonymously referred to as points) and a distance function  $d : V \times V \rightarrow \mathbb{R}_0^+$  that associates a distance value with each pair  $v_i, v_j$ . The goal is to find a tour of minimal length that visits each vertex exactly once and returns to the initial vertex. That is to compute a permutation  $\pi$  minimizing  $\sum_{i=1}^{n-1} d(v_{\pi(i)}, v_{\pi(i+1)}) + d(v_{\pi(n)}, v_{\pi(1)})$ .

A TSP instance is considered to be metric if its distance function is in a metric space. A metric space satisfies reflexivity, symmetry and triangle inequality conditions. A pair  $(V, d)$  of the set  $V$  and a function  $d : V \times V \rightarrow \mathbb{R}_0^+$  is called a metric space if for all  $v_i, v_j, v_k \in V$  the following properties are satisfied:

- $d(v_i, v_j) = 0$  if and only if  $v_i = v_j$ ,
- $d(v_i, v_j) = d(v_j, v_i)$
- $d(v_i, v_k) \leq d(v_i, v_j) + d(v_j, v_k)$ .

We consider  $n$  cities given by points  $v_i = (x_i, y_i)$ ,  $1 \leq i \leq n$ , in the plane. For a distance metric  $L_p$  the distance of two points  $v_i = (x_i, y_i)$  and  $v_j = (x_j, y_j)$  is

$$d_p(v_i, v_j) = (|x_i - x_j|^p + |y_i - y_j|^p)^{1/p}.$$

We study the Euclidean TSP and the Manhattan TSP which are two prominent cases of the metric TSP having the distance metric Euclidean ( $L_2$ ) and Manhattan ( $L_1$ ) respectively.

## 2.1 RLS and A Simple Evolutionary Algorithm

---

### Algorithm 1 RLS

---

```

 $x \leftarrow$  a random permutation of  $n$  cities.
repeat
   $y := x$ .
   $y \leftarrow$  apply a 2-Opt step chosen uniformly at random to  $y$ .
  if  $f(y) \leq f(x)$  then
     $x \leftarrow y$ .
  end if
until forever

```

---



---

### Algorithm 2 (1+1) EA

---

```

 $x \leftarrow$  a random permutation of  $n$  cities.
repeat
  draw  $s$  from a Poisson distribution with expectation 1.
   $y := x$ .
  for  $s + 1$  times do
     $y \leftarrow$  apply a 2-Opt step chosen uniformly at random to  $y$ .
  end for
  if  $f(y) \leq f(x)$  then
     $x \leftarrow y$ .
  end if
until forever

```

---

We consider simple randomized search heuristics and analyze them with respect to the progress that they make within a given time budget. Randomized Local Search (RLS) (Algorithm 1) and (1+1) EA (Algorithm 2) are two randomized search heuristics that have won a great popularity during recent years [7]. These algorithms work with a population size of one and produce one offspring in each generation. A basic mutation is given by the well-known 2-Opt operator. The usual effect of the 2-Opt is to delete the two edges  $\{v_{\pi(i-1)}, v_{\pi(i)}\}$  and  $\{v_{\pi(j)}, v_{\pi(j+1)}\}$  from the tour and reconnect the tour using edges  $\{v_{\pi(i-1)}, v_{\pi(j)}\}$  and  $\{v_{\pi(i)}, v_{\pi(j+1)}\}$  (see Figure 1).

RLS performs one mutation step in each generation to produce an offspring. In contrast, (1+1) EA chooses an integer variable  $s$  drawn from the Poisson distribution with expectation 1 in each mutation step and performs sequentially  $s + 1$  mutation operations. In case  $s + 1 = 1$ , we speak of a singular mutation, or a singular generation. (1+1) EA can simulate a mutation step in singular generations, that occurs with a probability of  $1/e$ ,  $e = \exp(1)$ . Moreover, it has a positive probability of generating a global optimum in every generation by executing the right number and sequence of mutation steps. Therefore, it is guaranteed to find a global optimum in finite time, though this time may be exponential in  $n$ . Note that in our algorithms we consider the notion of fitness with regard to the minimization of the tour-length. As evolutionary algorithms often maximize fitness, we use the term *fitness gain* to describe fitness improvements, that is, the decrease of the tour length.

We study these algorithms regarding the expected progress that they make within a given number of  $t$  generations. We consider the algorithms on random instances in

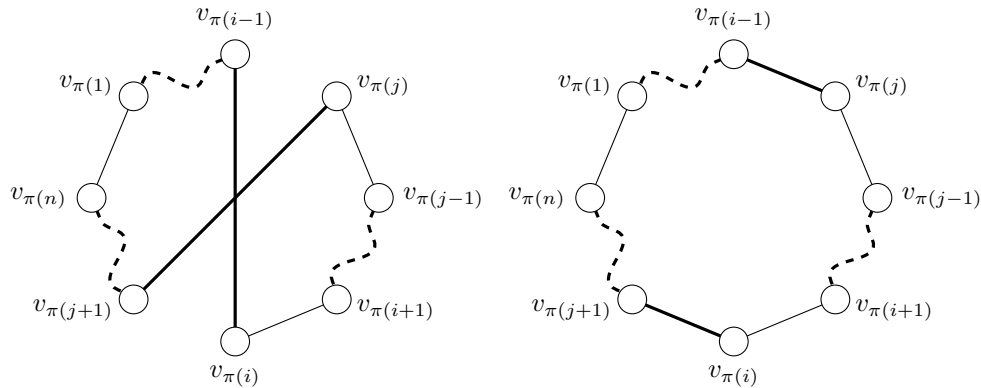


Figure 1: The effect of the 2-Opt operation on a TSP tour. Inverting a subsequence in the permutation representation corresponds to a 2-Opt move in which a pair of edges in the current tour is replaced by a pair of edges not in the tour.

the setting of smoothed analysis [9, 28].

In this model,  $n$  points are placed in a  $d$ -dimensional unit hypercube  $[0, 1]^d$  for  $d \geq 2$ . Each point  $v_i, i = 1, 2, \dots, n$ , is chosen independently according to its own probability density function  $f_i: [0, 1]^d \rightarrow [0, \phi]$  for some parameter  $\phi \geq 1$  defining the maximal density. For example, the uniform distribution has the probability density function

$$f(x) = \begin{cases} 1/(b-a) & \text{if } a \leq x \leq b \\ 0 & \text{otherwise.} \end{cases}$$

This means that for any given point within the interval  $[a, b]$  the only choice for the probability density is  $1/(b-a)$ . Accordingly, for the unit hypercube this would be 1. In fact this value is the maximal density  $\phi$  for the uniform distribution over the considered unit hypercube. To model worst-case instances, it is assumed that these densities are chosen by an adversary who is trying to create the most difficult random instances possible. By adjusting the parameter  $\phi$ , one can tune the power of this adversary and hence interpolate between worst and average case. The larger  $\phi$ , the more concentrated the probability mass can be, the better the adversary can approximate worst case instances by the distributions. On the other hand when  $\phi$  gets closer to 1 the probability mass gets less concentrated and thus the points are positioned more randomly. When  $\phi = 1$  the points are positioned uniformly at random in the unit hypercube resulting in average case analysis. The two types of instances are called  $\phi$ -perturbed Manhattan instances and  $\phi$ -perturbed Euclidean instances [9]. For our analysis we consider the  $n$  points in a 2 dimensional unit hypercube  $[0, 1]^2$ .

This model covers a smoothed analysis also with a slight modification. There the adversary determines the initial distribution of points and then a slight perturbation is applied to each position, adding a Gaussian random variable with small standard deviation  $\sigma$ . There  $\phi$  has to be set as  $1/(\sqrt{2\pi}\sigma)^d$  [9]<sup>1</sup>. This smoothed model is considered in the study by Manthey and Veestra [19].

Analyzing our algorithms in this setting, we may assume that any two different

<sup>1</sup>This follows from the fact that the density of a  $d$ -dimensional Gaussian with standard deviation  $\sigma$  is bounded by  $(2\sigma)^{-d}$

tours have different function values. Hence, both algorithms always accept strict improvements.

## 2.2 Minimum Improvement of a 2-Opt Step

We now summarize results by Englert et al. [9] on the minimal improvement of a 2-Opt step. Later on, these results will be used in our analysis of the randomized search heuristics. We denote the random variable  $\Delta$  that describes the fitness gain obtained in one generation and  $\Delta_t$  the fitness gain in  $t$  generations. We call a 2-Opt step *improving* if it decreases the tour length in TSP. A 2-Opt step is called *singular* if it is the only 2-Opt step executed in that generation. Based on the smallest improvement of any improving 2-Opt step we can find the expected improvement made in  $t$  generations of (1+1) EA. We set the interval  $(0, \varepsilon]$ , describing a range of a 2-Opt improvement for an  $\varepsilon > 0$ <sup>2</sup>. Let us first consider a fixed 2-Opt step in which the edges  $e_1$  and  $e_2$  are exchanged with the edges  $e_3$  and  $e_4$ . This 2-Opt step decreases the length of the tour by  $\Delta(e_1, e_2, e_3, e_4) = d(e_1) + d(e_2) - d(e_3) - d(e_4)$ .

Let  $\Delta_{\min}$  denote the smallest possible improvement made by any improving 2-Opt step:

$$\Delta_{\min} = \min\{\Delta \mid \Delta > 0\}.$$

Inspired by the original ideas of Kern [17], Chandra et al. [4] bounded the probability that this smallest improvement lies within the interval  $(0, \varepsilon]$  with a high probability for the uniform distribution.

We will make use of the following theorem by Englert et al. [9] which gives an upper bound on the probability that an improving 2-Opt step gives an improvement of at most  $\varepsilon$  for the Manhattan metric. Throughout this analysis we consider the two dimensional space.

**Theorem 1** (Manhattan metric [9, Theorem 7]). *For the Manhattan metric any  $\varepsilon > 0$ , it holds*

$$\Pr(\Delta_{\min} \leq \varepsilon) \leq 4!^2 \varepsilon n^4 \phi.$$

Based on this result we get a lower bound on the probability that the smallest improvement is greater than any given  $\varepsilon$  (see Theorem 5). It should be noted that here we consider the algorithms to accept non-strict improvements so that it may be possible to have  $\Delta = 0$ . However, this is not probable with the considered smoothed setting because a necessary condition for this event is that there exist two edges with the same length, which for two fixed edges has probability 0 (if  $\phi < \infty$ ). The union bound over all pairs of edges gives a probability upper bound of 0. Therefore, we almost surely have  $\Delta > 0$ .

Similar to the Manhattan instances, for the Euclidean instances also, the minimum improvement per a 2-Opt step is inspired by the original ideas of Kern [17]. Based on this, the expected runtime was proved polynomial for the uniform distribution by Chandra et al. [4]. This was later extended for a more generalized setting having any probability distribution by Englert et al. [9].

**Lemma 2** (Euclidean metric [9, Lemma 18]). *For the Euclidean metric and any  $\varepsilon > 0$ , it holds*

$$\Pr(\Delta_{\min} \leq \varepsilon) \leq 1536n^4 \cdot \varepsilon \cdot \log(1/\varepsilon) \cdot \phi^3.$$

<sup>2</sup> $\varepsilon > 0$  is in fact a variable which is used within the analysis. This does not appear in the fitness gain results. We choose a value for  $\varepsilon$  such that it satisfies a certain criteria. This value differs in each Theorem.

In case the considered algorithms reach a local optimum, we cannot guarantee a steady fitness gain. So instead we use the fact that local optima have a good approximation ratio. The approximation ratio for the worst local optimum with regard to 2-Opt was proven originally in Chandra et al. [4] for the uniform distribution. This was later generalized by Englert et al. [9] for any probability distribution with a given density function with maximal density  $\phi$ .

**Theorem 3** ([9, Theorem 4]). *Let  $p \in \mathbb{N} \cup \{\infty\}$ . For  $\phi$ -perturbed  $L_p$  instances the expected approximation ratio of the worst tour that is locally optimal for 2-Opt is  $O(\sqrt[p]{\phi})$ , where  $d$  represents the number of dimensions.*

### 3 Analysis for Manhattan Instances

In this section, we first present the analysis for RLS and (1+1) EA based on the minimum possible improvement for a single 2-Opt step. We later extend the analysis for the improvement in a sequence of consecutive 2-Opt steps. The fitness gain results hold for the considered algorithms if they have not reached a local optimum. Otherwise, we provide an upper bound on the approximation ratio. Some of our results are also stated for a variant of the (1+1) EA, called (1+1) EA\*, which is defined later on in Algorithm 3.

**Theorem 4.** *For  $\phi$ -perturbed Manhattan instances and for RLS, (1+1) EA and (1+1) EA\*, the approximation ratio for the worst local optimum is bounded above by  $O(\sqrt{\phi})$ .*

*Proof.* We consider Manhattan instances in a 2 dimensional unit hypercube  $[0, 1]^2$ . Then as a direct consequence from Theorem 3 the expected approximation ratio is at most  $O(\sqrt{\phi})$ .  $\square$

#### 3.1 Analysis of a Single 2-Opt Step

We start by showing a lower bound on the fitness gain achievable by RLS.

