Evolutionary Algorithms and Matroid Optimization Problems ⋆

Joachim Reichel[1], **Martin Skutella**[2]

[1] TU Berlin, e-mail: reichel@math.tu-berlin.de
[2] TU Berlin, e-mail: skutella@math.tu-berlin.de

Received: date / Revised version: date

Abstract We analyze the performance of evolutionary algorithms on various matroid optimization problems that encompass a vast number of efficiently solvable as well as NP-hard combinatorial optimization problems (including many well-known examples such as minimum spanning tree and maximum bipartite matching). We obtain very promising bounds on the expected running time and quality of the computed solution. Our results establish a better theoretical understanding of why randomized search heuristics yield empirically good results for many real-world optimization problems.

Key words evolutionary algorithms, matroids, minimum weight basis, matroid intersection, randomized search heuristics

1 Introduction

Motivation. While evolutionary algorithms are known to work (empirically) well for many optimization problems in practice, a satisfying and rigorous mathematical analysis of their performance is one of the main challenges in the area of genetic and evolutionary computing. Interesting results have been obtained for some important, isolated optimization problems but a general theoretical explanation of the behavior of evolutionary algorithms is still missing. The aim of this article is to make progress in this direction. We study the performance of evolutionary algorithms on a very general class of combinatorial optimization problems and obtain promising results on the running time and quality of the computed solutions. A summary of

⋆ This work was supported by the Deutsche Forschungsgemeinschaft (DFG) as part of the Collaborative Research Center "Computational Intelligence" (SFB 531).

problem	efficient algorithm known	exp. # iterations of (1+1) EA and RLS	lower bound on exp. # iterations of (1+1) EA and RLS
minimum weight basis	$O(\|E\| \log \|E\| + \|E\|\Theta)$	$O(\|E\|^2(\log r(E) + \log w_{\max}))$	$\Omega(\|E\|^2 \log r(E))$ [25]
unweighted matroid intersection	$O(\|E\|\, r(E)^{1.5}\Theta)$ [3]	$(1-\varepsilon)$-approximation in $O(\|E\|^{2\lceil 1/\varepsilon \rceil})$	exponential [14]
weighted matroid intersection	$O(\|E\|\, r(E)(r(E)\Theta + \log \|E\|))$ [31]	1/2-approximation in $O(\|E\|^4(\log \|E\| + \log w_{\max}))$	exponential [14]
intersection of $p \geq 3$ matroids	NP-hard [13]	$1/p$-approximation in $O(\|E\|^{p+2}(\log \|E\| + \log w_{\max}))$	exponential [14]

Table 1 A summary of results on various matroid optimization problems. The problems are described in the first column. The second column gives the running time of known efficient algorithms, where Θ is the (maximum) time complexity of the independence oracle(s). The third column describes the results for (1+1) EA and (modifications of) RLS obtained in this article. The last column gives lower bounds on the running time of (1+1) EA and RLS for obtaining an *optimal* solution. Note that the numbers in the two last columns denote the expected number of iterations rather than the total runtime.

our results together with an overview of the complexity of the considered optimization problems can be found in Table 1.

Independence Systems and Matroids. Matroid theory provides a framework in which a substantial class of problems in combinatorial optimization can be studied from a unified perspective. Matroids form a special class of *independence systems* that are given by a finite set E and a family of subsets $\mathcal{F} \subseteq 2^E$ such that $\mathcal{F}$ is closed under subsets. The subsets contained in $\mathcal{F}$ are called *independent* and a maximal independent subset is called a *basis* of the independence system. A precise definition of matroids along with some important classical results is given in Section 2.

Many combinatorial optimization problems can be formulated as follows: Given an independence system on a weighted set E, find a basis of minimum (or maximum) weight. We mention as an example the problem of finding a stable set of maximum weight in a given graph with weights on the nodes (here, the underlying independence system is *not* a matroid).

A famous result by Rado [27], Gale [12], and Edmonds [7] states that an independence system is a matroid if and only if the greedy algorithm computes a minimum weight basis for arbitrary weights on the elements of E. This algorithmic characterization of matroids highlights their relevance in the area of efficient algorithms and combinatorial optimization.

Examples of Matroids and First Results. *Graphic matroids* are an important example of matroids where E is the edge set of a graph $G = (V, E)$ and a subset of edges is independent if it does not contain a circuit. If the given graph G is connected, the problem of finding a minimum weight basis of the corresponding graphic matroid is the minimum span-

ning tree problem. In the context of evolutionary algorithms, RAIDL and JULSTROM [28] analyze different encodings for the minimum spanning tree problem and propose to work with so-called *edge sets*. NEUMANN and WEGENER [25] study evolutionary algorithms for the minimum spanning tree problem. They prove that two specific evolutionary algorithms, (1+1) EA and RLS, compute a minimum spanning tree in expected polynomial time, i.e., $O(|E|^2(\log|V| + \log w_{\max}))$, where $w_{\max}$ denotes the maximum weight of any edge; moreover, they establish a lower bound of $\Omega(|E|^2 \log|V|)$. In Section 3 we generalize this result to arbitrary matroids (see also Table 1). After establishing a crucial property of the problem our analysis follows that of [25].

Another important example of matroids are *linear matroids* where E is a set of vectors and a subset is independent if the vectors contained in it are linearly independent. Linear matroids occur, for example, in the minimum cycle basis problem where the task is to find a minimum weight basis of the cycle space of a given graph with weights on the edges. This problem is an important building block in various real-world optimization problems such as, for example, in electrical networks, structural engineering, chemistry and biochemistry, and in periodic timetabling; see, e.g., [23] for details.

Matroid Intersection. Matroids have even more algorithmic power than just that of the greedy method. EDMONDS [6] (see also [22]) observed that also the *(weighted) matroid intersection problem* can be solved efficiently. That is, a maximum weight common independent set in two matroids can be found in strongly polynomial time. The matroid intersection problem has applications in many settings such as, for example, edge connectivity [10], survivable network design [2], constrained [18] as well as degree-bounded [16] minimum spanning trees, and multicast network codes [17].

One of the most prominent examples of an optimization problem that can be formulated as a matroid intersection problem is the maximum weight matching problem in bipartite graphs. GIEL and WEGENER [14,15] analyze evolutionary algorithms for the maximum matching problem (with unit weights). They show that (1+1) EA and RLS are (randomized) polynomial-time approximation schemes with an expected runtime of $O(|E|^{2\lceil 1/\epsilon \rceil})$; moreover they construct a class of bipartite graphs for which these algorithms need an exponential expected running time until they find an optimal solution. Motivated by these results we prove in Section 4 that (1+1) EA and RLS are polynomial-time approximation schemes for the matroid intersection problem with unit weights (see also Table 1). Again, establishing a crucial property of the problem structure allows to extend the previous analysis for the special case to our problem.

Our result for the unweighted matroid intersection problem cannot be generalized easily to the weighted case and also from the viewpoint of efficient algorithms it is known that the weighted version of the problem is somewhat harder than the unweighted problem. In Section 5 we prove that (1+1) EA and a slightly modified version of RLS are $\frac{1}{2}$-approximation algorithms for the weighted matroid intersection problem (see also Table 1). The

techniques used to obtain this result differ from those for the unweighted case and we are not aware of previous work for (other) special cases.

NP-Hard Problems. The problem of finding a maximum-size common independent set in three or more matroids is NP-hard as finding a Hamiltonian circuit in a directed graph is a special case [19]. On the other hand it is known that any independence system can be represented as an intersection of finitely many matroids and a vast number of combinatorial optimization problem falls into this category. In Section 6 we prove that (1+1) EA and (an appropriately modified version of) RLS are $\frac{1}{p}$-approximation algorithms for the NP-hard problem to find a maximum-weight independent set in the intersection of $p \geq 3$ matroids (see also Table 1). This result is a generalization of our $\frac{1}{2}$-approximation for the weighted intersection of two matroids. Note that the approximation ratio of $\frac{1}{p}$ is the same as that of the greedy algorithm [20,21].

