

Fixed-Parameter Evolutionary Algorithms and the Vertex Cover Problem

Stefan Kratsch
Algorithms and Complexity Group
Max-Planck-Institut für Informatik
Saarbrücken, Germany

Frank Neumann
Algorithms and Complexity Group
Max-Planck-Institut für Informatik
Saarbrücken, Germany

ABSTRACT

In this paper, we consider multi-objective evolutionary algorithms for the VERTEX COVER problem in the context of parameterized complexity. We relate the runtime of our algorithms to the input size and the cost of a minimum solution and point out that the search process of evolutionary algorithms creates partial solutions that are similar to the effect of a kernelization (i.e. a special type of preprocessing from parameterized complexity). Based on this, we show that evolutionary algorithms solve the vertex cover problem efficiently if the size of a minimum vertex cover is not too large, i.e. the expected runtime is bounded by $O(f(\text{OPT}) \cdot n^c)$, where c is a constant and f a function that only depends on OPT . This shows that evolutionary algorithms are randomized fixed-parameter tractable algorithms for the vertex cover problem.

Categories and Subject Descriptors: F.2 [Theory of Computation]: Analysis of Algorithms and Problem Complexity

General Terms: Theory, Algorithms, Performance

Keywords: Combinatorial Optimization, Evolutionary Algorithms, Multi-Objective Optimization, Runtime Analysis

1. INTRODUCTION

General purpose algorithms such as evolutionary algorithms and ant colony optimization have been shown to be successful problem solvers for a wide range of combinatorial optimization problems. Such algorithms make use of random decisions which allows to consider them as a special class of randomized algorithms. Especially if the problem is new and there are not enough resources such as time, money, or knowledge about the problem to develop specific algorithms, general purpose algorithms often produce good results without a large development effort. Usually, it is just necessary

to think about a representation of possible solutions, a function to measure the quality of solutions, and operators that produce from a solution (or a set of solutions) a new solution (or a set of solutions).

Taking such a general approach to solve a given problem, it is clear that we cannot hope to beat algorithms that are tailored to the problem. However, such general approaches find many applications when no good problem specific algorithm is available. In addition to many experimental studies that confirm the success of these algorithms on problems from different domains, there has been increasing interest in understanding such algorithms also in a rigorous way. This line of research treats such algorithms as a class of randomized algorithms and analyzes them in a classical fashion, i.e. with respect to their runtime behavior and approximation ability in expected polynomial time. The results obtained in this research area confirm that general purpose algorithms often come up with optimal solutions quickly even if they do not use problem specific knowledge. Problems that have been studied among many others within this line of research are the shortest path problem [5, 16], maximum matchings [10], minimum spanning trees [12, 13], covering and scheduling problems [18]. Additionally, recent theoretical studies have investigated the learning ability of evolutionary algorithms [7, 17].

For NP-hard problems we cannot hope to prove practicality in the sense of a polynomial upper-bound on the worst-case runtime even though an algorithm might perform very well in practice. Nevertheless, the notion of fixed-parameter tractability may be helpful to explore that situation as well as guiding further algorithm design. Fixed-parameter tractability is a central concept of parameterized complexity. In that field, the complexity of input instances is measured in a two-dimensional way considering not only the input size but also one or more parameters, e.g. solution size, structural restrictions, or quality of approximation. One hopes to confine the inevitable combinatorial explosion in the runtime to a function in the parameter, with only polynomial dependence on the input size. The idea is that even large instances may exhibit a very restricted structure and can therefore be considered easy to solve, despite their size.

A parameterized problem with parameter k is *fixed-parameter tractable* (FPT) if there exists an algorithm that decides it in time $O(f(k) \cdot n^c)$ (or equivalently $O(g(k) + n^d)$), implying polynomial running time of degree independent of k for every fixed value of k . Such an algorithm is called an *FPT algorithm* for the problem, respectively *randomized*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO'09, July 8–12, 2009, Montreal, Canada.

Copyright 2007 ACM 978-1-59593-697-4/07/0007 ...\$5.00.

FPT algorithm in the case of expected runtime. Formally FPT is the class of all parameterized languages, i.e. decision problems, that can be decided by an FPT algorithm. However, for natural decision versions of optimization problems, i.e. asking whether the cost of optimal solutions is at most (respectively at least) k , FPT algorithms can be modified to compute the optimum cost and mostly also to compute an optimal solution.

In this paper we want to adopt a parameterized view on evolutionary algorithms and consider their expected runtime behavior related to the solution size OPT. We examine when evolutionary algorithms compute a solution quickly if OPT is small, i.e. in expected time $O(f(\text{OPT}) \cdot n^c)$. In relation to randomized FPT algorithms we call an evolutionary algorithm with such a runtime bound *evolutionary FPT algorithm* as it also solves the decision variant of the parameterized problem in expected time $O(f(k) \cdot n^c)$.

An important stepping stone in the analysis of our algorithms will be the fact that they create partial solutions that can be considered problem kernels of the original instance, given a feasible secondary measure. A *kernelization* or *reduction to a problem kernel* is a special form of polynomial-time data reduction for parameterized problems that produces an equivalent (and usually smaller) instance whose size is bounded by a function in the original parameter. It is known that a parameterized problem is fixed-parameter tractable if and only if there exists a kernelization for the problem.