**Theorem 5.** *In  $t$  generations, RLS achieves an expected fitness gain of  $\Omega(t/(n^6\phi))$  unless a local optimum is reached.*

*Proof.* Based on Theorem 1, we get

$$\Pr(\Delta_{\min} \geq \varepsilon) \geq 1 - 4!^2 \varepsilon n^4 \phi = 1 - 576 \varepsilon n^4 \phi.$$

as a lower bound on the probability that the minimum improvement is at least  $\varepsilon$ .

Let  $\Delta$  denote the random variable describing the fitness gain obtained in a single improving 2-Opt step. This is obviously no less than the minimum possible improvement  $\Delta_{\min}$ . For any fixed  $\varepsilon > 0$ , the expected fitness gain per one improving 2-Opt step can be bounded from below as follows:

$$\begin{aligned} E(\Delta) &= \int_{\Delta} \Pr(\Delta) \cdot \Delta \\ &\geq \Pr(\Delta \geq \varepsilon) \cdot \varepsilon \\ &\geq \Pr(\Delta_{\min} \geq \varepsilon) \cdot \varepsilon \\ &\geq (1 - 576 \varepsilon n^4 \phi) \cdot \varepsilon \end{aligned}$$

Setting  $\varepsilon = 1/(2 \cdot 576 n^4 \phi)$  we get

$$\Pr(\Delta_{\min} \geq \varepsilon) \geq 1/2$$

and accordingly

$$E(\Delta) \geq 1/(2304 n^4 \phi).$$

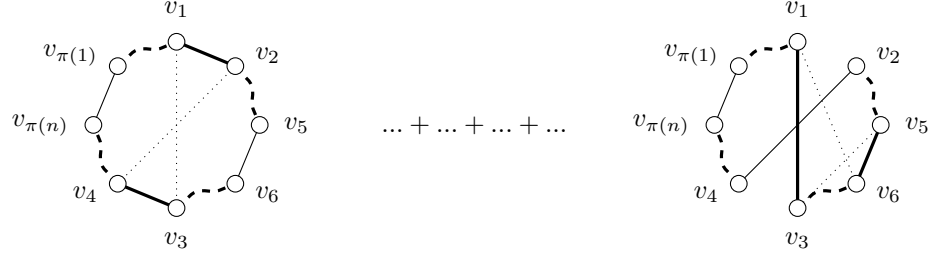


Figure 2: An example for a linked pair: The edges  $\{v_1, v_2\}$  and  $\{v_3, v_4\}$  are replaced by  $\{v_1, v_3\}$  and  $\{v_2, v_4\}$  in the first 2-Opt step, and  $\{v_1, v_3\}$  and  $\{v_5, v_6\}$  are replaced by  $\{v_1, v_6\}$  and  $\{v_3, v_5\}$  in the second 2-Opt step resulting in a linked pair formed by the edges  $\{v_2, v_4\}$  and  $\{v_5, v_6\}$ .

The number of mutations occurring in one generation is 1 and the probability for an improving 2-Opt step is at least  $1/\binom{n}{2} \geq 2/n^2$  if the current solution is not locally optimal. Therefore, the expected value for the fitness gain  $\Delta$  in any 2-Opt step can be bounded from below by

$$E(\Delta) \geq 1/(1152n^6\phi).$$

Hence, the expected value for the fitness gain in  $t$  generations if no locally optimal solution has been obtained in between can be derived as  $E(\Delta_t) \geq t/(1152n^6\phi)$ .  $\square$

**Theorem 6.** *In  $t$  generations (1+1) EA achieves an expected fitness gain of  $\Omega(t/(n^6\phi))$  unless a local optimum is reached.*

*Proof.* We only account for the progress in singular steps. The expectation for the fitness gain in a singular improving 2-Opt step can be derived following the above proof in Theorem 5. The probability of a singular mutation occurring in a generation is  $1/e$  due to the Poisson distribution with expectation 1. The probability of a singular improving 2-Opt step is therefore at least  $2/(en^2)$ . Accordingly, and following the steps in Theorem 5 the expected value for the fitness gain in  $t$  generations can be derived as  $E(\Delta_t) \geq t/(1152en^6\phi)$ .  $\square$

### 3.2 Analysis of Linked Steps for RLS

The lower bound for the expected fitness gain presented in the previous section is based on the minimum improvement in a single 2-Opt step. This bound can be improved further by considering the improvement made in a sequence of consecutive 2-Opt steps.

The analysis of consecutive steps in Englert et al. [9] is based on the number of disjoint pairs of 2-Opt steps linked by an edge, such that in one step an edge is added and in the other it is removed. We call such a pair of 2-Opt steps a *linked pair* (see Figure 2). Different types of linked pairs of 2-Opt steps are considered as follows. Let  $\{v_1, v_2\}$  and  $\{v_3, v_4\}$  be the edges that are replaced by  $\{v_1, v_3\}$  and  $\{v_2, v_4\}$  in the first 2-Opt step, and  $\{v_1, v_3\}$  and  $\{v_5, v_6\}$  be replaced by  $\{v_1, v_6\}$  and  $\{v_3, v_5\}$  in the second 2-Opt step. This second step could occur anytime after the first step in a sequence of 2-Opt steps.

Following [9], we consider three different types of steps:

type 0:  $|\{v_2, v_4\} \cap \{v_5, v_6\}| = 0$ .

type 1:  $|\{v_2, v_4\} \cap \{v_5, v_6\}| = 1$ .



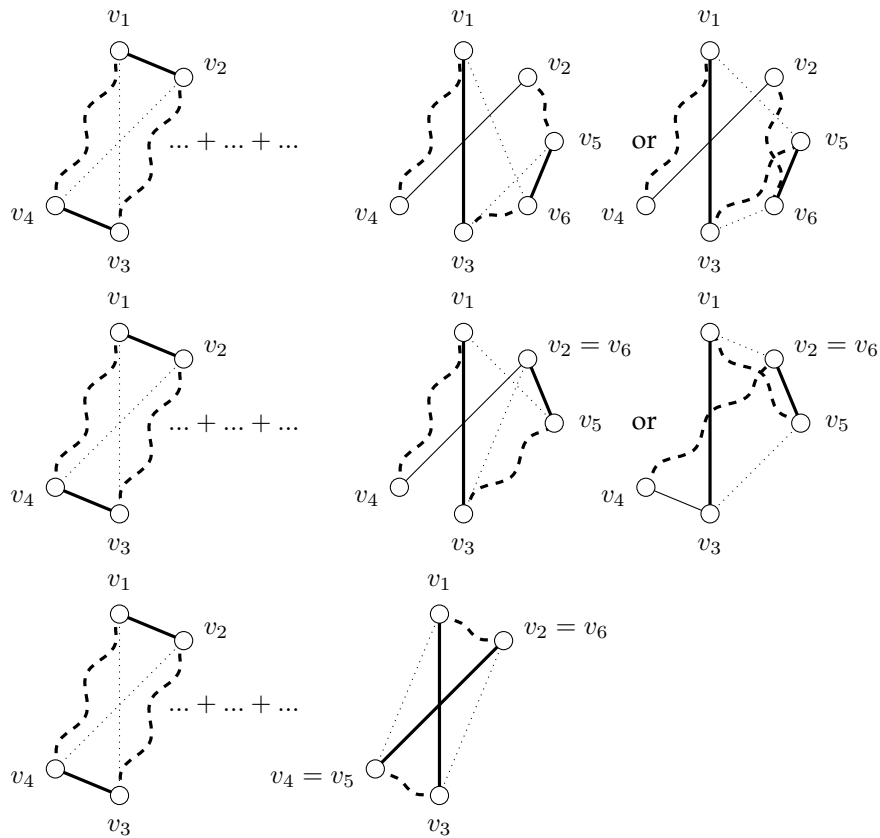


Figure 3: Types of linked pairs: type 0 (top) type 1 (middle) and type 2 (bottom). The existing edges to be removed, the new edges to be added by the 2-Opt step and the path segments connecting multiple edges are indicated in bold black lines, dotted lines and dashed path segments respectively. Note that for some cases (type 0 case 2, type 1 and 2 all cases), unrelated 2-Opt steps in between (represented by ... + ... + ...) are necessary to alter the tour to enable the second 2-Opt step.

$$\text{type 2: } |\{v_2, v_4\} \cap \{v_5, v_6\}| = 2.$$

Examples for the three types of the linked steps are shown in Figure 3. As explained in [9], it is important to limit the number of occurrences of type 2 as no guarantee on the fitness gain made by type 2 steps is available. We need to show that there is a sufficient number of linked pairs of type 0 and 1 as for the linked pairs of type 0 and 1 a good progress can be guaranteed.

Due to [9, Lemma 9] there are at least  $t/6 - 7n(n-1)/24$  such pairs in a sequence of  $t$  consecutive 2-Opt steps. The analysis in [9] considers all 2-Opt steps  $S_1, \dots, S_t$  in sequence and constructs (disjoint) linked pairs (of any type) in a greedy fashion. When processing some step  $S_i$ , we search for steps  $S_j$  and  $S'_j$ , where the two edges inserted by  $S_i$  are being removed again, if such steps exist. If either  $S_j$  or  $S'_j$  exist, the respective pair  $(S_i, S_j)$  or  $(S_i, S'_j)$  is being added to a list of disjoint linked 2-Opt steps, and both  $S_j$  and  $S'_j$  are being removed from the list to ensure disjointness of pairs. For an example of this process, we refer to Figure 6 in Englert et. al. [9]. The proof of [9, Lemma 9]

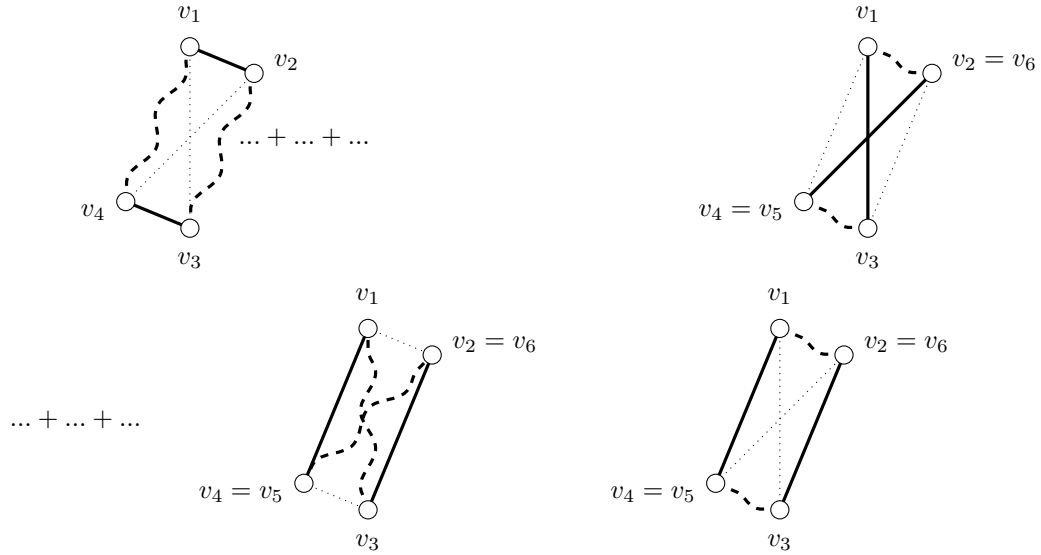


Figure 4: Example scenario that the steps  $S_i$  (top left),  $S_j$  (top right),  $j > i$  form a type 2 pair and then the steps  $S_j, S_l$  (bottom left),  $l > j$  and the steps  $S_j, S'_l$  (bottom right),  $l' > j$  form type 2 pairs. These steps are not accepted as the improvement is negative.

shows that when removing all pairs of type 2 from this list, at least  $t/6 - 7n(n - 1)/24$  pairs of type 0 or 1 remain.

We further improve this bound, considering the fact that the possible number of pairs excluded is constrained by the number of edges in the final tour.

**Lemma 7.** *Let  $u$  be the total number of linked pairs in an improving 2-Opt sequence. Then the number of linked pairs  $v$  of type 0 or 1 in that sequence is at least  $u/2 - n/4$ .*

*Proof.* Consider a fixed pair of 2-Opt steps linked by an edge. Without loss of generality assume that in the first step  $S_i$  the edges  $\{v_1, v_2\}$  and  $\{v_3, v_4\}$  are exchanged with the edges  $\{v_1, v_3\}$  and  $\{v_2, v_4\}$ , for distinct vertices  $v_1, \dots, v_4$ . Also without loss of generality assume that in the second step  $S_j$  the edges  $\{v_1, v_3\}$  and  $\{v_5, v_6\}$  are exchanged with the edges  $\{v_1, v_5\}$  and  $\{v_3, v_6\}$ . Consider the two steps  $S_i$  and  $S_j$  with  $i < j$  that form a pair of type 2. Let us consider the next steps as  $S_l$  with  $l > j$  in which the edge  $\{v_1, v_4\}$  is removed from the tour, if such a step exists, and  $S_{l'}$  with  $l' > j$  in which the edge  $\{v_2, v_3\}$  is removed from the tour if such a step exists. Observe that neither  $(S_j, S_l)$  nor  $(S_j, S'_l)$  can be a pair of type 2 because otherwise the improvement of one of the steps  $S_i, S_j$ , and  $S_l$  or one of the steps  $S_i, S_j$ , and  $S_{l'}$  must be negative. In particular, we must have  $l \neq l'$ . Figure 4 illustrates this situation.