More Results from the Literature. In the following we mention some further related results from the literature. Wegener [33,34] discusses randomized search heuristics as an alternative to exact algorithms in the context of discrete optimization problems and gives an overview of known results. Droste, Jansen and Wegener [5] study the behavior of (1+1) EA on pseudo-boolean functions. Sorting and shortest path problems are considered by Scharnow, Tinnefeld, and Wegener [30].

2 Preliminaries

Evolutionary algorithms are a class of randomized search heuristics that are inspired by biological evolution. Informally, their basic structure can be explained as follows. An element of the search space is referred to as *individual* and a set of such individuals is called *population*. The quality of an individual is measured by a *fitness function*. Following the idea of *survival of the fittest*, evolutionary algorithms try to maximize the fitness of the individuals in the population in an iterative way. Such an iteration is called *generation*. To this end, some individuals (*parents*) of the current population are chosen as seed for the next generation. By applying recombination and/or mutation operators, new individuals (*offspring*) are generated. The recombination operator is applied to two individuals and creates a new individual by recombining the information of the parents. The mutation operator is applied to one individual and creates one new individual. The offspring compete with the individuals of the current generation for a place in the population of the next generation. See [1] or [8] for a detailed introduction into the field of evolutionary algorithms.

We study the behavior of two very simple evolutionary algorithms, namely (1+1) EA and randomized local search (RLS), which is sometimes also called *local* (1+1) EA. These algorithms play an important role since their simplicity allows a theoretical analysis of their behavior. On the other hand, they

are complex enough to solve important combinatorial optimization problems. The population of both algorithms consists of exactly one individual, and there is no recombination operator. Almost all theoretical results for combinatorial optimization problems consider evolutionary algorithms without recombination operator (also called *crossover*). This is due to the fact that proofs become much more difficult if such an operator is introduced. It was an open question whether recombination is provably helpful for non-artificial problems. This question has recently been answered positively for the all-pairs shortest path problem [4] and the Ising model [9,32].

For both algorithms, (1+1) EA and RLS, the search space consists of all bitstrings of a fixed length. Initially, a bitstring $s \in \{0,1\}^n$ is chosen randomly (whenever we speak of randomness, we mean uniform randomness). The mutation operators of (1+1) EA and RLS are defined as follows:

(1+1) EA: Obtain the bitstring s' by flipping each bit of s independently of the other bits with probability $1/n$.

RLS: Choose $b \in \{0,1\}$ randomly. If $b = 0$, choose $i \in \{1,\dots,n\}$ randomly and obtain s' by flipping the i-th bit of s. If $b = 1$, choose $(i,j) \in \{(k,l) \mid 1 \le k < l \le n\}$ randomly and obtain s' by flipping the i-th and j-th bit of s.

Note that the mutation operator of RLS is capable of flipping two bits simultaneously. We shall see later that it is crucial that the neighborhood of a given search point contains other search points with a Hamming distance of two. The current search point s is replaced by s' if the fitness value of s' is better than or equal to that of s. Note that "better" corresponds to either "smaller" or "larger", depending on whether a minimization or maximization problem is considered.

Both algorithms do not use any stopping criteria. For theoretical investigations it is common to consider the algorithms as infinite stochastic processes and to consider the number of fitness evaluations as a measure of the runtime. Note that in (1+1) EA and RLS there is exactly one fitness evaluation per generation, thus the number of fitness evaluations equals the number of generations. Hence, our goal is to bound the expected number of generations (also called expected runtime) until the algorithms have discovered a desired search point.

We shall now give the definition of matroids as well as some well-known properties used in this article. See [22], [26], or [31], for a more detailed discussion.

Definition 1 *Let E be a finite set and $\mathcal{F} \subseteq 2^E$. The pair $M = (E,\mathcal{F})$ is called a* matroid *if*
(i) $\emptyset \in \mathcal{F}$,
(ii) $\forall X \subseteq Y \in \mathcal{F}: X \in \mathcal{F}$, *and*
(iii) $\forall X,Y \in \mathcal{F}, |X| > |Y| : \exists x \in X \setminus Y$ *with* $Y \cup \{x\} \in \mathcal{F}$.

The elements of $\mathcal{F}$ are called *independent*, the elements of $2^E \setminus \mathcal{F}$ are called *dependent*. The maximal independent sets are called *bases of* M, the minimal dependent sets are called *circuits*. For $X \subseteq E$, a maximal independent

subset of X is called a *basis of* X. The *rank* $r(X)$ *of* X is the maximal cardinality of a basis of X, $r(X) := \max\{|Y| \mid Y \subseteq X, Y \in \mathcal{F}\}$.

Proposition 1 *Let $(E, \mathcal{F})$ be a matroid. Then*
(i) for $X \subseteq E$, all bases of X have the same cardinality;
(ii) for all bases B_1, B_2 of M, $x \in B_1 \setminus B_2$ there exists $y \in B_2 \setminus B_1$ such that $(B_1 \setminus \{x\}) \cup \{y\}$ is a basis of M.

Proposition 2 *Let $(E, \mathcal{F})$ be a matroid and $r(\cdot)$ its rank function. Then*
(i) $r(\emptyset) = 0$,
(ii) $\forall X \subseteq E : r(X) \leq |X|$,
(iii) $\forall X \subseteq E : r(X) = |X| \Leftrightarrow X \in \mathcal{F}$,
(iv) $\forall X, Y \subseteq E, X \subseteq Y : r(X) \leq r(Y)$.

For $X \in \mathcal{F}$ and $y \in E$ such that $X \cup \{y\} \notin \mathcal{F}$, we use $C(X, y)$ to denote the unique circuit in $X \cup \{y\}$. If $X \cup \{y\} \in \mathcal{F}$, we define $C(X, y) := \emptyset$.

Note that $|\mathcal{F}|$ might be exponential in $|E|$. Hence, $|\mathcal{F}|$ is usually not given explicitly. Instead one usually assumes the existence of an appropriate oracle. Such an oracle can be understood as a black box that reveals some information about $|\mathcal{F}|$. For example, an independence oracle decides in constant time for a given set $X \subseteq E$ whether $X \in \mathcal{F}$ holds. Here we assume that the set $\mathcal{F}$ is implicitly given by a rank oracle which for any set $X \subseteq E$ computes its rank $r(X)$ in constant time. Note that a rank oracle is polynomially equivalent to an independence oracle [22].

3 Minimum Weight Basis

The results presented in this section generalize and are motivated by the corresponding results for the minimum spanning tree problem in [25]. We consider the following problem. Given a matroid $M = (E, \mathcal{F})$ and a weight function $w : E \to \mathbb{N}$, find a basis $B \subseteq E$ of M of minimum weight. The weight of a subset of E is defined as the sum of the weights of its elements. The weight of an optimal solution is denoted by w_{OPT}.

The search space equals $S = \{0,1\}^{|E|}$, where each position of the bitstring corresponds to an element from $E = \{e_1, \ldots, e_{|E|}\}$. A search point $s \in \{0,1\}^{|E|}$ corresponds to the subset $E(s) := \{e_i \in E \mid s_i = 1, 1 \leq i \leq |E|\}$ of E. We define the weight $w(s)$ of a bitstring s as the weight of the corresponding set $E(s)$. In a similar way we define the rank $r(s)$ of a bitstring s as $r(s) := r(E(s))$.

We consider two fitness functions f and f'. Let $w_{\max}$ denote the maximum weight of any element in E. Then $w_{ub} := |E| \cdot w_{\max}$ is an upper bound on the weight of any subset of E. Let

$$f(s) := (r(E) - r(s)) \cdot |E| \cdot w_{ub} + (|E(s)| - r(E)) \cdot w_{ub} + \sum_{e \in E(s)} w(e)$$

be the first fitness function which is to be minimized. The expression is dominated by the first term which encodes the number $r(E) - r(s)$ of elements that have to be added to $E(s)$ to get a superset of a basis. If $E(s)$ is a superset of a basis, this term vanishes and the second term dominates the expression. The factor $|E(s)| - r(E)$ denotes the number of elements that have to be removed from $E(s)$ to get a basis of M. If $E(s)$ is a basis of M, both the first and second term vanish and the fitness of s corresponds to the weight of the elements in $E(s)$.