A well known fixed-parameter tractable problem is the (standard) parameterized VERTEX COVER problem. Given an undirected graph and an integer k it has to be decided whether there exists a set of at most k vertices such that each edge contains at least one of these vertices, parameterized by k . This problem can be solved in time $O(1.2738^k + kn)$ via kernelization followed by a bounded search tree algorithm [3].

The VERTEX COVER problem has also been studied in the field of evolutionary computation from a theoretical point of view. Rigorous runtime analysis has been given for the well-known (1+1) EA and population based algorithms for single-objective optimization [8, 14, 15]. Additionally, it has been shown that a multi-objective model can help the optimization process of an evolutionary algorithm to find good solutions quicker than in a single-objective one [9]. Due to the results obtained in [9] we consider two different multi-objective models for the VERTEX COVER problem. Both models take as the first objective the goal to minimize the number of chosen vertices. The second criteria should be a penalty function which has to be minimized such that a vertex cover is obtained.

Minimizing the number of uncovered edges as the second objective has already been investigated in [9] and we study this approach with respect to the approximation quality depending on the value of OPT. Afterwards, we examine this approach with respect to the expected runtime in dependence of OPT and show that this approach leads to evolutionary FPT algorithms.

Our second approach is to take the minimum cost of a fractional vertex cover for the uncovered edges as one objective. We show that this approach leads to a 2-approximation for VERTEX COVER in expected polynomial time and to evolutionary FPT algorithms whose expected runtime can be bounded by $O(n^3 + n \cdot 4^{\text{OPT}})$. For the case where one is

interested in computing a $(1 + \epsilon)$ -approximation, we reduce the runtime bound of this approach to $O(n^3 + n \cdot 4^{(1-\epsilon)\text{OPT}})$.

The outline of the paper is as follows. In Section 2, we introduce the VERTEX COVER problem as well as the algorithms and problem formulations that are subject to our investigations. In Sections 3 and 4 we consider two different multi-objective models for VERTEX COVER. In Section 5 we summarize our results and give possible directions for further research.

2. PRELIMINARIES

The VERTEX COVER problem is one of the well-known NP-hard combinatorial optimization problems. Given an undirected graph $G = (V, E)$ where $|V| = n$ and $|E| = m$ the aim is to find a subset $V' \subseteq V$ of minimum cardinality such that for each $e \in E$, $e \cap V' \neq \emptyset$ holds. Many simple approximation algorithms achieve a worst-case approximation ratio of 2 (cf. [4]). For example such an approximation can be achieved in polynomial time by computing a maximal matching in the given graph and choosing for each edge of the matching the corresponding two vertices.

The VERTEX COVER problem can be formulated as an integer linear program (ILP) in the following way:

$$\begin{aligned} \min & \sum_{i=1}^n x_i \\ \text{s.t.} & x_i + x_j \geq 1 \quad \forall \{i, j\} \in E \\ & x_i \in \{0, 1\} \end{aligned}$$

Relaxing the integrality constraint $x_i \in \{0, 1\}$ to fractional values between 0 and 1, i.e. $x_i \in [0, 1]$, yields a linear program formulation of the FRACTIONAL VERTEX COVER problem. Clearly, for any graph, the cost of an optimal fractional vertex cover is a lower bound on the cardinality of a minimum (integral) vertex cover. The dual problem of FRACTIONAL VERTEX COVER is FRACTIONAL MAXIMUM MATCHING, i.e. MAXIMUM MATCHING with relaxed integrality.

It has been pointed out that simple evolutionary algorithms cannot achieve a non-trivial approximation guarantee, i.e. there are instances where the well-known (1+1) EA cannot obtain a better approximation than a factor $\Theta(n)$ in expected polynomial time [9]. In contrast to this a multi-objective model in conjunction with a simple evolutionary algorithm leads to a $O(\log n)$ -approximation on the much broader class of setcover problems. We follow this approach and examine the multi-objective model for VERTEX COVER in conjunction with the simple multi-objective evolutionary algorithm called Global SEMO (Global Simple Evolutionary Multi-Objective Optimizer). This algorithm has already been studied for a wide range of multi-objective optimization problems and can be considered as the generalization of the (1+1) EA to the multi-objective case.

Global SEMO (see Algorithm 1) keeps at each time step for each non dominated objective vector found so far one single solution. In this way it preserves an approximation of the Pareto front. The algorithm starts with an initial solution that is chosen uniformly at random from the underlying search space. In each iteration, a solution x from the current population P is chosen uniformly at random. A mutation operator flipping each bit of x with probability $1/n$ is applied to obtain an offspring x' . This solution x' is introduced into the population iff it is not dominated by any other solution

in the population. If this is the case, all solutions that are weakly dominated by x' are deleted from P .