Thus, there cannot be a type 2 linked pair that associates with another type 2 linked pair. Therefore, each of the type 2 pairs  $(S_i, S_j)$  can be associated with at most two different pairs  $(S_j, S_l)$  and  $(S_j, S_{l'})$  of type 0 or 1, unless the steps  $S_l$  or  $S_{l'}$  are undefined. This happens if the edges added to the tour in  $S_j$  are never removed. Since the final tour contains  $n$  edges, at most  $n/2$  pairs are excluded due to this. If we consider the number of type 2 pairs as  $x$  then the total number of pairs of type 0 or 1 must be at least  $x - n/2$ . This implies  $u \geq x + (x - n/2)$  and  $x \leq u/2 + n/4$ . The number of good (type 0 or 1) pairs is therefore  $u - x \geq u/2 - n/4$ .  $\square$

Now we can estimate the number of good pairs in a sequence of consecutive 2-Opt steps. Following the above argument, we can improve [9, Lemma 8] on the total number of (disjoint) linked pairs. The below Lemma 8 provides an improved bound for the number of good disjoint pairs in a sequence of 2-Opt steps.

**Lemma 8.** *In every sequence of  $t$  consecutive 2-Opt steps, the number of disjoint pairs of 2-Opt steps of type 0 or 1 is at least  $t/6 - n/2$ .*

*Proof.* From [9, Lemma 8], for each processed 2-Opt step  $S_i$  at most two other steps  $S_j$   $S_{j'}$  are excluded from being processed if  $j$  or  $j'$  is defined. Hence, for a sequence of  $t$  steps there are at least  $t/3$  pairs, from which we have to subtract the number of steps  $S_i$  where neither  $j$  nor  $j'$  are defined. This happens if they are currently in the tour. Therefore, this number is  $n/2$  considering that the number of edges in the final tour is exactly  $n$ , and  $S_i$  can only be excluded if both edges inserted in the tour are never removed again. Therefore, the total number of disjoint pairs is at least  $u = t/3 - n/2$ . Combining the result of above Lemma 7 to this we obtain the number of disjoint pairs of type 0 and 1.  $\square$

Due to above Lemma 8 there are at least  $t/6 - n/2$  such pairs in a sequence of  $t$  consecutive 2-Opt steps. Here we consider the probability of both 2-Opt steps in a linked pair having improvement at least  $\varepsilon$ . We derive this probability from the following Lemma from Englert et al. [9] which bounds the probability that improvement falls within the interval of  $(0, \varepsilon]$ .

**Lemma 9** ([9, Lemma 10]). *In a  $\phi$ -perturbed  $L_1$  instance with  $n$  vertices, the probability that there exists a pair of type 0 or 1 in which both 2-Opt steps are improvements by at most  $\varepsilon$  is bounded by  $O(n^6 \cdot \varepsilon^2 \cdot \phi^2)$ .*

Based on the above Lemmas (8 and 9), we can bound the fitness gain for a given number of  $t$  generations. Note that the theorem requires a lower bound on the number of generations as only for large enough  $t$  we can guarantee that linked 2-Opt steps of type 0 or 1 do exist.

**Theorem 10.** *In  $t \geq cn^3$  generations,  $c > 3/2$  constant, RLS obtains an expected fitness gain of  $\Omega(t/(n^5\phi))$  unless a local optimum is reached.*

*Proof.* Let  $\Delta_{\min}$  denote the minimum possible improvement made by any pair of type 0 or 1. Using above result in Lemma 9,

$$\Pr(\Delta_{\min} \geq \varepsilon) \geq 1 - c'n^6 \cdot \varepsilon^2 \cdot \phi^2$$

for a constant  $c' > 0$ . Let  $\Delta$  be the random variable describing the fitness gain obtained in a pair of linked 2-Opt steps of type 0 or 1. For any  $\varepsilon > 0$ , the expected fitness gain  $E(\Delta)$  can be bounded as follows.

$$\begin{aligned} E(\Delta) &= \int_{\Delta} \Pr(\Delta) \cdot \Delta \\ &\geq \Pr(\Delta \geq \varepsilon) \cdot \varepsilon \\ &\geq (1 - c'n^6 \cdot \varepsilon^2 \cdot \phi^2) \cdot \varepsilon. \end{aligned}$$

Setting  $\varepsilon = 1/(\sqrt{(c'2n^6 \cdot \phi^2)})$  we get  $\Pr(\Delta_{\min} \geq \varepsilon) \geq 1/2$  and as a consequence

$$E(\Delta) \geq 1/(2\sqrt{(2c'n^6 \cdot \phi^2)}) \geq 1/(2\sqrt{2}\sqrt{c'n^3\phi}) = \Omega(1/(n^3\phi)).$$

The probability of making an improving 2-Opt step is at least  $1/\binom{n}{2} \geq 2/n^2$ , so long as no local optimum has been reached. Therefore, the expected number of improving 2-Opt steps made in  $t$  generations is at least  $2t/n^2$ . Let  $t^*$  be the number of improving steps. The improving steps are the steps accepted by RLS generating a sequence of consecutive steps. By Lemma 8 we know there are at least  $t^*/6 - n/2$  disjoint type 0 or 1 pairs in such a sequence of  $t^*$  consecutive steps. Therefore,

$$t^*/6 - n/2 \geq \frac{2t}{6n^2} - \frac{n}{2}.$$

As  $t \geq cn^3$  for  $c > 3/2$ , we have  $2t = (2 - 3/c) \cdot t + (3t)/c \geq (2 - 3/c) \cdot t + 3n^3$ . Then

$$\frac{2t}{6n^2} - \frac{n}{2} \geq \frac{(2 - 3/c) \cdot t}{6n^2} = \Omega(t/n^2).$$

A lower bound for the expected fitness gain for  $t$  generations is therefore

$$E(\Delta_t) = E(\Delta) \cdot \Omega(t/n^2) \geq \Omega(t/(n^5\phi)).$$

□

### 3.3 Analysis of Linked Steps for (1+1) EA

The challenge for analyzing the (1+1) EA instead of RLS lies in the fact that the (1+1) EA can execute multiple 2-Opt steps in one generation. For RLS Englert et al. [9] showed that certain pairs of improving 2-Opt steps yield a large fitness increase on perturbed instances, with high probability. Executing multiple 2-Opt steps in one generation complicates this argument, as some of these mutations may not be improving. As such, they might interfere with the mentioned pairs, and prohibit a large fitness increase. In the following, we show that there are sufficiently many linked 2-Opt operations that take place in generations where only one 2-Opt step is executed.

To this end, we consider a slightly modified variant of the (1+1) EA, which we call (1+1) EA\* (see Algorithm 3). The (1+1) EA\* will exclude generations containing multiple 2-Opt steps where an edge  $e$  is being inserted in one of these 2-Opt steps and being removed in a later 2-Opt step of the same generation.

---

#### Algorithm 3 (1+1) EA\*

---

```

 $x \leftarrow$  a random permutation of  $n$  cities.
repeat
  draw  $s$  from a Poisson distribution with expectation 1.
   $y := x$ 
  for  $s + 1$  times do
     $y \leftarrow$  apply a 2-Opt step chosen uniformly at random to  $y$ .
  end for
  if  $f(y) \leq f(x)$  then
    Check whether one of the above mutations has removed an edge that was inserted
    in the same generation. If so, reject  $y$ . Otherwise,  $x \leftarrow y$ .
  end if
until forever

```

---

The purpose of this modification is to enable a theoretical analysis as some of the excluded steps are difficult to handle. (1+1) EA and (1+1) EA\* show identical behavior most of the time; it is easy to show that the probability of removing an edge that was inserted in the same generation is  $\Theta(1/n)$ . So (1+1) EA and (1+1) EA\* are identical most

of the time, apart from a small fraction of steps. However, even a small fraction can be harmful as it can take the algorithm on a different trajectory, with unforeseen consequences. Note that, when the (1+1) EA\* does behave differently from the (1+1) EA, it rejects a new offspring that would otherwise improve the current tour. We therefore believe that we are being pessimistic by considering the progress of the (1+1) EA\* instead of that of the (1+1) EA. Experiments presented in Section 6 will further investigate the difference between the (1+1) EA and the (1+1) EA\*.

The following lemma tries to estimate the number of disjoint linked pairs of 2-Opt steps within a sequence of  $t$  generations. First, we estimate the probability of a 2-Opt step forming a linked pair if one of the matching 2-Opt steps exists. Therefore, inside the lemma we consider the internal 2-Opt steps  $x_{\ell^0}$  to  $x_{\ell^k}$  within a generation  $\ell$ . Then we combine this result with the number of improving and singular steps in a sequence following Lemma 7 to obtain a lower bound on the expected number of good pairs having all singular 2-Opt steps.

**Lemma 11.** *In every sequence of  $t$  generations of the (1+1) EA\*, the expected number of disjoint pairs of 2-Opt steps, all of which are singular, is at least*

$$\frac{t}{3e^2n^2} - n/2,$$

*unless a local optimum is reached beforehand.*

*Proof.* Recall the definitions for *improving* and *singular* 2-Opt steps in Section 2.2. We adapt the proof of Lemma 8 in [9] to take into account steps that are rejected by the (1+1) EA\*, and the fact that the (1+1) EA\* can accept non-improving 2-Opt steps in generations with multiple 2-Opt steps.

Let  $\mathcal{S} = S_1, S_2, \dots$  be a list of all 2-Opt steps executed in  $t$  generations. Then we process this list to create a list  $\mathcal{L}$  of linked 2-Opt steps, all of which are singular.

The probability of the (1+1) EA\* making an improving 2-Opt step is at least  $1/\binom{n}{2} \geq 2/n^2$ , so long as no local optimum has been reached. The probability of a singular generation is  $1/e$  due to the Poisson distribution. So, the probability of having an improving and singular step is at least  $2/(en^2)$ .

Following the proof of Lemma 9 we process steps  $S_i, S_j$  and  $S'_j$  to create  $\mathcal{L}$ . The only difference is that here we consider  $S_i$  to be both improving and singular.

We estimate the probability of a step  $S_j$  occurring and being a singular step. Let  $A(e_3)$  denote the event that an accepted generation contains an improving 2-Opt step where  $e_3$  is being removed from the tour.

Let  $R(e_3)$  denote the set of all edges  $e$  such that a 2-Opt move removing  $e_3$  and  $e$  results in a strict fitness improvement. Let  $x_{\ell^k}$  denote the search point of the (1+1) EA\* at generation  $\ell$ , 2-Opt step  $k$  within that generation, and let us regard  $x_{\ell^k}$  as a set of edges in the tour. Note that then  $|R(e_3) \cap x_{\ell^k}|$  describes the number of improving 2-Opt moves where  $e_3$  is being removed from the tour.

Let  $\ell^0$  be the index of the first 2-Opt step in a new generation, and let  $S + 1$  be the random number of 2-Opt steps being executed in that generation. If  $S = 0$ , that is, only one 2-Opt step is executed, the conditional probability of  $A(e_3)$  is given by

$$\Pr(A(e_3) \mid S = 0) = \frac{|R(e_3) \cap x_{\ell^0}|}{\binom{n}{2}} = p. \quad (1)$$

If  $S + 1 > 1$  operations are being executed in that generation, the probability of  $A(e_3)$

is bounded by the union bound:

$$\Pr(A(e_3) \mid S = s) \leq \sum_{k=0}^s \frac{|R(e_3) \cap x_{\ell^k}|}{\binom{n}{2}}.$$

Note that  $|R(e_3) \cap x_{\ell^k}|$  might increase if edges from  $R(e_3)$  are being inserted into the tour. However, the additional selection criterion on the (1+1) EA\* implies that, if a following step removes  $e_3$  and one of the edges inserted previously, in the same generation, this sequence of 2-Opt steps will be rejected. Thus, only  $|R(e_3) \cap x_{\ell^k}|$  edges can cause  $A(e_3)$  and

$$\Pr(A(e_3) \mid S = s) \leq \sum_{k=0}^s \frac{|R(e_3) \cap x_{\ell^k}|}{\binom{n}{2}} = (s+1) \cdot p.$$

Note that, using the law of total probability,

$$\begin{aligned} \Pr(A(e_3)) &= \sum_{s=0}^{\infty} \frac{1}{e^s} \cdot \Pr(A(e_3) \mid S = s) \\ &\leq \sum_{s=0}^{\infty} \frac{1}{e^s} \cdot (s+1)p = 2p. \end{aligned} \tag{2}$$

Combining (1) and (2) with Bayes' Theorem, we get

$$\begin{aligned} \Pr(S = 0 \mid A(e_3)) &= \frac{\Pr(A(e_3) \mid S = 0) \cdot \Pr(S = 0)}{\Pr(A(e_3))} \\ &\geq \frac{p \cdot 1/e}{2p} = \frac{1}{2e}. \end{aligned}$$

It follows that the probability of finding a linked pair  $(S_i, S_j)$  or  $(S_i, S'_j)$  is at least  $1/(2e)$ , if one of the steps  $S_j$  or  $S'_j$  exists.