The second term in the fitness function f explicitly penalizes circuits. This is not necessary, since the removal of an element of a circuit leads to a weight decrease by itself. Therefore, we also investigate the fitness function

$$f'(s) := (r(E) - r(s)) \cdot w_{ub} + \sum_{e \in E(s)} w(e).$$

Note that $f'(s)$ equals $f(s)$ if s describes a spanning tree. We shall see that the additional information concerning $|E(s)|$ in the first fitness function f allows us to obtain better bounds.

The remainder of this section is structured as follows. After a simple inequality to bound probabilities we prove two propositions concerning the expected number of generations until a superset of a basis and a basis have been constructed. Then we present several propositions about basis transition properties which are needed to prove the main theorems of this section.

Proposition 3 *Let $\alpha - 1 \geq \beta \geq 1$ and $\gamma \geq 1$. Then*

$$\left(1 - \frac{1}{\alpha}\right)^{\beta} \cdot \left(1 - \left(1 - \frac{1}{\alpha}\right)^{\gamma}\right) \geq e^{-1} \cdot \frac{\gamma}{\alpha + \gamma}.$$

Proof Since $\beta \leq \alpha - 1$, we get

$$\left(1 - \frac{1}{\alpha}\right)^{\beta} \geq \left(1 - \frac{1}{\alpha}\right)^{\alpha - 1} \geq e^{-1}.$$

Since $1 - x \leq e^{-x}$ for all $x \in \mathbb{R}$, we have

$$1 - \left(1 - \frac{1}{\alpha}\right)^{\gamma} \geq 1 - e^{-\frac{\gamma}{\alpha}} = 1 - \frac{1}{e^{\gamma/\alpha}}.$$

Again, since $e^x \geq 1 + x$ for all $x \in \mathbb{R}$, we obtain

$$1 - \frac{1}{e^{\gamma/\alpha}} \geq 1 - \frac{1}{1 + \gamma/\alpha} = \frac{\gamma}{\alpha + \gamma},$$

which concludes the proof.

Proposition 4 *The expected number of generations until RLS or (1+1) EA working on one of the fitness functions f or f' constructs a superset of a basis of M is bounded by $O(|E| \log r(E))$.*

Proof Suppose the initial search point s does not describe a superset of a basis. Then $r(s) < r(E)$ holds. Both fitness functions f and f' are defined in such a way that the rank of $E(s)$ will never decrease in accepted steps. For each subset $X \subseteq E$, there are at least $r(E) - r(X)$ elements of E whose inclusion increases the rank of X by 1.

The probability that RLS performs a 1-bit flip is $\frac{1}{2}$ and there are $|E|$ possible 1-bit flips. Therefore, for RLS, the probability that a step increases the rank of $E(s)$ is at least $\frac{1}{2} \cdot \frac{r(E)-r(s)}{|E|}$. For (1+1) EA we consider the event that $r(s)$ elements of a fixed basis of $E(s)$ remain unchanged and at least one of $r(E) - r(s)$ elements that enlarge this independent subset is flipped. The probability of this event is given by

$$\left(1 - |E|^{-1}\right)^{r(s)} \cdot \left(1 - \left(1 - |E|^{-1}\right)^{r(E)-r(s)}\right),$$

which, by Proposition 3, can be lower bounded by $\frac{1}{2e} \cdot \frac{r(E)-r(s)}{|E|}$. Hence, the expected number of generations until s describes a superset of some basis is bounded from above by

$$\sum_{i=0}^{r(E)-1} \frac{2e|E|}{r(E)-i} = O(|E| \log r(E)).$$

This concludes the proof. □

Proposition 5 *The expected number of generations until RLS or (1+1) EA working on the fitness function f constructs a basis of M starting from a superset of a basis is bounded by $O(|E| \log |E|)$.*

Proof Suppose the initial search point s describes a proper superset of some basis of M. Then $|E(s)| > r(E)$ holds. The fitness function f is defined in such a way that only supersets of bases are accepted, i.e., the rank of $E(s)$ does not change. Furthermore, the cardinality of $E(s)$ never increases.

The probability that a step decreases the cardinality of $E(s)$ while maintaining a superset of a basis is at least $\frac{1}{2} \cdot \frac{|E(s)|-r(E)}{|E|}$ for RLS. For (1+1) EA we consider the event that $r(E)$ elements of a fixed basis of M contained in $E(s)$ remain unchanged and at least one of the remaining $|E(s)| - r(E)$ elements is flipped. The probability of this event is given by

$$\left(1 - |E|^{-1}\right)^{r(E)} \cdot \left(1 - \left(1 - |E|^{-1}\right)^{|E(s)|-r(E)}\right),$$

which, by Proposition 3, can be lower bounded by $\frac{1}{2e} \cdot \frac{|E(s)|-r(E)}{|E|}$. Hence, the expected number of generations until s describes a basis of M is bounded from above by

$$\sum_{i=r(E)+1}^{|E|} \frac{2e|E|}{i-r(E)} = O(|E| \log |E|).$$

This concludes the proof. □

We remark that Proposition 5 holds also for RLS and the fitness function f', but not for (1+1) EA and f'. Since RLS flips at most two bits per step, an increase in $|E(s)|$ implies an increasing weight. Consequently, steps increasing $|E(s)|$ are not accepted. This argument does not hold for the (1+1) EA, which might exclude a heavy element and include two or more light elements instead while maintaining a superset of a basis.

The following proposition will later turn out to be useful in order to prove a bound on the number of steps needed to get from an arbitrary basis to a minimum weight basis.

For a matroid $M = (E, \mathcal{F})$ and two sets $A, B \in \mathcal{F}$ we define the graph $G_{A,B} := (V_{A,B}, E_{A,B})$, $V_{A,B} := A \triangle B$, and $E_{A,B} := \{(a, b) \mid a \in A \setminus B, b \in C(B, a) \setminus A\}$.

Proposition 6 *Given a matroid $M = (E, \mathcal{F})$ and two sets $A, B \in \mathcal{F}$ such that $C(B, a) \neq \emptyset$ for all $a \in A \setminus B$ holds. The graph $G_{A,B}$ has a bipartite matching covering $A \setminus B$.*

Proof Note that $G_{A,B}$ is a bipartite graph with bipartition $V_{A,B} = (A \setminus B) \dot\cup (B \setminus A)$. For $X \subseteq A \setminus B$ define $N(X) := \{b \in B \setminus A \mid \exists\, x \in X : (x, b) \in E_{A,B}\}$. We show $|N(X)| \geq |X|$ for all $X \subseteq A \setminus B$. Then the claim follows by the classical Theorem of Hall (see, e.g., [22]).

Suppose there exists $X \subseteq A \setminus B$ with $|N(X)| < |X|$. By Definition 1(ii), the sets X and $N(X)$ are independent (and disjoint) since they are subsets of $A \setminus B$ and $B \setminus A$, respectively. Moreover, the sets $X \dot\cup (A \cap B)$ and $N(X) \dot\cup (A \cap B)$ are independent as subsets of A and B, respectively, and $|N(X) \dot\cup (A \cap B)| < |X \dot\cup (A \cap B)|$ holds. Hence, by Definition 1, there exists $x \in (X \dot\cup (A \cap B)) \setminus (N(X) \dot\cup (A \cap B)) = X \setminus N(X) = X$ such that $N(X) \dot\cup (A \cap B) \dot\cup \{x\} \in \mathcal{F}$. On the other hand, by the definition of $N(\cdot)$, the set $N(X) \dot\cup (A \cap B) \dot\cup \{x\}$ contains the cycle $C(B, x)$. □

Proposition 7 *Let s be a search point describing a non-minimum weight basis B of M. Then there exists some $k \in \{1, \ldots, r(E)\}$ and k different accepted 2-bit flips such that the average weight decrease of these flips is $(w(s) - w_{OPT})/k$.*

Proof Let B^* denote a minimum weight basis of M and define $k := |B^* \setminus B|$. By applying Proposition 6 to B and B^* we obtain an injective function $\alpha : B^* \setminus B \to B \setminus B^*$ such that $\alpha(e) \in C(B, e)$ for all $e \in B^* \setminus B$. By Proposition 1(i), all bases of a matroid have equal cardinality, hence, the function α is a bijection.