Recall that the expected number of singular and improving steps  $S_i$  in  $t$  generations is at least  $2t/(en^2)$ . Following the same arguments in Lemma 7 provides an expected number of at most  $2t/(3en^2) - n/2$  processed elements  $S_i$ , each of which has a probability of  $1/(2e)$  for pairing with some  $S_j$  or  $S'_j$ , if one of them exists. Hence, the resulting expected number of pairs is at least

$$\frac{t}{3e^2n^2} - n/2.$$

□

The following Theorem now gives a lower bound on the expected fitness gain of the (1+1) EA\*.

**Theorem 12.** *In  $t \geq cn^3$  generations,  $c > 3e^2$  constant, (1+1) EA\* obtains an expected fitness gain of  $\Omega(t/(n^5\phi))$  unless a local optimum is reached.*

*Proof.* As in the proof of Theorem 10, we have for a constant  $c' > 0$

$$E(\Delta) \geq 1/(c'n^3\phi).$$

From Lemma 11 we know that the expected number of disjoint pairs of 2-Opt steps, both of which are singular, is at least

$$u = \frac{t}{3e^2n^2} - \frac{n}{2}.$$

Lemma 7 implies that among these there are at least

$$\frac{u}{2} - \frac{n}{4} = \frac{t}{6e^2n^2} - \frac{n}{2} \geq \frac{(1 - 3e^2/c)t}{6e^2n^2} = \Omega(t/n^2)$$

type 0 or 1 pairs. The expected fitness gain in  $t$  generations,  $E(\Delta_t)$ , is therefore at least

$$E(\Delta) \cdot \Omega(t/n^2) \geq \Omega(t/(n^5\phi)).$$

□

### 3.4 The Approximation Ratio over the Algorithm Run

We further interpret our fitness gain results in terms of the expected approximation ratio. The approximation ratio of an iterative algorithm is defined as

$$\text{App}_t := \frac{\text{Tour}_t}{\text{OPT}}$$

where  $\text{Tour}_t$  is the tour length of the algorithm at time  $t$ , and  $\text{OPT}$  is the length of an optimal tour. In order to bound  $\text{OPT}$ , we state the following result as a consequence of Theorem 4 of Englert et al. [9].

**Corollary 13.** *For metric TSP instances in  $[0, 1]^2$  with probability  $1 - e^{-n/128}$*

$$\text{OPT} \geq \frac{\sqrt{n}}{288\sqrt{\phi}}.$$

*Proof.* The statement was implicitly shown in the proof of Theorem 4 of Englert et al. [9]. Their result applies for unit hypercubes of arbitrary dimensions  $d$ , but here we specialise their result to  $d = 2$ . They show for an integer  $\ell \leq \sqrt{n\phi}$  that the length of an optimal tour is at least [9, page 236]

$$\text{OPT} \geq \frac{n}{2^5 3^2 \ell} \geq \frac{\sqrt{n}}{288\sqrt{\phi}}$$

in a case that occurs with probability  $1 - e^{-n/128}$  [9, (5.2)].

□

As the bound on the length of the optimal tour holds with overwhelming probability, the following lemma shows that the expected approximation ratio after  $t$  steps is bounded by the expected tour length after  $t$  steps, divided by the upper bound on  $\text{OPT}$ , modulo an exponentially small additive term.

**Lemma 14.** *The expected approximation ratio after  $t$  steps is at most*

$$E(\text{App}_t) \leq E(\text{Tour}_t) \cdot \frac{288\sqrt{\phi}}{\sqrt{n}} + e^{-\Omega(n)}.$$

*Proof.* By the law of total expectation,

$$\begin{aligned} E\left(\frac{\text{Tour}_t}{\text{OPT}}\right) &= E\left(\frac{\text{Tour}_t}{\text{OPT}} \mid \text{OPT} \geq \frac{\sqrt{n}}{288\sqrt{\phi}}\right) \cdot \Pr\left(\text{OPT} \geq \frac{\sqrt{n}}{288\sqrt{\phi}}\right) \\ &\quad + E\left(\frac{\text{Tour}_t}{\text{OPT}} \mid \text{OPT} < \frac{\sqrt{n}}{288\sqrt{\phi}}\right) \cdot \Pr\left(\text{OPT} < \frac{\sqrt{n}}{288\sqrt{\phi}}\right) \\ &\leq E(\text{Tour}_t) \cdot \frac{288\sqrt{\phi}}{\sqrt{n}} + E\left(\frac{\text{Tour}_t}{\text{OPT}} \mid \text{OPT} < \frac{\sqrt{n}}{288\sqrt{\phi}}\right) \cdot e^{-n/128}. \end{aligned}$$

We estimate the remaining conditional expectation by  $n/2$ , using the following argument from [9, page 236] that applies to all tours. Consider the longest edge in  $\text{Tour}_t$  and let  $\alpha$  be its length and  $u, v$  denote its end points. We have  $\text{Tour}_t \leq n \cdot \alpha$  as  $\alpha$  is the length of its longest edge. Each optimal tour also has to connect  $u$  and  $v$ . By the triangle inequality, any path between  $u$  and  $v$  is no shorter than the direct edge of length  $\alpha$ . Hence  $\text{OPT} \geq 2 \cdot \alpha$  and  $\text{Tour}_t / \text{OPT} \leq n/2$ .

The expected approximation ratio is thus at most

$$E(\text{Tour}_t) \cdot \frac{288\sqrt{\phi}}{\sqrt{n}} + \frac{n}{2} \cdot e^{-n/128}$$

proving the claim as  $n/2 \cdot e^{-n/128} = e^{-n/128 + \ln(n/2)} = e^{-\Omega(n)}$ .  $\square$

**Theorem 15.** *Based on single step improvements, for Manhattan instances and for RLS, the expected approximation ratio after  $t$  generations is at most  $(576/3) \cdot \sqrt{n\phi} - t/(4n^{6.5}\sqrt{\phi}) + e^{-\Omega(n)}$  if the current solution is not already locally optimal.*

*Proof.* Based on Lemma 14, the expected approximation ratio at iteration  $t$  is

$$E(\text{App}_t) \leq E(\text{Tour}_t) \cdot \frac{288\sqrt{\phi}}{\sqrt{n}} + e^{-\Omega(n)} \leq E(\text{Tour}_0 - \Delta_t) \cdot \frac{288\sqrt{\phi}}{\sqrt{n}} + e^{-\Omega(n)}.$$

where the expected fitness gain for  $t$  iterations is  $\Delta_t$ . As the initial tour  $\text{Tour}_0$  is generated uniformly at random,  $E(\text{Tour}_0)$  is bounded as follows. By linearity of expectation,  $E(\text{Tour}_0)$  equals  $n$  times the expected Manhattan distance between two points  $P_1(x_1, y_1), P_2(x_2, y_2)$  chosen uniformly at random from the unit square. The latter is twice the expected absolute difference of  $x$ -coordinates:

$$\begin{aligned} E(|x_1 - x_2|) &= \int_0^1 \int_0^1 |x_1 - x_2| dx_2 dx_1 \\ &= \int_0^1 \left( \int_0^{x_1} z dz + \int_0^{1-x_1} z dz \right) dx_1 \\ &= \int_0^1 \left( \frac{x_1^2}{2} + \frac{(1-x_1)^2}{2} \right) dx_1 \\ &= \int_0^1 \left( \frac{1}{2} - x_1 + x_1^2 \right) dx_1 = \frac{1}{3}. \end{aligned}$$



Hence  $E(\text{Tour}_0) = 2n/3$ . Therefore, substituting the values for the starting tour  $\text{Tour}_0$  and the expected fitness gain  $E(\Delta)$  from the proof of Theorem 5 this yields

$$\begin{aligned} E(\text{App}_t) &\leq E(\text{Tour}_0 - \Delta_t) \cdot \frac{288\sqrt{\phi}}{\sqrt{n}} + e^{-\Omega(n)} \leq ((2n/3) - t/(1152n^6 \cdot \phi)) \cdot \frac{288\sqrt{\phi}}{\sqrt{n}} + e^{-\Omega(n)} \\ &= \left( (576/3) \cdot \sqrt{n\phi} - t/(4n^{6.5}\sqrt{\phi}) \right) + e^{-\Omega(n)}. \end{aligned}$$

to the final result of the theorem if not locally optimal.  $\square$

Similarly, we can derive upper bounds on the expected approximation ratio after  $t$  generations for the considered algorithms and the Manhattan metric as presented in Table 1.

Algorithm	Manhattan metric	
	Single Step (any $t$ )	Consecutive Steps
RLS	$(576/3) \cdot \sqrt{n\phi} - \frac{t}{4n^{6.5}\sqrt{\phi}}$	$(576/3) \cdot \sqrt{n\phi} - \frac{288c_{10}t}{n^{5.5}\sqrt{\phi}}, t \geq (3/2) \cdot n^3$
(1+1) EA/(1+1) EA*	$(576/3) \cdot \sqrt{n\phi} - \frac{t}{4\epsilon n^{6.5}\sqrt{\phi}}$	$(576/3) \cdot \sqrt{n\phi} - \frac{288c_{12}t}{n^{5.5}\sqrt{\phi}}, t \geq 3e^2 \cdot n^3$

Table 1: Upper bounds on the expected approximation ratio in  $t$  generations for RLS and (1+1) EA or (1+1) EA\* for Manhattan instances due to single-step and consecutive-steps analysis if the current solution is not locally optimal. (The  $e^{-\Omega(n)}$  term is omitted for clarity.) Otherwise, the approximation ratio is  $O(\sqrt{\phi})$ . The results for RLS single step, consecutive steps and (1+1) EA single step and consecutive steps are based on Theorems 5, 10, 6 and 12 respectively. The constants  $c_{10}$  and  $c_{12}$  are the hidden constants in the expected fitness gain from Theorem 10 and Theorem 12, respectively.

## 4 Analysis for Euclidean Instances

We now turn our attention to Euclidean instances. First, we obtain the expected progress based on a single 2-Opt step for RLS and (1+1) EA, and later improve these results by analyzing a sequence of consecutive 2-Opt steps. The fitness gain results presented in this section hold for the considered algorithms if they have not reached a local optimum. Otherwise, we provide an upper bound on the approximation ratio.

**Theorem 16.** *For  $\phi$ -perturbed Euclidean instances and for RLS, (1+1) EA and (1+1) EA\*, the approximation ratio for the worst local optimum is bounded above by  $O(\sqrt{\phi})$ .*

*Proof.* We consider Euclidean instances in a 2 dimensional unit hypercube  $[0, 1]^2$ . Then as a direct consequence from Theorem 3 the expected approximation ratio is at most  $O(\sqrt{\phi})$ .  $\square$

### 4.1 Analysis of a Single 2-Opt Step

**Theorem 17.** *In  $t$  generations RLS achieves an expected fitness gain of  $\Omega(t/(\log(n\phi) \cdot (n^6\phi^3)))$  unless a local optimum is reached.*

*Proof.* Due to Lemma 2, we have

$$\Pr(\Delta_{\min} \geq \varepsilon) \geq 1 - 1536n^4 \cdot \varepsilon \cdot \log(1/\varepsilon) \cdot \phi^3.$$

Let  $\Delta$  denote the random variable that describes the fitness gain in an improving 2-Opt step. Then similar to the proof for the Manhattan instances (Theorem 5), we get

$$E(\Delta) \geq (1 - (1536n^4 \cdot \varepsilon \cdot \log(1/\varepsilon) \cdot \phi^3)) \cdot \varepsilon.$$

Let us set  $\varepsilon = c/(1536 \log(n\phi) \cdot (n^4\phi^3))$ ,  $c > 0$  a small enough constant such that  $\Pr(\Delta_{\min} \geq \varepsilon) \geq 1/2$ , which is established in the following calculation. Let us consider the formula  $1536 \cdot n^4\phi^3 \cdot \varepsilon \cdot \log(1/\varepsilon)$ . By substituting the value for the  $\varepsilon$  this gives,

$$\begin{aligned} & 1536 \cdot n^4\phi^3 \cdot \frac{c}{1536 \cdot n^4\phi^3 \cdot \log(n\phi)} \cdot \log(c^{-1} \cdot 1536 \cdot n^4\phi^3 \cdot \log(n\phi)) \\ &= \frac{c}{\log(n\phi)} \cdot \log(c^{-1} \cdot 1536 \cdot n^4\phi^3 \cdot \log(n\phi)) \end{aligned}$$

$\log(n\phi)$  is less than  $n\phi$ . Therefore the above formula is

$$\begin{aligned} & \leq \frac{c}{\log(n\phi)} \cdot \log(c^{-1} \cdot 1536 \cdot n^5\phi^4) \\ & \leq \frac{c}{\log(n\phi)} \cdot (5 \log(n\phi) + \log(1536/c)) \\ & = 5c + \frac{c \log(1536/c)}{\log(n\phi)} \leq 1/2, \text{ for a small enough } c > 0. \end{aligned}$$

Subsequently, we get

$$E(\Delta) \geq c/(\log(n\phi) \cdot (2n^4\phi^3)).$$

The number of mutations occurring in one generation is 1 and the probability for an improving 2-Opt step is at least  $1/\binom{n}{2} \geq 2/n^2$ . Therefore, the expected value for the fitness gain  $\Delta$  in any generation is

$$E(\Delta) \geq c/(\log(n\phi) \cdot (n^6\phi^3)).$$

The expected value for the fitness gain in  $t$  generations is

$$\begin{aligned} E(\Delta_t) & \geq ct/(\log(n\phi) \cdot (n^6\phi^3)) \\ & = \Omega(t/(\log(n\phi) \cdot (n^6\phi^3))). \end{aligned}$$

□

**Theorem 18.** *In  $t$  generations (1+1) EA achieves an expected fitness gain of  $\Omega(t/(\log(n\phi) \cdot (n^6\phi^3)))$  unless a local optimum is reached.*

*Proof.* The expected fitness gain for an improving singular generation can be derived following the above proof on RLS in Theorem 17 with the exception that here the probability of a single step mutation occur in a generation is  $1/e$ . Hence, the expected fitness gain  $E(\Delta)$  for any generation is

$$E(\Delta) \geq c/(\log(n\phi) \cdot (en^6\phi^3)).$$

Accordingly, the expected value for the fitness gain in  $t$  generations is  $ct/(\log(n\phi) \cdot (en^6\phi^3)) = \Omega(t/(\log(n\phi) \cdot (n^6\phi^3)))$ .

□

## 4.2 Analysis of Linked Steps for RLS

The above lower bounds are based on the minimum possible improvement a single 2-Opt step can make. We can further improve this bound considering the improvement made in a sequence of consecutive steps. Similar to the analysis on the consecutive steps for Manhattan instances in Section 3 here we also consider the set of linked pairs of type 0 and 1. In a sequence of  $t$  generations, there are at least  $t/6 - n/2$  such pairs due to Lemma 8. Here we consider Lemma 14 in Englert et al. [9] related to the probability of existence of each of the two types of linked pairs in a sequence of consecutive steps for Euclidean instances. Based on these we can bound the expected fitness gain made in  $t$  generations.

**Lemma 19** ([9, Lemma 14]). *For  $\phi$ -perturbed  $L_2$  instances, the probability that there exists a pair of type 0 and 1 in which both 2-Opt steps are improvements by at most  $\varepsilon \leq 1/2$  is bounded by  $O(n^6 \cdot \phi^5 \cdot \varepsilon^2 \cdot \log^2(1/\varepsilon)) + O(n^5 \cdot \phi^4 \cdot \varepsilon^{3/2} \log(1/\varepsilon))$ .*

**Theorem 20.** *In  $t \geq cn^3$  generations,  $c > 3/2$  constant, RLS achieves an expected fitness gain of  $\Omega(t/(\log(n\phi)(n^{16/3}\phi^{8/3})))$  unless a local optimum is reached.*

*Proof.* Using Lemma 19, the probability that the improvement  $\Delta_{\min}$  in a linked 2-Opt step of type 0 or 1 is less than  $\varepsilon$  is at most

$$\Pr(\Delta_{\min} \leq \varepsilon) = O(n^6 \cdot \phi^5 \cdot \varepsilon^2 \cdot \log^2(1/\varepsilon)) + O(n^5 \cdot \phi^4 \cdot \varepsilon^{3/2} \cdot \log(1/\varepsilon)).$$

Following the proof ideas of Theorem 10 on the consecutive 2-Opt steps for Manhattan instances, the expected fitness gain  $E(\Delta)$  for a pair of linked 2-Opt steps of type 0 or 1 can be bounded from below as

$$\begin{aligned} E(\Delta) &\geq (1 - \Pr(\Delta_{\min} \leq \varepsilon)) \cdot \varepsilon \\ &\geq (1 - c_1(n^6 \cdot \phi^5 \cdot \varepsilon^2 \cdot \log^2(1/\varepsilon)) - c_2(n^5 \cdot \phi^4 \cdot \varepsilon^{3/2} \cdot \log(1/\varepsilon))) \cdot \varepsilon. \end{aligned}$$

We set  $\varepsilon = c'/(\log(n\phi) \cdot n^{10/3}\phi^{8/3})$  for a small enough constant  $c' > 0$  such that  $\Pr(\Delta_{\min} > \varepsilon) \geq 1/2$ , justified in the following. When plugging in  $\varepsilon$ ,

$$c_1(n^6 \cdot \phi^5 \cdot \varepsilon^2 \cdot \log^2(1/\varepsilon)) = c'^2 \cdot O(n^{6-20/3}\phi^{5-16/3}) = c'^2 \cdot o(1)$$

since  $\frac{\log^2((\log(n\phi)n^{10/3}\phi^{8/3})/c')}{\log^2(n\phi)} = O(1)$ . Likewise, when plugging in  $\varepsilon$ ,

$$c_2(n^5 \cdot \phi^4 \cdot \varepsilon^{3/2} \cdot \log(1/\varepsilon)) = c'^{3/2} \cdot O(n^0\phi^0) = c'^{3/2} \cdot O(1)$$

since  $\frac{\log((\log(n\phi)n^{10/3}\phi^{8/3})/c')}{\log^{3/2}(n\phi)} = o(1)$ . For both terms, the implicit constants can be made arbitrarily small by choosing  $c'$  small, thus establishing  $\Pr(\Delta_{\min} > \varepsilon) \geq 1/2$ .

This implies

$$E(\Delta) \geq c'/(\log(n\phi) \cdot (2n^{10/3}\phi^{8/3})).$$

Following the same argument in Theorem 10 the expected number of type 0 or 1 pairs in a sequence of  $t$  steps is  $\Omega(t/n^2)$ . A lower bound for the expected fitness gain for  $t$  generations is therefore

$$E(\Delta_t) \geq E(\Delta) \cdot \Omega(t/n^2) = \Omega(t/(\log(n\phi) \cdot (n^{16/3}\phi^{8/3}))).$$

□

Metric	RLS		(1+1) EA/(1+1) EA*	
	Single Step (any $t$ )	Consecutive ( $t \geq \frac{3}{2} \cdot n^3$ )	Single	Con. ( $t \geq 3e^2 \cdot n^3$ )
Man	$\Omega\left(\frac{t}{n^6\phi}\right)$	$\Omega\left(\frac{t}{n^5\phi}\right)$	$\Omega\left(\frac{t}{n^6\phi}\right)$	$\Omega\left(\frac{t}{n^5\phi}\right)$
Euc	$\Omega\left(\frac{t}{\log(n\phi) \cdot n^6\phi^3}\right)$	$\Omega\left(\frac{t}{\log(n\phi) \cdot n^{16/3}\phi^{8/3}}\right)$	$\Omega\left(\frac{t}{\log(n\phi) \cdot n^6\phi^3}\right)$	$\Omega\left(\frac{t}{\log(n\phi) \cdot n^{16/3}\phi^{8/3}}\right)$

Table 2: Expected fitness gain in  $t$  generations for RLS and (1+1) EA for Manhattan and Euclidean instances due to single-step and consecutive-steps analysis. The former applies for any time span  $t$ ; the latter requires  $t = \Omega(n^3)$ . The consecutive-steps analysis was formally proven for the (1+1) EA\* and transfers to the (1+1) EA if, as conjectured, the latter does not perform worse. All fitness gains assume that no local optimum is reached. Otherwise the expected approximation ratio is  $O(\sqrt{\phi})$ . The results for RLS single step, consecutive steps and (1+1) EA single step and consecutive steps are based on Theorems 5, 10, 6 and 12 for Manhattan metric and Theorems 17, 20, 18 and 21 for Euclidean metric respectively.

### 4.3 Analysis of Linked Steps for (1+1) EA

We improve the current results for (1+1) EA with the analysis for consecutive 2-Opt steps in a similar way to the analysis presented in the previous section. Again, we consider the (1+1) EA\* but conjecture that the expected fitness gain in the (1+1) EA is no smaller than that for the (1+1) EA\*. Based on our arguments on the number of type 0 or 1 linked pairs from Lemmas 11 and 8 and the stated Lemma 19 of [9] on the probability of the existence of a pair of both improving steps we can bound the expected fitness gain in  $t$  generations.

**Theorem 21.** *In  $t \geq cn^3$  generations,  $c > 3e^2$  constant, (1+1) EA\* achieves an expected fitness gain of  $\Omega(t/(\log(n\phi) \cdot (n^{16/3}\phi^{8/3})))$  unless a local optimum is reached.*

*Proof.* Following the proof ideas in above Theorem 20 we get for a  $c' > 0$

$$E(\Delta) \geq c'/(\log(n\phi) \cdot (2n^{10/3}\phi^{8/3})).$$

From Lemma 11 we know that the number of disjoint pairs of 2-Opt steps, both of which are singular, is at least

$$u = \frac{t}{3e^2n^2} - \frac{n}{2}.$$

Lemma 7 implies that among these there are at least

$$\frac{u}{2} - \frac{n}{4} = \frac{t}{6e^2n^2} - \frac{n}{2} \geq \frac{(1 - 3e^2/c)t}{6e^2n^2} = \Omega(t/n^2).$$

type 0 or 1 pairs.

The expected fitness gain in  $t$  generations,  $E(\Delta_t)$ , is therefore at least

$$E(\Delta) \cdot \Omega(t/n^2) = \Omega(t/(\log(n\phi) \cdot (n^{16/3}\phi^{8/3}))).$$

□

The lower bounds on the expected fitness gain for RLS, (1+1) EA and (1+1) EA\* are presented in Table 2.

#### 4.4 The Approximation Ratio over the Algorithm Run

We again interpret our fitness gain results in terms of the expected approximation ratio, similar to the previous analysis on Manhattan instances. Here, we refer to the previous study of Burgstaller and Pillichshammer [3] to derive the expected length of the initial tour.

**Theorem 22.** *Based on the single step improvements, for Euclidean instances and for RLS, the expected approximation ratio after  $t$  generations is at most  $151\sqrt{n\phi} - 288c_{17}t/(\log(n\phi)n^{6.5}\phi^{2.5}) + e^{-\Omega(n)}$ , for  $c_{17}$  the hidden constant in the expected fitness gain in Theorem 17, if the current solution is not already locally optimal.*

*Proof.* The proof is essentially the same as that of Theorem 15. For the Euclidean metric  $E(\text{Tour}_0)$  equals  $n$  times the expected Euclidean distance between two points  $P_1(x_1, y_1), P_2(x_2, y_2)$  chosen uniformly at random from the unit square. The latter is derived as  $E(|P_1 - P_2|) = (2 + \sqrt{2} + 5 \log(\sqrt{2} + 1)) \cdot \frac{1}{15}$  [3, Example 3]. This gives  $E(|P_1 - P_2|) \leq 0.522$ . Accordingly,  $E(\text{Tour}_0) \leq 0.522n$ , and using Lemma 14 and Theorem 17 yields an expected approximation ratio of at most

$$\begin{aligned} E(\text{App}_t) &\leq E(\text{Tour}_0 - \Delta_t) \cdot \frac{288\sqrt{\phi}}{\sqrt{n}} + e^{-\Omega(n)} \\ &\leq 151\sqrt{n\phi} - \frac{288c_{17}t}{\log(n\phi) \cdot n^{6.5}\phi^{2.5}} + e^{-\Omega(n)}. \end{aligned}$$

□

Similarly, we can derive upper bounds on the expected approximation ratio after  $t$  generations for the considered algorithms and the Euclidean metric as presented in Table 3.

Algorithm	Euclidean metric	
	Single Steps	Consecutive Steps
RLS	$151\sqrt{n\phi} - \frac{288c_{17}t}{\log(n\phi) \cdot n^{6.5}\phi^{2.5}}$	$151\sqrt{n\phi} - \frac{288c_{20}t}{\log(n\phi) \cdot n^{35/6}\phi^{13/6}}, t \geq 3/2 \cdot n^3$
(1+1) EA/EA*	$151\sqrt{n\phi} - \frac{288c_{18}t}{e \log(n\phi) \cdot n^{6.5}\phi^{2.5}}$	$151\sqrt{n\phi} - \frac{288c_{21}t}{\log(n\phi) \cdot n^{35/6}\phi^{13/6}}, t \geq 3e^2 \cdot n^3$

Table 3: Upper bounds on the expected approximation ratio in  $t$  generations for RLS and (1+1) EA or (1+1) EA\* for Euclidean instances due to single-step and consecutive-steps analysis if the current solution is not locally optimal (The  $e^{-\Omega(n)}$  term is omitted for clarity). Otherwise, the approximation ratio is  $O(\sqrt{\phi})$ . The results are based on the theorems mentioned in the caption of Table 2. The constants  $c_{17}, c_{20}, c_{18}$ , and  $c_{21}$  are the hidden constants in the expected fitness gain from Theorems 17, 20, 18, and 21, respectively.

## 5 Population-based Algorithms

We now extend our considerations to search heuristics with offspring populations and investigate the algorithms (1+ $\lambda$ ) RLS and (1+ $\lambda$ ) EA shown in Algorithms 4 and 5, respectively. Both algorithms generate  $\lambda$  offspring in one generation and replace the current solution  $x$  by an offspring  $y_{\text{best}}$  (individual of best fitness among the  $\lambda$  individuals) if its fitness is not inferior to  $x$ . In the context of TSP the best individual represents the shortest tour. Offspring populations are popular as all offspring can be generated and

evaluated in parallel. In this sense, the number of generations is considered as the parallel runtime of the algorithm. The total number of function evaluations can be derived by multiplying the number of generations by  $\lambda$ .