By Proposition 1(ii), $B \cup \{e\} \setminus \{\alpha(e)\}$ is again a basis of M. Furthermore, since B^* is an optimal basis $w(e) \leq w(\alpha(e))$ holds for all $e \in B^* \setminus B$. Hence, exchanging e and $\alpha(e)$ does not increase the total weight and the 2-bit flip involving e and $\alpha(e)$ is accepted. All k 2-bit flips together change B into B^* and the total weight decrease is $w(s) - w_{OPT}$. Hence, the average weight decrease is $(w(s) - w_{OPT})/k$. □

The analysis performed later can be simplified if the parameter k in Proposition 7 is independent of the search point s. This can be easily accomplished by allowing non-accepted 2-bit flips whose weight decrease is defined as 0. We add $r(E) - k$ non-accepted 2-bit flips to the k 2-bit flips from Proposition 7.

Proposition 8 *Let s be a search point describing a basis B of M. Then there exists a set of $r(E)$ 2-bit flips such that the average weight decrease of these flips is $(w(s) - w_{OPT})/r(E)$.*

Since Proposition 5 does not hold for the fitness function f' in combination with (1+1) EA, we need a result similar to Proposition 8 for supersets of a basis. Since we start from supersets of a basis, we need to allow also 1-bit flips to reach a basis.

Proposition 9 *Let s be a search point describing a superset of a basis. Then there exists a set of $|E| - r(E)$ 1-bit flips and a set of $r(E)$ 2-bit flips such that the average weight decrease is $(w(s) - w_{OPT})/|E|$.*

Proof Let $B' \subseteq E(s)$ denote a basis of M and s' the corresponding search point. Consider the set of $|E(s)| - r(E)$ 1-bit flips corresponding to the elements in $E(s) \setminus B'$. Their removal from $E(s)$ does not change the rank of $E(s)$, hence the 1-bit flips are accepted. We obtain the basis B' and apply Proposition 8. Alltogether, we obtain a weight decrease of $w(s) - w_{OPT}$ and performed $|E(s)|$ flips.

Similar to Proposition 8 we allow non-accepted 1-bit flips whose weight decrease is defined as 0. By adding $|E| - |E(s)|$ non-accepted 1-bit flips we obtain the claimed result. □

Theorem 1 *The expected number of generations until RLS or (1+1) EA working on the fitness function f constructs a minimum weight basis is bounded by $O(|E|^2(\log r(E) + \log w_{\max}))$.*

Proof By Propositions 4 and 5, it is sufficient to consider the search process after having found a search point s describing a basis. Then, by Proposition 8, there exists a set of $r(E)$ 2-bit flips whose average weight decrease is $(w(s) - w_{OPT})/r(E)$. The choice of such a 2-bit flip is called a *good step*. The probability of performing such a good step equals $\Theta(r(E)/|E|^2)$ and each of the good steps is chosen with the same probability. A good step decreases the difference between the weight of the current search point s and w_{OPT} on average by a factor of $1/r(E)$. This holds independently of previous good steps. Hence, after N good steps, the expected difference between $w(s)$ and w_{OPT} is given by $(1-1/r(E))^N \cdot (w(s) - w_{OPT})$. Since $w(s) \leq r(E) \cdot w_{\max}$ and $w_{OPT} \geq 0$, we obtain the upper bound $(1-1/r(E))^N \cdot D$ on the expected difference between $w(s)$ and w_{OPT} after N good steps, where $D := r(E) \cdot w_{\max}$.

If $N := \lceil (\ln 2) \cdot r(E) \cdot (\log 2D) \rceil$, this bound is at most $\frac{1}{2}$. Since the difference is not negative, by Markov's inequality, the probability that the bound is less than 1 is at least 1/2. The difference is an integer implying

that the probability of having found a minimum weight basis is at least 1/2. Therefore, the expected number of good steps until a minimum weight basis is found is bounded by $2N = O(r(E) \log D) = O(r(E)(\log r(E) + \log w_{\max}))$.

By our construction, there are always exactly $r(E)$ good steps. Therefore, the probability of a good step does not depend on the current search point. Hence, the expected number of generations until l good steps have been made equals $\Theta(l|E|^2/r(E))$. Altogether, the expected number of iterations is bounded by

$$O(N|E|^2/r(E)) = O(|E|^2(\log r(E) + \log w_{\max})).$$

This concludes the proof. □

A slightly worse bound can be shown for the fitness function f' by applying Proposition 9 instead of Proposition 8.

Theorem 2 *The expected number of generations until RLS or (1+1) EA working on the fitness function f' constructs a minimum weight basis is bounded by $O(|E|^2(\log |E| + \log w_{\max}))$.*

Proof By Proposition 4, it is sufficient to consider the search process after having found a search point s describing a superset of a basis. Then, by Proposition 9, for each step there exists a set of $|E| - r(E)$ 1-bit flips and a set of $r(E)$ 2-bit flips such that the total weight decrease is $w(s) - w_{OPT}$. The choice of such a bit flip is called a *good step*. If the total weight decrease of the 1-bit flips is larger than the total weight decrease of the 2-bit flips, the step is called a 1-step. Otherwise, it is called a 2-step. Note that the notion of a 2-step does not imply that we actually perform a 2-bit flip, similarly for 1-steps and 1-bit flips.

Consider the sequence of all steps until a minimum weight basis is reached. Suppose that at least half of the required steps are 2-steps. Consider only these 2-steps. Since there are $r(E)$ good 2-bit flips, the probability of a good 2-bit flip equals $\Theta(r(E)/|E|^2)$. The expected weight decrease of such a 2-bit flip in a 2-step is at least $\frac{1}{2}(w(s) - w_{OPT})/r(E)$. Thus, a good 2-bit flip in a 2-step decreases the difference between the weight of the current search point s and w_{OPT} on average by a factor not larger than $1 - 1/(2r(E))$. Hence, we can adapt the proof of Theorem 1 with $N' := \lceil (\ln 2) \cdot 2r(E) \cdot (\log 2D') \rceil$, $D' := |E| \cdot w_{\max}$ and obtain the bound $O(N'|E|^2/r(E))$ for the expected number of 2-steps. Since the majority of all steps are 2-steps, the claimed result follows.

Now suppose that at least half of the required steps are 1-steps and consider only these 1-steps. Since there are $k := |E| - r(E)$ good 1-bit flips, the probability of a good 1-bit flip equals $\Theta(k/|E|)$. The expected weight decrease of such a 1-bit flip in a 1-step is at least $\frac{1}{2}(w(s) - w_{OPT})/k$. Thus, a good 1-bit flip in a 1-step decreases the difference between the weight of the current search point s and w_{OPT} on average by a factor not larger than $1 - 1/(2k)$. Again, we can apply the proof technique of Theorem 1 where

$N'' := \lceil 2 \cdot (\ln 2) \cdot k \cdot (\log 2D') \rceil = O(|E|(\log |E| + \log w_{\max}))$ takes the role of N. We obtain the upper bound

$$O(N''|E|/k) = O(|E|(\log |E| + \log w_{\max}))$$

for the expected number of 1-steps. Since the majority of all steps are 1-steps, the total number of steps is of the same order, which is even smaller than the proposed bound. □

Neumann and Wegener [25] show a lower bound of $\Omega(|E|^2 \log r(E))$ for a special class of instances of the minimum spanning tree problem. Hence, $\Omega(|E|^2 \log r(E))$ is also a lower bound for the minimum weight basis problem.