---

**Algorithm 4**  $(1+\lambda)$  RLS
 

---

```

 $x \leftarrow$  a random permutation of  $n$  cities.
repeat
  for  $i = 1$  to  $\lambda$  do
     $y_i := x$ .
     $y_i \leftarrow$  apply a 2-Opt step chosen uniformly at random to  $y_i$ .
  end for
  Let  $y_{\text{best}}$  be an individual with minimum tour length among  $y_1, \dots, y_\lambda$ .
  if  $f(y_{\text{best}}) \leq f(x)$  then
     $x \leftarrow y_{\text{best}}$ .
  end if
until forever

```

---



---

**Algorithm 5**  $(1+\lambda)$  EA
 

---

```

 $x \leftarrow$  a random permutation of  $n$  cities.
repeat
  for  $i = 1$  to  $\lambda$  do
    draw  $s$  from a Poisson distribution with expectation 1.
     $y_i := x$ .
    for  $s + 1$  times do
       $y_i \leftarrow$  apply a 2-Opt step chosen uniformly at random to  $y_i$ .
    end for
  end for
  Let  $y_{\text{best}}$  be an individual with minimum tour length among  $y_1, \dots, y_\lambda$ .
  if  $f(y_{\text{best}}) \leq f(x)$  then
     $x \leftarrow y_{\text{best}}$ .
  end if
until forever

```

---

### 5.1 Fitness Gain based on Single Step Improvements

In one generation,  $\lambda$  offspring are created and out of the  $(1+\lambda)$  individuals a fittest individual is selected for the next generation. Hence, we can calculate the fitness gain based on the probability that any of these offspring creates a sufficiently large improvement.

We first consider the case for RLS and Manhattan instances. The following lemma estimates the probability that one of  $\lambda$  independent trials will yield a desirable event that happens with some probability  $p$ .

**Lemma 23.** For any  $0 \leq p \leq 1$ , and any  $\lambda \in \mathbb{N}$

$$1 - (1 - p)^\lambda \geq \frac{\lambda p}{1 + \lambda p} \geq \frac{\min(\lambda p, 1)}{2}.$$

*Proof.* The first inequality follows easily from Lemma 8 in [26] which states that  $(1 - x)^n \leq \frac{1}{1 + nx}$ . The second one follows from

$$\frac{\lambda p}{1 + \lambda p} \geq \frac{\lambda p}{2 \max(\lambda p, 1)} = \frac{\min(\lambda p, 1)}{2}.$$

□

The following theorem shows that the lower bound on the expected fitness gain of  $(1+\lambda)$  RLS is by a factor of order  $\min(\lambda, n^2)$  larger than the lower bound on the expected fitness gain of RLS, at the expense of making  $\lambda$  fitness evaluations in one generation.

**Theorem 24.** *For Manhattan instances in  $t$  generations,  $(1+\lambda)$  RLS and  $(1+\lambda)$  EA achieve an expected fitness gain of  $\Omega(t \min(\lambda, n^2) \cdot 1/(n^6 \phi))$  unless a local optimum is reached.*

The minimum in  $\min(\lambda, n^2)$  can be explained as follows. For small offspring population sizes  $\lambda$ , the probability of making a successful 2-Opt step increases linearly with  $\lambda$  (cf. Lemma 23). For  $\lambda = n^2$  this probability is  $\Omega(1)$ , and this asymptotic growth does not increase, even if  $\lambda$  increases beyond  $n^2$ . Hence, increasing the offspring population size beyond  $n^2$  does not lead to a further increase of the lower bound on the fitness gain.

Note that, to see the expected fitness gain *per fitness evaluation*, we may simply divide the expected fitness gain per generation by  $\lambda$ . Then the above factor becomes  $\min(1, n^2/\lambda)$ , that is, our bounds on the expected fitness gain per fitness evaluation do not improve when using offspring populations. This makes sense as an elitist  $(1+1)$  algorithm mutates the best search point seen so far, whereas offspring populations make  $\lambda$  mutations of the previous generation's best search point. The advantage of offspring populations lies in the fact that offspring generation can be parallelized. For that reason and for simplicity, we state the fitness gain with respect to the number of generations.

*Proof of Theorem 24.* From Theorem 5 we can derive the expected fitness gain given that a step is an improving step as

$$E(\Delta) \geq 1/(2304n^4\phi).$$

For each offspring generated in  $(1+\lambda)$  RLS, the probability for an improving 2-Opt step is at least  $1/\binom{n}{2} \geq 2/n^2$  if the current solution is not locally optimal. For each offspring in the  $(1+\lambda)$  EA this probability is at least  $2/(en^2)$  as in Theorem 6. Since we have  $\lambda$  offspring, the probability of the best one making an improving 2-Opt step can be derived according to Lemma 23 for  $p = 2/n^2$  and  $p = 2/(en^2)$ , respectively. In both cases

$$1 - (1 - p)^\lambda \geq \frac{\min(\lambda p, 1)}{2} \geq \Omega(\min(\lambda/n^2, 1)).$$

Therefore, for both algorithms the expected value for the fitness gain  $\Delta$  in any 2-Opt step can be bounded from below as

$$E(\Delta) \geq \Omega(\min(\lambda/n^2, 1)/(2304n^4\phi)) = \Omega(\min(\lambda, n^2)/(n^6\phi)).$$

Hence, the expected value for the fitness gain in  $t$  generations if no locally optimal solution has been obtained in between can be derived as  $E(\Delta_t) \geq \Omega(t \cdot \min(\lambda, n^2)/(n^6\phi))$ .  $\square$

Similarly for Euclidean instances the fitness gains can be derived following Theorem 24, Theorem 17 and Theorem 18.

**Corollary 25.** *For Euclidean instances in  $t$  generations  $(1+\lambda)$  RLS and  $(1+\lambda)$  EA achieve an expected fitness gain of  $\Omega(t \min(\lambda, n^2)/(\log(n\phi) \cdot (n^6\phi^3)))$  unless a locally optimal solution is reached.*

## 5.2 Fitness Gain based on Linked Steps

For  $(1+\lambda)$  RLS the analysis on linked steps follows from the previous analysis of RLS (Theorem 10). Recall that Theorem 10 required  $t = \Omega(n^3)$ . Compared to this, the minimum value for  $t$  for  $(1+\lambda)$  RLS is by a factor of order  $\min(\lambda, n^2)$  smaller.

**Theorem 26.** *For Manhattan instances in  $t \geq cn^3 / \min(2\lambda, n^2)$  generations,  $c > 6$  constant,  $(1+\lambda)$  RLS achieves an expected fitness gain of  $\Omega(t \min(\lambda, n^2) / (n^5 \phi))$  unless a local optimum is reached.*

*Proof.* Let  $\Delta$  be the random variable that describes the fitness gain obtained in a pair of linked 2-Opt steps of type 0 or 1. For a  $c' > 0$ , the expected fitness gain  $E(\Delta)$  can be bounded following Theorem 10.

$$E(\Delta) \geq 1/(c' \sqrt{(n^6 \cdot \phi^2)}) \geq 1/(c' n^3 \phi).$$

The probability  $p$  for an improving 2-Opt step is at least  $1/\binom{n}{2} \geq 2/n^2$  if the current solution is not locally optimal. By substituting this probability  $p$  to Lemma 23 we obtain the probability of improving 2-Opt steps made in one generation as at least  $\min(2\lambda/n^2, 1)/2$ . The expected number of improving 2-Opt steps made in  $t$  generations is therefore  $t \min(2\lambda/n^2, 1)/2$ .

Let  $t^*$  be the number of improving steps. These steps in fact form a sequence of consecutive 2-Opt steps due to RLS accepting only the improving steps. By Lemma 8 we know there are at least  $t^*/6 - n/2$  disjoint type 0 or 1 pairs in a sequence of  $t^*$  consecutive steps. By substituting the value for expected number of improving steps  $t \min(2\lambda/n^2, 1)/2$  for  $t^*$  we obtain the desired result.

$$t^*/6 - n/2 \geq \frac{t \min(2\lambda/n^2, 1)}{12} - \frac{n}{2}.$$

As  $t \geq cn^3 / (\min(2\lambda, n^2))$  this is greater than  $\frac{(1-6/c)t \cdot \min(2\lambda/n^2, 1)}{12}$ . For  $c > 6$ , we get  $\Omega(t \min(\lambda/n^2, 1))$ .

$$t^*/6 - n/2 \geq \frac{t \min(2\lambda/n^2, 1)}{12} - \frac{n}{2} \geq \frac{(1-6/c)t \cdot \min(2\lambda/n^2, 1)}{12} \geq \Omega(t \min(\lambda/n^2, 1)).$$

A lower bound for the expected fitness gain for  $t$  generations is therefore

$$E(\Delta_t) \geq E(\Delta) \cdot \Omega(t \min(\lambda/n^2, 1)) = \Omega(t \min(\lambda, n^2) / (n^5 \phi)).$$

□

For Euclidean instances also the results follows from Theorem 20 in a similar way to Theorem 26.

**Corollary 27.** *For Euclidean instances in  $t \geq cn^3 / \min(2\lambda, n^2)$  generations,  $c > 6$  constant,  $(1+\lambda)$  RLS achieves an expected fitness gain of  $\Omega(t \min(\lambda, n^2) / (\log(n\phi) \cdot (n^{16/3} \phi^{8/3})))$  unless a local optimum is reached.*

Extending our analysis of linked steps to the  $(1+\lambda)$  EA, or the  $(1+\lambda)$  EA\*, turns out to be very challenging. Our analysis of linked steps in the  $(1+1)$  EA\* (Theorems 12 and 21) were based on improving steps made in singular generations, as only singular generations allowed us to form linked pairs of improving 2-Opt steps.



Metric	(1+ $\lambda$ ) RLS		(1+ $\lambda$ ) EA
	Single Step (any $t$ )	Con. ( $t \geq \frac{cn^3}{\min(2\lambda, n^2)}, c > 6$ )	Single Step (any $t$ )
Man	$\Omega\left(\frac{t \min(\lambda, n^2)}{n^6 \phi}\right)$	$\Omega\left(\frac{t \min(\lambda, n^2)}{n^6 \phi}\right)$	$\Omega\left(\frac{t \min(\lambda, n^2)}{n^6 \phi}\right)$
Euc	$\Omega\left(\frac{t \min(\lambda, n^2)}{\log(n\phi) \cdot n^6 \phi^3}\right)$	$\Omega\left(\frac{t \min(\lambda, n^2)}{\log(n\phi) \cdot n^{16/3} \phi^{8/3}}\right)$	$\Omega\left(\frac{t \min(\lambda, n^2)}{\log(n\phi) \cdot n^6 \phi^3}\right)$

Table 4: Expected fitness gain in  $t$  generations for (1+ $\lambda$ ) RLS and (1+ $\lambda$ ) EA for Manhattan and Euclidean instances due to single-step and consecutive-steps analysis if the algorithm is not in a local optimum. The former applies for any time span  $t$ ; the latter requires  $t = \Omega(n^3 / \min(\lambda, n^2))$ . Otherwise (cf. in a local optima) the expected approximation ratio is  $O(\sqrt{\phi})$ . The results for (1+ $\lambda$ ) RLS single step, consecutive steps and (1+ $\lambda$ ) EA single step are based on Theorems 24, 26 for Manhattan metric and Corollaries 25, 27 for Euclidean metric respectively.

The proof of Theorem 12 showed that the next improving step where a particular edge is being removed is made in a singular generation with probability at least  $1/(2e)$ . This argument does not carry over to the (1+ $\lambda$ ) EA: the (1+ $\lambda$ ) EA accepts the best out of  $\lambda$  offspring, and each offspring may be created in a singular or multi-step mutation. If improvements are easy to find, there is a good chance that the best offspring will result from a multi-step mutation.

To demonstrate this effect, assume that there are constants  $0 < \alpha, \beta \leq 1$  such that the fraction of 2-Opt steps increasing the fitness by more than  $\alpha \Delta_{\max}$  is at least  $\beta$ , where  $\Delta_{\max}$  is the maximum fitness increase by any 2-Opt step in the considered metric (Manhattan or Euclidean distances). Such a situation is particularly likely at the start of a run, when the tour contains many long edges. Note that singular mutations yield to a fitness gain of at most  $\Delta_{\max}$ . The probability of one offspring increasing the fitness by more than  $\Delta_{\max}$ , through a mutation making  $\lceil 1/\alpha \rceil$  2-Opt steps, is at least

$$\Pr(S = \lceil 1/\alpha \rceil - 1) \cdot \beta^{\lceil 1/\alpha \rceil} \cdot (1 - o(1)) = \Omega(1)$$

where the  $1 - o(1)$  factor accounts for the fact that the number of 2-Opt steps increasing the fitness by  $\alpha$  may decrease during the application of the  $S + 1$  2-Opt moves.

The probability that no offspring leads to a fitness increase larger than  $\Delta_{\max}$  is then  $2^{-\Omega(\lambda)}$ , hence with probability  $1 - 2^{-\Omega(\lambda)}$  the best offspring has a fitness gain larger than  $\Delta_{\max}$ , which can only happen in a multi-step mutation. So, in this situation with probability at least  $1 - 2^{-\Omega(\lambda)}$  the best offspring will result from a multi-step mutation.

Unless  $\lambda$  is very small, this drastically decreases the chances of finding linked pairs of 2-Opt steps as in the proof of Theorem 12, and the analysis breaks down. Note that this is a shortcoming of our theoretical approach; we still conjecture that the expected fitness gain can be bounded from below by  $\Omega(\min(\lambda, n^2))$  times the lower bound for the (1+1) EA\* from Theorem 12. However, proving this requires novel ideas and possibly a different approach.

The lower bounds on the expected fitness gain for (1+ $\lambda$ ) RLS and (1+ $\lambda$ ) EA are summarized in Table 4.

Above fitness gain results hold for the considered algorithms if they are not in a local optimum. Otherwise, we provide an upper bound on the approximation ratio.

**Theorem 28.** *For  $\phi$ -perturbed Manhattan and Euclidean instances and for (1+ $\lambda$ ) RLS, (1+ $\lambda$ ) EA and (1+ $\lambda$ ) EA\*, the approximation ratio for the worst local optimum is bounded above by  $O(\sqrt{\phi})$ .*

*Proof.* We consider Manhattan and Euclidean instances in a 2 dimensional unit hyper-

cube  $[0, 1]^2$ . Then as a direct consequence from Theorem 3 the expected approximation ratio is at most  $O(\sqrt{\phi})$ .  $\square$

### 5.3 The Approximation Ratio over the Algorithm Run

The results on the expected approximation ratio follow similarly for the population-based algorithms. In Table 1 and Table 3 the subtrahend, which is proportional to the (bound on the) expected fitness gain, is multiplied by a factor of  $\Theta(\min(\lambda, n^2))$ , except for the consecutive steps results for the  $(1+\lambda)$  EA.

## 6 Experiments

This section presents experimental results on the studied algorithms in the above sections. In particular, the three simple algorithms RLS,  $(1+1)$  EA and  $(1+1)$  EA\* and the three population-based variants  $(1+\lambda)$  RLS,  $(1+\lambda)$  EA and  $(1+\lambda)$  EA\* are run on Euclidean TSP instances with  $n = 100$  and  $n = 500$ . Whereas our theoretical results hold for distributions of arbitrary density, in this section we restrict our considerations to the special case of uniform distributions of points.

100 instances are considered for each instance size 100 and 500. Each point is generated uniformly at random and positioned in the  $[0, 1]^2$  unit square. More precisely, we consider the probability distribution with the density functions  $f_i: [0, 1]^d \rightarrow [0, \phi]$  where  $\phi = 1$  and  $d = 2$  for  $i \in \{1, \dots, n\}$ . For every instance size, the algorithms are run on 100 randomly generated instances, each for 1.000.000 generations. For each generation the average fitness over the 100 instances is calculated and the results are displayed in the following graphs.

Our theoretical progress bounds are pessimistic in that they assume that there always exists only one 2-Opt move that improves the fitness. In the early stages of a run, this assumption is too pessimistic as there are many 2-Opt moves that improve an initial solution generated uniformly at random. It is hence not surprising that the tour length drops sharply during the first few generations, as shown in Figure 5.

Note that our theoretical progress bounds are general in that they apply to every non-optimal search point. They allow us to address the question how much the fitness is likely to improve if the algorithm run is continued for  $t$  further generations. We expect that our bounds are more precise in such scenarios, where the number of improving steps is reasonably low.

The rest of the figures therefore focus on the time span after the initial major fitness gain has been made. The theoretical progress bounds are plotted starting from several fitness points. These curves represent the respective expected minimum fitness gain  $(E(\Delta))$  for the considered algorithm. Within the following figures these positions (fitness points) are chosen (except for Figure 5) closer to the end of the major fitness gain, in order to have a zoomed picture into the fitness curves after the major/dramatic fitness gain.

Recall that our theoretical progress bounds assume that the algorithm never encounters a local optimum. However, the experimental results consist of the fitness data achieved by an algorithm at a specified generation regardless of whether its in a local optimum or not. Furthermore,  $(1+1)$  EA and  $(1+\lambda)$  EA have the capability to escape from a local optimum hence these algorithms may gain fitness after being stuck in a local optimum for some time. Therefore, a comparison of the experimental results with theoretical progress bounds should be considered in this light, and used to gain some general insights only.

As shown in Figure 6 (left) for the instance size 100, the simple algorithms perform

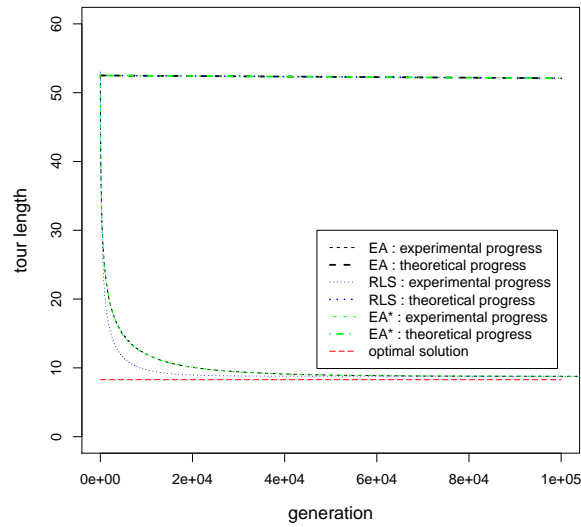


Figure 5: Average tour length values for 100 Euclidean instances with  $n = 100$  in which the points are distributed uniformly at random in a  $[0, 1]^2$  square.

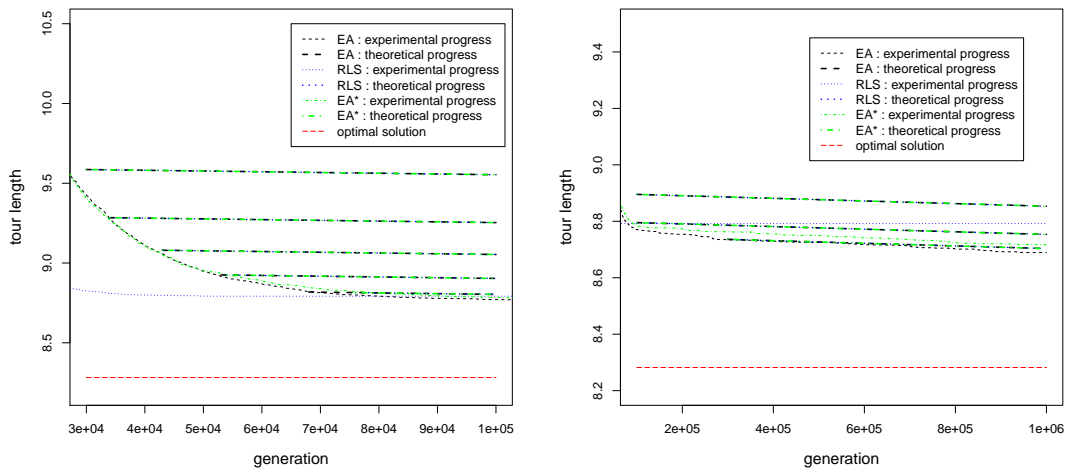


Figure 6: Average tour length values for 100 random Euclidean instances with  $n = 100$ . Left: the range is considered after the major fitness gain achieved; 30.000–100.000 generations. Right: the range is 100.000–1.000.000 generations. Theoretical progress curves represent the respective  $E(\Delta)$  values starting from the specified fitness points.

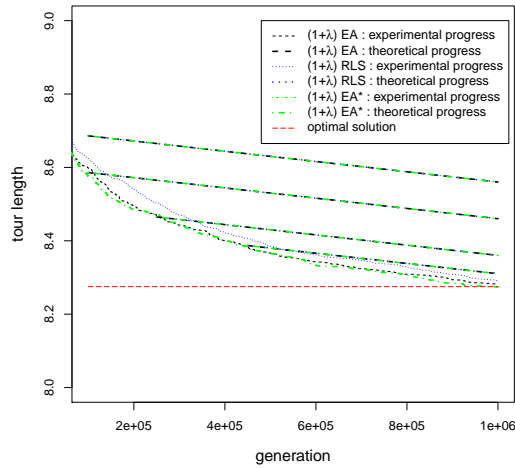


Figure 7: Average tour length values for  $\lambda = 30$  and for 100 random Euclidean instances with  $n = 100$ .

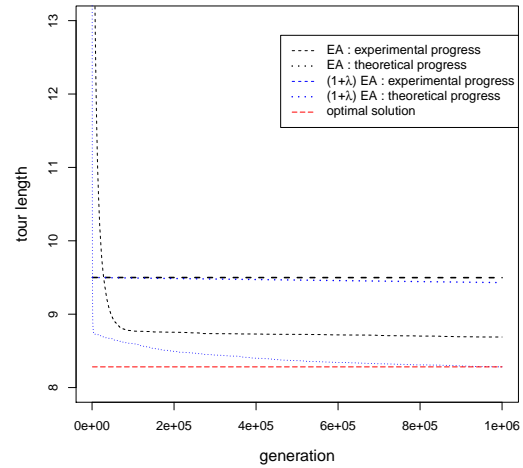


Figure 8: Comparison of (1+1) EA and (1+ $\lambda$ ) EA for  $\lambda = 30$  and for 100 random Euclidean instances with  $n = 100$ .

significantly better than the lower bound on the expected fitness gain. Comparatively, for the considered data set RLS achieves a higher fitness (hence a lower tour length value) within fewer generations than the simple evolutionary algorithms. However, RLS seems to get stuck in a local optimum while evolutionary algorithms may escape. This can be observed in the zoomed version of the fitness gain as shown in Figure 6 (right). Furthermore, the theoretical progress curves align better with the experimental progress in the latter part of the algorithm run (see Figure 6 (right)). The pattern holds for the population-based algorithms although with a greater fitness gain within the given number of generations (see Figure 7 and Figure 8).

Both (1+1) EA and (1+1) EA\* perform almost similar for the considered instances. This supports our conjecture on these two algorithms that they perform similar except for a small number of steps. The results of Wilcoxon signed rank test [30] further supports our claim. The results of the test for instance size 100 is shown in Table 5. A rank test is performed every 1000 generations. For a test, 100 samples representing the tour length values achieved by the two algorithms (1+1) EA and (1+1) EA\* for the 100 TSP instances for a fixed generation are considered. The statistical significance is represented by the confidence values ( $p$ -value). A general observation from the results of the rank test is that none of the tests provide statistically significant results based on a significance level of  $p = 0.01$ . Thus we fail to reject the null hypothesis that the tour length values obtained by the two algorithms are not different. This indicates that the two algorithms perform similarly.

A similar pattern holds for the instance size 500 also (see Figure 9). Comparatively, for each algorithm the number of generations taken to converge is greater for the instance size 500 than for the instance size 100. Due to space limitations we do not include the rank test results for the instance size 500 here. Similar to the rank test for the instance size 100, here also none of the test results are significant for a significance level of  $p = 0.01$ . Thus we fail to reject the null hypothesis that the tour length values

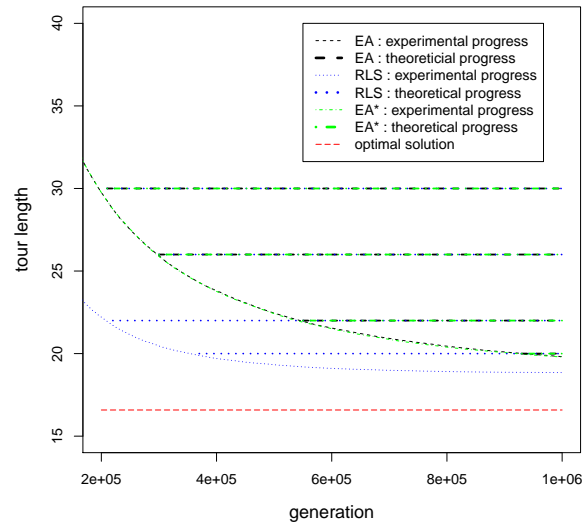


Figure 9: Average tour length values for 100 Euclidean instances with  $n = 500$ .

obtained by the two algorithms are not different. This indicates that the two algorithms perform similarly for instance size 500 also.

## 7 Conclusions

We have carried out a fixed budget analysis of randomized local search and evolutionary algorithms on the well-known Traveling Salesperson Problem (TSP). Our analysis allows to estimate the progress, or fitness gain, that these algorithms make within a given number of  $t$  generations.

This is, in particular, useful as it gives a guarantee to practitioners on the progress that such algorithms can make when deciding between stopping the algorithm or giving it additional running time.

We analyzed the algorithms in the setting of smoothed complexity for the Manhattan and Euclidean metric. First, we provided lower bounds on the expected fitness gain based on the minimum improvement the algorithms RLS and (1+1) EA can make in a generation. The results show that for any number of generations both algorithms gain a fair improvement based on single 2-Opt steps.

We further improved these results by analyzing a sequence of consecutive 2-Opt steps together to identify linked pairs. Table 2 summarizes these results. It is observed that a larger lower bound on the improvement can be obtained considering the consecutive steps, for this however, the number of generations  $t$  needs to be at least  $\Omega(n^3)$ .

The variant of the (1+1) EA ((1+1) EA\*) analyzed for the linked steps accepts fewer solutions than the classical (1+1) EA and therefore we expect (1+1) EA\* to be slower than (1+1) EA. Proving this is an interesting technical open problem and would give additional insights into the advantages of mutations that make multiple changes at the same time.