We briefly want to mention the benefits of more problem-specific mutation operators. After having found a basis of a matroid, we are only interested in offspring with the same cardinality. The probability of such an offspring can be increased using the following two mutation operators: If RLS flips two bits, it chooses randomly a 0-bit and randomly a 1-bit. If s contains k 1-bits, (1+1) EA flips each 1-bit with probability $1/k$ and each 0-bit with probability $1/(|E| - k)$.

Using the modified mutation operators, the probability of a specific element exchange for bases increases from $\Theta(1/|E|^2)$ to $\Theta(r(E)^{-1}(|E| - r(E))^{-1})$. Therefore, the bound of Theorem 1 can be replaced by $O(r(E)|E| \cdot (\log r(E) + \log w_{\max}) + |E| \log |E|)$. In the case of Theorem 2 we obtain the bound $O(r(E)|E|(\log |E| + \log w_{\max}))$.

The expected number of generations can be further reduced by using parallel versions of (1+1) EA and RLS. The term *parallel* refers to the fact that in each iteration (1+λ) EA and λ-PRLS (λ-parallel RLS) independently produce λ offspring from the currently considered individual. The selection procedure selects an individual with the smallest fitness value among the parent and its offspring. In the proofs of Theorem 1 and Theorem 2 the probability of a good step is $O(r(E)/|E|^2)$. Choosing $\lambda := \lceil |E|^2/r(E) \rceil$, this probability is increased to a positive constant. As before, the expected number of good steps is bounded by $O(r(E)(\log r(E) + \log w_{\max}))$ and $O(r(E)(\log |E| + \log w_{\max}))$, respectively. This leads to the following result.

Theorem 3 *The expected number of generations until λ-PRLS or (1+λ) EA with $\lambda := \lceil |E|^2/r(E) \rceil$ children constructs a minimum weight basis is bounded by $O(r(E) \log w_{\max} + |E| \log |E|)$.*

Using the modified mutation operator mentioned above, the probability of a good step reduces to $O(1/|E|)$ and we obtain the same bound on the expected number of generations as in Theorem 3 already for $\lambda := |E|$.

The bounds of Theorems 1 and 2 depend on the maximum weight $w_{\max}$. For RLS we can actually proof the following strongly polynomial bound.

Theorem 4 *The expected number of generations until RLS working on one of the fitness functions f or f' constructs a minimum weight basis is bounded by $O(|E|^2(\log |E|))$.*

Proof We construct a different set of weights as follows. Sort the weights w_i non-decreasingly. Define the weight $w'_i, 1 \leq i \leq m$ as the rank of w_i in the sorted sequence (with the convention that equal weights have equal rank). Now for all $i, j \in \{1, \ldots m\}$ it holds that $w'_i - w'_j \leq 0$ if and only if $w_i - w_j \leq 0$. This implies that the behavior of RLS does not change if the old weights w_i are replaced by the new weights w'_i. Since $w'_{\max} \leq |E|$, the claimed bound follows from Theorems 1 and 2. □

Neumann and Wegener [24] also consider multi-objective optimization techniques for the (single-objective) minimum spanning tree problem. They study the behavior of two multi-objective evolutionary algorithms called SEMO and GSEMO. The analysis can be carried over to the minimum weight basis problem for matroids and results in a bound of $O(r(E)|E|(r(E) + \log |E| + \log w_{\max}))$ on the number of generations.

Raidl, Koller and Julstrom [29] study specialized edge selection strategies for the minimum spanning tree problem. For average case instances they are able to reduce the total expected runtime excluding preprocessing from $O(|V|^3 \log |V|)$ to $O(|V|^{3/2} \log |V|)$. It seems likely that some results can be extended to the minimum weight basis problem.

4 Matroid Intersection

The results presented in this section are motivated by the results for the maximum matching problem in [14]. We consider the *matroid intersection problem* which is defined as follows. Given two matroids $M_1 = (E, \mathcal{F}_1)$ and $M_2 = (E, \mathcal{F}_2)$ on the same ground set E by their independence oracles, compute a set $X \in \mathcal{F}_1 \cap \mathcal{F}_2$ such that $|X|$ is maximum. Let OPT denote such an optimal element of $\mathcal{F}_1 \cap \mathcal{F}_2$. The well-known matroid intersection algorithm by Edmonds [6,22] starts with $X := \emptyset$. In each iteration, it searches a shortest S_X-T_X-path in an auxiliary graph G_X. This so-called *augmenting path* gives rise to $X' \in \mathcal{F}_1 \cap \mathcal{F}_2$ with $|X'| = |X| + 1$. The algorithm terminates if there is no augmenting path.

The auxiliary graph G_X for $X \in \mathcal{F}_1 \cap \mathcal{F}_2$ is defined as follows. Its node set is E, the edges are given by $A_X \cup B_X$ with $A_X := \{(x, y) \mid y \in E \setminus X,\ x \in C_1(X, y) \setminus \{y\}\}$ and $B_X := \{(y, x) \mid y \in E \setminus X,\ x \in C_2(X, y) \setminus \{y\}\}$. We set $S_X := \{y \in E \setminus X \mid X \cup \{y\} \in \mathcal{F}_1\}$ and $T_X := \{y \in E \setminus X \mid X \cup \{y\} \in \mathcal{F}_2\}$.

Let the node sequence $y_0, x_1, y_1, \ldots, x_n, y_n$ denote any shortest S_X-T_X-path and define $X' := X \setminus \{x_1, \ldots, x_n\} \cup \{y_0, y_1, \ldots, y_n\}$. Then this path is an augmenting path, i.e., $X' \in \mathcal{F}_1 \cup \mathcal{F}_2$ and $|X'| = |X| + 1$. In the EA setting, such an augmentation step corresponds to simultaneously flipping exactly the elements corresponding to the nodes of the augmenting path.

The above algorithm solves the matroid intersection problem in $O(|E|^3\theta)$ time, where θ is the maximum complexity of both independence oracles. An improved algorithm due to Cunningham [3] requires $O(|E|r(E)^{3/2}\theta)$ time. Gabow and Xu [11] consider the special case of linear matroids.

Their algorithms require $O(|E|r(E)^{1.62})$ time in the unweighted case and $O(|E|r(E)^{1.77}(\log r(E) + \log w_{\max}))$ time in the weighted case.

We study the performance of evolutionary algorithms for the matroid intersection problem. We assume that we are given rank oracles r_1 and r_2 that compute for any set $X \subseteq E$ its rank with respect to M_1 and M_2, respectively. Again, we consider the RLS and (1+1) EA algorithm. We consider the fitness function

$$f(s) := -\Phi(s) \cdot |E| + |E(s)|,$$

where $\Phi(s) := 2|E(s)| - r_1(E(s)) - r_2(E(s))$. The expression is dominated by $\Phi(s)$, which measures the infeasibility of $E(s)$. If $E(s)$ is a common independent set, this first term vanishes and the fitness of s equals the cardinality of $E(s)$.

A more precise way to measure the infeasibility of $E(s)$ is to replace $\Phi(s)$ by $\Psi(s) := \min\{|X||X \subseteq E, E(s) \setminus X \in \mathcal{F}_1 \cap \mathcal{F}_2\}|$. However, $\Psi(s)$ cannot be easily computed. Hence, we resort to $\Phi(s)$. Note that $\frac{1}{2}\Phi(s) \leq \Psi(s) \leq \Phi(s) \leq 2|E|$ holds for all $s \in \{0,1\}^{|E|}$.

First, we consider the phase until a common independent set has been constructed. Note that the empty set is a trivial common independent set. Hence, the first phase can also be skipped entirely.

Proposition 10 *The expected number of generations until RLS or (1+1) EA working on the fitness function f constructs a common independent set is bounded by $O(|E| \log |E|)$.*

Proof Suppose $E(s) \notin \mathcal{F}_1 \cap \mathcal{F}_2$ holds for the initial search point s. The fitness function f is defined in such a way that the infeasibility $\Phi(s)$ never increases. There are at least $\frac{1}{2}\Phi(s)$ elements in $E(s)$ whose individual removal from $E(s)$ decreases the infeasibility $\Phi(s)$. The probability that a step decreases the infeasibility $\Phi(s)$ is at least $\frac{1}{2} \cdot \frac{\Phi(s)}{2|E|}$ for RLS. For (1+1) EA we consider the event that the complement of $E(s)$ remains fixed and at least one of $\frac{1}{2}\Phi(s)$ infeasible elements is removed from $E(s)$. The probability of this event is given by

$$(1 - |E|^{-1})^{|E \setminus E(s)|} \cdot (1 - (1 - |E|^{-1})^{\Phi(s)/2}),$$

which, by Proposition 3, can be lower bounded by $\frac{1}{2e} \cdot \frac{\Phi(s)}{2|E|}$. Hence, the expected number of generations until s describes an element in $\mathcal{F}_1 \cap \mathcal{F}_2$ is bounded from above by

$$\sum_{i=1}^{\Phi(s)} \frac{4e|E|}{i} = O(|E| \log |E|).$$

This concludes the proof. □

Next, we consider the search process after having found a common independent set $X \in \mathcal{F}_1 \cap \mathcal{F}_2$. We show that the length of a shortest S_X-T_X-path in G_X can be bounded in terms of $|X|$ and $|OPT|$.

Proposition 11 *Let $\epsilon > 0$ and $X \in \mathcal{F}_1 \cap \mathcal{F}_2$ such that $|X| < (1-\epsilon)|OPT|$. There exists an S_X-T_X-path in G_X with length at most $2\lceil 1/\epsilon \rceil - 2$.*

Proof It was shown by CUNNINGHAM [3] that G_X contains $k := |OPT| - |X|$ vertex-disjoint S_X-T_X-paths. Hence, there exists an S_X-T_X-path with length at most $2\lfloor |X|/k \rfloor$. Since $|X| < (1-\epsilon)\,|OPT|$, we have $|X|/k < (1-\epsilon)/\epsilon < 1/\epsilon$. Thus, the length of this path is bounded by $2\lceil 1/\epsilon \rceil - 2$. □

The bound on the length of an augmenting path allows us to lower bound the probability that RLS or (1+1) EA finds such an augmenting path. These bounds lead to upper bounds on the expected number of generations until $|E(s)|$ is increased, and finally, until an $(1-\epsilon)$-approximation is constructed.

Theorem 5 *For $\epsilon > 0$, the expected number of generations until RLS or (1+1) EA working on the fitness function f constructs an $(1-\epsilon)$-approximation of a maximum element of $\mathcal{F}_1 \cap \mathcal{F}_2$ is bounded by $O(|E|^{2\lceil 1/\epsilon \rceil})$.*

Proof By Proposition 10, it suffices to consider the search process after having found a search point s with $E(s) \in \mathcal{F}_1 \cap \mathcal{F}_2$. The fitness function f is designed such that only steps leading to search points s' describing common independent sets of at least the same cardinality as s are accepted. Assume that $|E(s)| < (1-\epsilon)|OPT|$.

By Proposition 11, there exists an augmenting path in $G_{E(s)}$ of length at most $l := 2\lceil 1/\epsilon \rceil - 2$. The (1+1) EA flips exactly the $l+1$ elements corresponding to the nodes of this path with probability $\Omega(|E|^{-l-1})$. The RLS algorithm needs $l/2$ 2-bit flips shortening the augmenting path and a final 1-bit flip to increase $|E(s)|$. The probability that this happens within the next $l/2+1$ steps is bounded from below by $\Omega((|E|^{-2})^{l/2} \cdot |E|^{-1}) = \Omega(|E|^{-l-1})$. Hence, the expected number of generations to improve $|E(s)|$ is bounded by $O(|E|^{l+1})$ for (1+1) EA and by $O(l \cdot |E|^{l+1})$ for RLS. A more careful analysis for RLS yields the bound $O(|E|^{l+1})$ (see [14]). Since $|OPT| \leq |E|$, the expected number of generations until RLS or (1+1) EA constructs an $(1-\epsilon)$-approximation of a maximum element of $\mathcal{F}_1 \cap \mathcal{F}_2$ is bounded by $O(|E| \log |E|) + O(|E|) \cdot O(|E|^{l+1}) = O(|E|^{2\lceil 1/\epsilon \rceil})$. □

GIEL and WEGENER [14] have shown that RLS and (1+1) EA require an exponential expected running time to find an optimal solution for certain bipartite maximum matching problems. Since bipartite matching is a special case of matroid intersection, we know that the matroid intersection problem cannot be solved by RLS or (1+1) EA in polynomial time.

5 Weighted Matroid Intersection

In the *weighted matroid intersection problem* we additionally consider a weight function $w : E \to \mathbb{N}$ which assigns a non-negative weight $w(e)$ to each element $e \in E$ of the ground set. The task is to compute a common

independent set $X \in \mathcal{F}_1 \cap \mathcal{F}_2$ such that its weight $w(X) := \sum_{e \in X} w(e)$ is maximum.

Similar to the unweighted case we consider the fitness function

$$f(s) := -\Phi(s) \cdot w_{ub} + \sum_{e \in E(s)} w(e),$$

where $w_{ub} := |E| \cdot w_{\max}$ is an upper bound on the weight of any subset of E.

Note that the RLS algorithm is not suited for the weighted matroid intersection problem since, in general, simultaneous flips of more than two bits are required. In the unweighted case, a long augmenting path can be broken into a series of 2-bit flips maintaining the fitness value and one final 1-bit flip. In the weighted case, there are simple examples of paths of length three where such a decomposition into a sequence of 2-bit flips with non-negative difference of the fitness value does not exist. Consider two matroids over $E = \{a, b, c\}$ given by their bases sets $\mathcal{B}_1 = \{\{a, b\}, \{a, c\}\}$ and $\mathcal{B}_2 = \{\{a, c\}, \{b, c\}\}$. Let $w(a) = w(c) = 2$, $w(b) = 3$, and $X = \{b\}$. Then X and $E \setminus X$ are common independent sets with $w(X) = 3$ and $w(E \setminus X) = 4$. The unique S_X-T_X-path in G_X is a-b-c. Any 1- or 2-bit flip decreases the weight or leads to a set that is dependent in at least one of the matroids.

Although bit flips of at most two bits are not sufficient in the weighted case, it is possible to obtain a $\frac{1}{2}$-approximation using only bit flips of at most three bits. We restrict ourselves in the following to this special setting and analyze the expected number of generations to obtain a $\frac{1}{2}$-approximation.

Proposition 12 *Let s be a search point such that its fitness value $f(s)$ cannot be improved by flipping at most three bits. Then $w(s) \geq \frac{1}{2} w_{OPT}$ holds.*

Proof Define $A := OPT$ and $B := E(s)$. Since the fitness value $f(s)$ cannot be improved by flipping one or two bits of s, we have $C_1(B, a) \neq \emptyset$ and $C_2(B, a) \neq \emptyset$ for all $a \in A \setminus B$. Define $X := A \setminus B = \{x_1, \ldots x_k\}$.

By Proposition 6, there exist sets $Y' = \{y'_1, \ldots, y'_k\} \subseteq B \setminus A$ and $Y'' = \{y''_1, \ldots, y''_k\} \subseteq B \setminus A$ such that $y'_i \in C_1(B, x_i)$ and $y''_i \in C_2(B, x_i)$ for $1 \leq i \leq k$. Since the fitness value $f(s)$ cannot be improved by flipping at most three bits of s, we have $w(x_i) \leq w(y'_i) + w(y''_i)$ for $1 \leq i \leq k$. Summing up these inequalities yield $w(A \setminus B) = w(X) \leq w(Y') + w(Y'') \leq 2\, w(B \setminus A)$. Hence, $w(s) \geq \frac{1}{2} w_{OPT}$ holds. □

In order to analyze the expected number of generations until a $\frac{1}{2}$-approximate solution is found we prove that there exists at least one bit flip with a certain weight increase.