Further we have extended the analysis for population-based algorithms (1+ $\lambda$ ) RLS

and  $(1+\lambda)$  EA and the results are summarized in Table 4. The analysis based on single steps for both algorithms and the analysis based on consecutive steps for  $(1+\lambda)$  RLS follows from our previous analysis on RLS and  $(1+1)$  EA whereas the consecutive step analysis for  $(1+\lambda)$  EA does not. We have discussed the existing problem with this approach and leave an open problem to find novel methods for this analysis.

Additionally, we have analyzed the approximation ratio during the algorithm run. In particular, we have proved an upper bound on the expected approximation ratio after  $t$  generations for the considered algorithms and metrics.

Moreover, experiments are conducted to observe the fitness gain in a typical algorithm run. In particular, Euclidean instances and the uniform distribution are considered with instance sizes 100 and 500. Results support our conjecture that  $(1+1)$  EA\* performs similar to  $(1+1)$  EA and further indicate an increased fitness gain in population-based variants over simple evolutionary algorithms. It would be interesting to see if the increased fitness gain for the population-based variants is mainly due to the fact that we use the notion of generation or whether there are other considerable factors. Hence, exploring the effects of populations on the fitness gain would be an interesting future research direction.

**Acknowledgments:** This research has been supported by the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no 618091 (SAGE) and by the Australian Research Council through grants DP130104395 and DP140103400. We thank Joseph Kempka and Anh Quang Nguyen for early discussions, and the participants and organizers of Dagstuhl seminar 13271 “Theory of Evolutionary Algorithms”, where this research started.

## References

- [1] Auger, A. and Doerr, B. (2011). *Theory of Randomized Search Heuristics: Foundations and Recent Developments*. World Scientific Publishing Co., Inc.
- [2] Beyer, H.-G. (1992). Some Aspects of the ‘Evolution Strategy’ for Solving TSP-like Optimization Problems. In *Parallel Problem Solving from Nature, 2*, pages 361–370, Amsterdam. Elsevier.
- [3] Burgstaller, B. and Pillichshammer, F. (2009). The average distance between two points. *Bulletin of the Australian Mathematical Society*, 80:353–359.
- [4] Chandra, B., Karloff, H., and Tovey, C. (1994). New results on the old  $k$ -Opt algorithm for the TSP. In *Proceedings of 5th ACM-SIAM Symposium on Discrete Algorithms*, pages 150–159.
- [5] Corus, D., He, J., Jansen, T., Oliveto, P. S., Sudholt, D., and Zarges, C. (2015). On easiest functions for somatic contiguous hypermutations and standard bit mutations. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO ’15*, pages 1399–1406. ACM.
- [6] Doerr, B., Jansen, T., Witt, C., and Zarges, C. (2013). A method to derive fixed budget results from expected optimisation times. In *Proceeding of the fifteenth annual conference on Genetic and evolutionary computation, GECCO ’13*, pages 1581–1588, New York, NY, USA. ACM.
- [7] Eiben, A. E. and Smith, J. E. (2003). *Introduction to Evolutionary Computing*. SpringerVerlag.
- [8] Englert, M., Röglin, H., and Vöcking, B. (2007). Worst case and probabilistic analysis of the 2-Opt algorithm for the TSP. In *In Proceedings of the 18th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1295–1304.
- [9] Englert, M., Röglin, H., and Vöcking, B. (2014). Worst case and probabilistic analysis of the 2-Opt algorithm for the TSP. *Algorithmica*, 68(1):190–264.

- [10] Fischer, T. and Merz, P. (2008). Reducing the size of travelling salesman problem instances by fixing edges. In *Recent Advances in Evolutionary Computation for Combinatorial Optimization*, pages 243–258.
- [11] Jansen, T. (2013). *Analyzing Evolutionary Algorithms - The Computer Science Perspective*. Natural Computing Series. Springer.
- [12] Jansen, T. and Zarges, C. (2011). Analysis of evolutionary algorithms: From computational complexity analysis to algorithm engineering. In *Proceedings of the 11th Workshop Proceedings on Foundations of Genetic Algorithms, FOGA '11*, pages 1–14, New York, NY, USA. ACM.
- [13] Jansen, T. and Zarges, C. (2012). Fixed budget computations: a different perspective on run time analysis. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2012)*, pages 1325–1332.
- [14] Jansen, T. and Zarges, C. (2014a). Evolutionary algorithms and artificial immune systems on a bi-stable dynamic optimisation problem. In *Proceedings of the 2014 Conference on Genetic and Evolutionary Computation, GECCO '14*, pages 975–982, New York, NY, USA. ACM.
- [15] Jansen, T. and Zarges, C. (2014b). Performance analysis of randomised search heuristics operating with a fixed budget. *Theoretical Computer Science*, 545:39 – 58.
- [16] Jansen, T. and Zarges, C. (2014c). Reevaluating immune-inspired hypermutations using the fixed budget perspective. *Evolutionary Computation, IEEE Transactions on*, 18(5):674–688.
- [17] Kern, W. (1989). A probabilistic analysis of the switching algorithm for the Euclidean TSP. *Mathematical programming: Series A*, 44(1-3):213–219.
- [18] Lengler, J. and Spooner, N. (2015). Fixed budget performance of the (1+1)-EA on linear function. In *Proceedings of the 11th Workshop on Foundations of Genetic Algorithms, FOGA '15*, pages 52–61. ACM.
- [19] Manthey, B. and Veenstra, R. (2013). Smoothed analysis of the 2-Opt heuristic for the TSP: Polynomial bounds for Gaussian noise. In *Proceedings of the 24th international Symposium on Algorithms and Computation (ISAAC '13)*. Springer.
- [20] Merz, P. and Huhse, J. (2008). An iterated local search approach for finding provably good solutions for very large TSP instances. In *Proceedings of the 10th International Conference on Parallel Problem Solving from Nature (PPSN X)*, pages 929–939.
- [21] Motwani, R. and Raghavan, P. (1995). *Randomized Algorithms*. Cambridge University Press, New York, NY, USA.
- [22] Nallaperuma, S., Neumann, F., and Sudholt, D. (2014). A fixed budget analysis of randomized search heuristics for the traveling salesperson problem. In *Genetic and Evolutionary Computation Conference, GECCO '14, Vancouver, BC, Canada, July 12-16, 2014*, pages 807–814. ACM.
- [23] Neumann, F. and Witt, C. (2010). *Bioinspired Computation in Combinatorial Optimization: Algorithms and Their Computational Complexity*. Springer.
- [24] Nürnberg, H.-T. and Beyer, H.-G. (1997). The Dynamics of Evolution Strategies in the Optimization of Traveling Salesman Problems. In *Evolutionary Programming VI: Proceedings of the Sixth Annual Conference on Evolutionary Programming*, pages 349–359, Heidelberg. Springer-Verlag.
- [25] Oliveto, P. S., He, J., and Yao, X. (2007). Time complexity of evolutionary algorithms for combinatorial optimization: A decade of results. *International Journal of Automation and Computing*, 4(3):281–293.
- [26] Rowe, J. E. and Sudholt, D. (2014). The choice of the offspring population size in the  $(1, \lambda)$  evolutionary algorithm. *Theoretical Computer Science*, 545:20–38.

- [27] Sanders, P. and Wagner, D. (2013). Algorithm engineering. *Informatik-Spektrum*, 36(2):187–190.
- [28] Spielman, D. A. and Teng, S.-H. (2004). Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *J. ACM*, 51(3):385–463.
- [29] Tao, G. and Michalewicz, Z. (1998). Inver-over operator for the TSP. In *Parallel Problem Solving from Nature PPSN V*, volume 1498 of *Lecture Notes in Computer Science*, pages 803–812. Springer Berlin Heidelberg.
- [30] Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics Bulletin*, 1(6):80–83.
- [31] Zhou, D., Luo, D., Lu, R., and Han, Z. (2012). The use of tail inequalities on the probable computational time of randomized search heuristics. *Theoretical Computer Science*, 436:106 – 117.



Expected Fitness Gains of RSH for the TSP

generation	test 1		test 2	
	W	p-value	W	p-value
1000	2155	0.8987	2895	0.1020
2000	2317	0.7633	2733	0.2378
3000	2913	0.0914	2137	0.9092
4000	2851	0.1315	2199	0.8692
5000	2882	0.1101	2168	0.8905
6000	3061	0.0327	1988	0.9676
7000	3003	0.0503	2047	0.9500
8000	2717	0.2551	2333	0.7460
9000	2604	0.3936	2446	0.6077
10000	2709	0.2640	2341	0.7371
11000	2666	0.3145	2384	0.6867
12000	2486	0.5540	2564	0.4473
13000	2418	0.6442	2632	0.3571
14000	2353	0.7234	2697	0.2777
15000	2280	0.8002	2769	0.2007
16000	2501	0.5329	2548	0.4685
17000	2574	0.4338	2476	0.5676
18000	2740	0.2304	2310	0.7706
19000	2682	0.2947	2367	0.7065
20000	2678	0.3000	2372	0.7012
21000	2633	0.3552	2416	0.6461
22000	2713	0.2596	2337	0.7415
23000	2600	0.3989	2450	0.6024
24000	2579	0.4270	2471	0.5743
25000	2634	0.3546	2416	0.6467
26000	2743	0.2268	2306	0.7743
27000	2782	0.1889	2268	0.8120
28000	2737	0.2335	2313	0.7675
29000	2772	0.1983	2278	0.8026
30000	2797	0.1753	2253	0.8256
31000	2822	0.1540	2228	0.8468
32000	2740	0.2304	2310	0.7706
33000	2794	0.1780	2256	0.8229
34000	2849	0.1330	2201	0.8677
35000	2704	0.2697	2346	0.7314
36000	2622	0.3700	2428	0.6313
37000	2668	0.3121	2382	0.6891
38000	2744	0.2262	2306	0.7748
39000	2731	0.2399	2319	0.7612
40000	2718	0.2540	2332	0.7471
41000	2655	0.3281	2395	0.6732
42000	2691	0.2847	2359	0.7165
43000	2690	0.2858	2360	0.7153
44000	2728	0.2431	2322	0.7579
45000	2718	0.2540	2332	0.7471
46000	2748	0.2221	2302	0.7789
47000	2768	0.2022	2282	0.7988
48000	2797	0.1753	2253	0.8256
49000	2791	0.1807	2259	0.8202
50000	2769	0.2012	2281	0.7997

generation	test1		test2	
	W	p-value	W	p-value
51000	2783	0.1880	2267	0.8129
52000	2753	0.2170	2297	0.7840
53000	2746	0.2242	2304	0.7768
54000	2699	0.2754	2351	0.7257
55000	2730	0.2410	2320	0.7601
56000	2697	0.2777	2353	0.7234
57000	2727	0.2442	2323	0.7569
58000	2780	0.1908	2270	0.8102
59000	2766	0.2041	2284	0.7968
60000	2759	0.2110	2291	0.7900
61000	2778	0.1926	2272	0.8083
62000	2745	0.2252	2305	0.7758
63000	2724	0.2475	2326	0.7536
64000	2739	0.2314	2311	0.7696
65000	2717	0.2551	2333	0.7460
66000	2727	0.2442	2323	0.7569
67000	2747	0.2232	2303	0.7779
68000	2700	0.2743	2350	0.7269
69000	2693	0.2823	2357	0.7188
70000	2682	0.2953	2368	0.7059
71000	2667	0.3133	2383	0.6879
72000	2676	0.3024	2374	0.6988
73000	2648	0.3368	2402	0.6644
74000	2654	0.3293	2396	0.6719
75000	2620	0.3726	2430	0.6287
76000	2626	0.3648	2424	0.6365
77000	2637	0.3507	2413	0.6506
78000	2655	0.3281	2395	0.6732
79000	2660	0.3219	2390	0.6794
80000	2659	0.3231	2391	0.6781
81000	2676	0.3024	2374	0.6988
82000	2675	0.3036	2375	0.6976
83000	2674	0.3048	2376	0.6964
84000	2642	0.3444	2408	0.6569
85000	2653	0.3306	2397	0.6707
86000	2648	0.3368	2402	0.6644
87000	2661	0.3206	2389	0.6806
88000	2662	0.3194	2388	0.6818
89000	2665	0.3157	2385	0.6855
90000	2666	0.3145	2384	0.6867
91000	2681	0.2964	2369	0.7047
92000	2687	0.2893	2363	0.7118
93000	2683	0.2941	2367	0.7071
94000	2697	0.2777	2353	0.7234
95000	2702	0.2720	2348	0.7292
96000	2703	0.2708	2347	0.7303
97000	2717	0.2551	2333	0.7460
98000	2710	0.2629	2340	0.7382
99000	2697	0.2777	2353	0.7234
100000	2686	0.2905	2364	0.7107

Table 5: Results of Wilcoxon signed rank tests for the tour length values obtained by the algorithms (1+1) EA and (1+1) EA\* for  $n = 100$ . (1+1) EA > (1+1) EA\* (*test1*), (1+1) EA\* > (1+1) EA (*test2*), positive rank sums (*W*) and confidence (*p*) values are displayed accordingly. The row *i* represents the test results based on 100 samples of tour length values (from the 100 TSP instances) for the generation.