Proposition 13 *Let s be a search point such that $w(s) \leq (\frac{1}{2} - \epsilon)\, w_{OPT}$ holds for some $\epsilon > 0$. Then there exists an accepted bit flip involving at most three bits with a weight increase of at least $\frac{2\epsilon}{|E|} w_{OPT}$.*

Proof Define $A := OPT$, $B := E(s)$ and $X := A \setminus B = \{x_1, \ldots, x_k\}$. We use the index sets I' and I'' to denote those elements of X whose addition to B causes a cycle in the matroid M_1 and M_2, respectively.

$$I' := \{i \mid 1 \leq i \leq k,\, C_1(B, x_i) \neq \emptyset\}$$
$$I'' := \{i \mid 1 \leq i \leq k,\, C_2(B, x_i) \neq \emptyset\}$$

By Proposition 6, there exist a set $Y' = \{y'_i \mid i \in I'\} \subseteq B \setminus A$ such that $y'_i \in C_1(B, x_i)$ for all $i \in I'$. Likewise, there exists a set $Y'' = \{y''_i \mid i \in I''\} \subseteq B \setminus A$ such that $y''_i \in C_2(B, x_i)$ for all $i \in I''$. We define the weights w_i, w'_i and w''_i for $1 \leq i \leq k$ as follows:

$$w_i := w(x_i)$$
$$w'_i := \begin{cases} w(y'_i) & \text{if } i \in I' \\ 0 & \text{otherwise} \end{cases}$$
$$w''_i := \begin{cases} w(y''_i) & \text{if } i \in I'' \\ 0 & \text{otherwise} \end{cases}$$

By assumption, $w(B) \leq (\frac{1}{2} - \epsilon)\, w(A)$ holds. Hence, we have $w(A \setminus B) - w(B \setminus A) \geq (\frac{1}{2} + \epsilon)\, w(A)$. Since $w(B \setminus A) \leq w(B) \leq (\frac{1}{2} - \epsilon)\, w(A)$, it follows that $w(A \setminus B) - 2w(B \setminus A) \geq 2\epsilon\, w(A)$. We have

$$\begin{aligned} \sum_{i=1}^{k} w_i - w'_i - w''_i &= w(X) - w(Y') - w(Y'') \\ &\geq w(A \setminus B) - 2w(B \setminus A) \geq 2\epsilon\, w(A) \end{aligned}$$

Hence, there exists an $i \in \{1, \ldots, k\}$ such that $w_i - w'_i - w''_i \geq \frac{2\epsilon}{k} w(A)$. Consider the bit flip that adds the element x_i and removes the elements y'_i and y''_i if $i \in I'$ and $i \in I''$, respectively (note that y'_i and y''_i might be identical). This bit flip involves at most three bits and has a weight increase of at least $\frac{2\epsilon}{|E|} w(A)$. By construction, the resulting bit string encodes a common independent set and the bit flip is accepted. □

Now we can prove our main result, the expected number of generations for a $\frac{1}{2}$-approximation of the weighted matroid intersection problem.

Theorem 6 *The expected number of generations until* (1+1) *EA working on the fitness function f constructs a $\frac{1}{2}$-approximation of a maximum weight element of $\mathcal{F}_1 \cap \mathcal{F}_2$ is bounded by $O(|E|^4(\log r + \log w_{\max}))$, where $r := \min\{r_1(E), r_2(E)\}$.*

Proof By Proposition 10 (which also holds for the weighted case), it suffices to consider the search process after having found a search point s with $E(s) \in \mathcal{F}_1 \cap \mathcal{F}_2$. The fitness function f is designed such that only steps leading to search points s' that describe common independent sets of at least the same weight as s are accepted.

Now consider any search point s with $E(s) \in \mathcal{F}_1 \cap \mathcal{F}_2$ and $w(s) < \frac{1}{2} w_{OPT}$. Define $\epsilon := \frac{1}{2} - \frac{w(s)}{w_{OPT}}$, i.e., $w(s) = (\frac{1}{2} - \epsilon)\, w_{OPT}$ holds. By Proposition 13 there exists an accepted bit flip involving at most three bits with a weight increase of at least $\frac{2\epsilon}{|E|} w_{OPT}$. Such a step is called a *good step*. A good step decreases the difference $\epsilon \cdot w_{OPT}$ between the weight $w(s)$ of the current search point s and $\frac{1}{2} w_{OPT}$ on average by a factor of $2/|E|$. Hence, after N good steps, the expected difference between $w(s)$ and $\frac{1}{2} w_{OPT}$ is given by $(1 - 2/|E|)^N \cdot (\frac{1}{2} w_{OPT} - w(s))$. Since $w_{OPT} \le r \cdot w_{\max}$ and $w(s) \ge 0$, we obtain the upper bound $(1 - 2/|E|)^N \cdot D$, where $D := \frac{1}{2} r \cdot w_{\max}$.

If $N := \lceil (\ln 2) \cdot \frac{|E|}{2} \cdot \log(3D) \rceil$, this bound is at most $\frac{1}{3}$. The difference is half-integral which implies that we have actually reached a $\frac{1}{2}$-approximation after at most N good steps. The probability of a good step is bounded from below by $\Omega(|E|^{-3})$. Hence, the expected number of generations for N good steps is bounded by

$$O(N|E|^3) = O(|E|^4 (\log r + \log w_{\max})).$$

This concludes the proof. □

Consider the following modification of the RLS algorithm. Choose $b \in \{0, 1, 2\}$ randomly. If $b < 2$ proceed as before. Otherwise, choose $(i, j, k) \in \{(a, b, c) \mid 1 \le a < b < c \le |E|\}$ randomly and flip the i-th, j-th and k-th bit of s. We call this algorithm RLS3.

Since we restrict ourselves to bit flips involving at most three bits, all good steps that are accepted by the (1+1) EA can also be achieved using RLS3. Moreover, the probability of a particular bit flip is again bounded from below by $\Omega(|E|^{-3})$. Hence, Theorem 6 does not only hold for (1+1)EA, but also for RLS3.

6 Intersection of three or more matroids

Furthermore, the result of Theorem 6 can be easily generalized to the intersection of p matroids $M_i = (E, \mathcal{F}_i)$, $1 \le i \le p$. The task is to compute an independent set $X \in \bigcap_{i=1}^{p} \mathcal{F}_i$ with maximum weight. This problem is NP-hard for $p \ge 3$, as finding a Hamiltonian circuit in a directed graph is a special case; see [19].

Similar to the previous case of $p = 2$, there are situations in which simultaneous flips of at least $p + 1$ bits are required. Therefore, we do not consider the RLS algorithm in this section. A modification of the RLS algorithm similar to that described in the last paragraphs of the preceding section is still possible though.

The definition of the function $\Phi(s)$ is adjusted as follows. Let

$$\Phi(s) := p\,|E(s)| - \sum_{i=1}^{p} r_i(E(s)) \text{ for all } s \in \{0,1\}^{|E|}.$$

The bound of Proposition 10 increases to $O(p|E|\log|E|)$. The results of Proposition 13 carry over to the intersection of p matroids, although the achieved approximation ratio is worse.

Proposition 14 *Let s be a search point such that $w(s) \leq (\frac{1}{p}-\epsilon)\, w_{OPT}$ holds for some $\epsilon > 0$. Then there exists an accepted bit flip involving at most $p+1$ bits with a weight increase of at least $\frac{p\,\epsilon}{|E|} w_{OPT}$.*

The lower bound for the probability of picking a particular bit flip of at most $p+1$ bits reduces to $\Omega(|E|^{-p-1})$. This observation leads to the following generalization of Theorem 6.

Theorem 7 *Given p matroids $M_i = (E, \mathcal{F}_i)$, $1 \leq i \leq p$, the expected number of generations until $(1+1)\,EA$ working on the fitness function f constructs a $\frac{1}{p}$-approximation of a maximum weight element of $\bigcap_{1\leq i\leq p} \mathcal{F}_i$ is bounded by $O(|E|^{p+2}(\log r + \log w_{\max}))$, where $r := \min\{r_i(E) \mid 1 \leq i \leq p\}$.*

Similar to the minimum weight basis problem we can use parallel versions of (1+1) EA and RLS to reduce the number of generations. Choosing the number of offspring per generation as $\lambda := |E|^{p+1}$ improves the probability of a good step from $\Omega(|E|^{-p-1})$ to a positive constant. As before, the expected number of good steps is bounded by $O(|E|(\log r + \log w_{\max}))$. This leads to the following result.

Corollary 1 *Given p matroids $M_i = (E, \mathcal{F}_i)$, $1 \leq i \leq p$, the expected number of generations until (1+λ) EA with $\lambda := |E|^{p+1}$ children working on the fitness function f constructs a $\frac{1}{p}$-approximation of a maximum weight element of $\bigcap_{1\leq i\leq p} \mathcal{F}_i$ is bounded by $O(|E|(\log r + \log w_{\max}))$.*

7 Conclusion

We have analyzed the performance of (1+1) EA and RLS on a very general class of combinatorial optimization problems ranging from very simple problems that can be solved optimally by the greedy method up to NP-hard problems. Our results provide an indication of the enormous power of evolutionary algorithms from a theoretical point of view. It turns out that the very general and abstract structure of matroid optimization problems suffices to lead evolutionary algorithms into promising directions and to finally obtain optimal or at least provably good solutions after only polynomially many iterations.

Acknowledgements. The authors thank Alexander Souza for helpful discussions on the topic of this article. We thank Heiko Röglin for pointing out the idea behind Theorem 4.

References

1. T. Bäck, D. B. Fogel, and Z. Michalewicz. *Handbook of Evolutionary Computation*. Oxford University Press, 1997.
2. A. Balakrishnan, T. L. Magnanti, and P. Mirchandani. Designing hierarchical survivable networks. *Operations Research*, 46(1):116–136, 1998.
3. W. H. Cunningham. Improved bounds for matroid partition and intersection algorithms. *SIAM Journal on Computing*, 15(4):948–957, 1986.
4. B. Dörr, E. Happ, and C. Klein. Crossover can provably be useful in evolutionary computation. In *Proc. of the 10th Genetic and Evolutionary Computation Conference (GECCO '08), Atlanta, USA*, 2008.
5. S. Droste, T. Jansen, and I. Wegener. On the analysis of the (1+1) evolutionary algorithm. *Theoretical Computer Science*, 276:51–81, 2002.
6. J. Edmonds. Submodular functions, matroids, and certain polyhedra. In R. Guy, H. Hanani, N. Sauer, and J. Schönheim, editors, *Combinatorial Structures and Their Applications*, pages 69–87. Gordon and Breach, New York, 1970.
7. J. Edmonds. Matroids and the greedy algorithm. *Mathematical Programming*, 1:127–136, 1971.
8. A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing*. Springer, 2nd edition, 2007.
9. S. Fischer and I. Wegener. The one-dimensional Ising model: Mutation versus recombination. *Theoretical Comput. Sci.*, 344(2–3):208–225, 2005.
10. H. N. Gabow. A matroid approach to finding edge connectivity and packing arborescences. In *Proc. of the 23rd Annual ACM Symp. on Theory of Computing (STOC '91), New Orleans, USA*, pages 112–122, 1991.
11. H. N. Gabow and Y. Xu. Efficient theoretic and practical algorithms for linear matroid intersection problems. *Journal of Computer and System Sciences*, 53:129–147, 1996.
12. D. Gale. Optimal assignments in an ordered set: an application of matroid theory. *Journal of Combinatorial Theory*, 4:176–180, 1968.
13. M. R. Garey and D. S. Johnson. *Computers and Intractability. A Guide to the Theory of NP-Completeness*. Freeman, 1979.
14. O. Giel and I. Wegener. Evolutionary algorithms and the maximum matching problem. In *Proc. of the 20th Symp. on Theoretical Aspects of Computer Science (STACS '03)*, pages 415–426, 2003.
15. O. Giel and I. Wegener. Maximum cardinality matchings on trees by randomized local search. In *Proc. of the 8th Genetic and Evolutionary Computation Conference (GECCO '06), Seattle, USA*, pages 539–546, 2006.
16. M. X. Goemans. Minimum bounded degree spanning trees. In *Proc. of the 47th Annual IEEE Symp. on Foundations of Computer Science (FOCS '06)*, pages 273–282, 2006.
17. N. J. A. Harvey, D. R. Karger, and K. Murota. Deterministic network coding by matrix completion. In *Proc. of the 16th Annual ACM-SIAM Symp. on Discrete Algorithms (SODA '05)*, pages 489–498, 2005.
18. R. Hassin and A. Levin. An efficient polynomial time approximation scheme for the constrained minimum spanning tree problem using matroid intersection. *SIAM Journal on Computing*, 33(2):261–268, 2004.
19. M. Held and R. M. Karp. The traveling-salesman problem and minimum spanning trees. *Operations Research*, 18:1138–1162, 1970.

20. T. A. Jenkyns. The efficiency of the greedy algorithm. In *Proc. of the 7th S-E Conference on Combinatorics, Graph Theory, and Computing, Utilitas Mathematica*, pages 341–350, 1976.
21. B. Korte and D. Hausmann. An analysis of the greedy heuristic for independence systems. In B. Alspach, P. Hell, and D. J. Miller, editors, *Aspects of Combinatorics*, volume 2 of *Annals of Discrete Mathematics*, pages 65–74. North-Holland, 1978.
22. B. Korte and J. Vygen. *Combinatorial Optimization: Theory and Applications*. Springer, 3rd edition, 2006.
23. D. Michail. *Minimum cycle basis: algorithms and applications*. PhD thesis, Saarland University, 2006.
24. F. Neumann and I. Wegener. Minimum spanning trees made easier via multi-objective optimization. *Natural Computing*, 5(3):305–319, 2006.
25. F. Neumann and I. Wegener. Randomized local search, evolutionary algorithms and the minimum spanning tree problem. *Theoretical Computer Science*, 378(1):32–40, 2007.
26. J. G. Oxley. *Matroid Theory*. Oxford University Press, 1992.
27. R. Rado. Note on independence functions. *Proc. of the London Mathematical Society*, 7(3):300–320, 1957.
28. G. R. Raidl and B. A. Julstrom. Edge sets: An effective evolutionary coding of spanning trees. *IEEE Transactions on Evolutionary Computation*, 7(3):225–239, 2003.
29. G. R. Raidl, G. Koller, and B. A. Julstrom. Biased mutation operators for subgraph-selection problems. *IEEE Transactions on Evolutionary Computation*, 10(2):145–156, 2006.
30. J. Scharnow, K. Tinnefeld, and I. Wegener. The analysis of evolutionary algorithms on sorting and shortest path problems. *Journal of Mathematical Modelling and Algorithms*, 3:349–366, 2004.
31. A. Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*. Springer, Berlin, 2003.
32. D. Sudholt. Crossover is provably essential for the Ising model on trees. In *Proc. of the 7th Genetic and Evolutionary Computation Conference (GECCO '05), Washington DC, USA*, pages 1161–1167, 2005.
33. I. Wegener. Towards a theory of randomized search heuristics. In *Proc. of Mathematical Foundations of Computer Science (MFCS'03)*, pages 125–141, 2003.
34. I. Wegener. Randomized search heuristics as an alternative to exact optimization. In W. Lenski, editor, *Logic versus Approximation*, pages 138–149. 2004.