ALGORITHM 1. *Global SEMO*

1. Choose $x \in \{0, 1\}^n$ uniformly at random.
2. Determine $f(x)$.
3. $P \leftarrow \{x\}$.
4. Repeat
 - Choose $x \in P$ uniformly at random.
 - Create x' by flipping each bit x_i of x with probability $1/n$.
 - Determine $f(x')$.
 - If x' is not dominated by any other search point in P , include x' into P and delete all other solutions $z \in P$ with $f(x') \leq f(z)$ from P .

Denote by $E(x) \subseteq E$ the set of edges for which at least one vertex is chosen by x . As each edge $e \in E$ has to be covered by at least one vertex to obtain a vertex cover, it may be helpful to flip vertices which are incident with uncovered edges with a larger probability. This leads to the following alternative mutation operator.

ALGORITHM 2. *Alternative mutation operator*

- Choose $b \in \{0, 1\}$ uniformly a random.
- If $b = 1$ and there exists an edge $\{v_i, v_j\} \in E \setminus E(x)$, flip x_i with probability $1/2$. Otherwise flip x_i with probability $1/n$.

In the alternative mutation operator vertices that are incident with an uncovered edge may be flipped with a larger probability of $1/2$. These are exactly the non-isolated vertices of $G(x) = (V, E \setminus E(x))$. Replacing the mutation operator of Global SEMO by Algorithm 2 we call this algorithm Global SEMO_{alt}.

The fitness function

$$f_1(x) = (|x|_1, u(x))$$

where $|x|_1$ denotes the number of chosen vertices and $u(x)$ denotes the number of edges that are not covered by any vertex chosen by x has already been considered in [9].

Additionally, we also examine the fitness function

$$f_2(x) = (|x|_1, LP(x))$$

where $LP(x)$ denotes the optimum value of the relaxed VERTEX COVER ILP for $G(x)$, i.e. the cost of an optimal fractional vertex cover of $G(x)$.

Our goal is to analyze our algorithms until they have found an optimal solution or a good approximation of an optimal one. Our algorithms using the function f_1 (or f_2) have produced an r -approximation for the VERTEX COVER problem iff they have produced a solution x with objective vector $f_1 = (|x|_1, 0)$ (or $f_2 = (|x|_1, 0)$) where $\frac{|x|_1}{\text{OPT}} \leq r$.

To measure the runtime of our algorithms, we consider the number of fitness evaluations until a minimum vertex cover or a good approximation of such a solution has been

obtained. The expected optimization time refers to the expected number of fitness evaluations until a minimum vertex cover has been obtained. Similar we often consider the expected time to achieve intermediate goals, e.g. partial solutions of a vertex cover that fulfill certain properties.

For both introduced fitness functions, the search point 0^n is Pareto optimal as the first objective for all functions is to minimize the number of ones in the bitstring. A key idea in the remaining part of the paper is to proceed from this solution towards a minimum vertex cover or a vertex cover of a certain approximation quality.

LEMMA 1. *The expected number of iterations of Global SEMO or Global SEMO_{alt} until the population contains the search point 0^n is $O(n^2 \log n)$ for the fitness functions f_1 and f_2 .*

Proof. The size of the population is upper bounded by $n+1$ as the population keeps for each i , $0 \leq i \leq n$, at most one solution x for each fixed number of ones in the bitstring. We consider in each step the individual $y = \text{argmin}_{z \in P} |z|_1$. The probability to choose this individual in the next step is at least $\frac{1}{n+1}$. Let $i = |y|_1$ be the number of ones in this bitstring. The probability of producing a solution with a smaller number of ones is lower bounded by

$$\frac{1}{n+1} \cdot \frac{1}{2} \cdot \frac{i}{en} = \Omega\left(\frac{i}{n^2}\right)$$

and the expected waiting time until a solution with a most $i-1$ ones has been produced is therefore $O(n^2/i)$. Using the method of fitness based partitions and summing up over the different values of i , the time until the search point 0^n has been included into the population is $O(n^2 \log n)$. \square

After an expected number of $O(n^2 \log n)$ iterations both algorithms working on the fitness function f_1 or f_2 introduce the search point 0^n into the population. Afterwards, this search point stays in the population. The population size of both algorithms is upper bounded by $n+1$. This may be used to give a bound on the expected time to reach a minimum vertex cover depending on OPT

Let x be an arbitrary solution that remains in the population during the optimization process. The probability of producing a specific solution x' that has Hamming-distance c to x in the next step is lower bounded by

$$\frac{1}{2(n+1)} \cdot \left(\frac{1}{n}\right)^c \cdot (1-1/n)^{n-c} = \Omega(n^{-(c+1)})$$

which implies that the expected time to produce such a solution is $O(n^{c+1})$

Hence, both algorithms obtain an optimal solution in expected time $O(n^{\text{OPT}+1})$ after they have obtained the search point 0^n . Note, that this time bound is not sufficient for our definition of evolutionary FPT algorithms.

3. MINIMIZING THE NUMBER OF UNCOVERED EDGES

In this section, we consider the effect of minimizing the number of uncovered edges as the second criteria by investigating the fitness function f_1 . Note, that this approach has already

been investigated in [9]. In that paper, it has been showed that there are bipartite graphs where the (1+1) EA cannot achieve a good approximation in expected polynomial time. Running Global SEMO on these instances solves the problem quickly. For general graphs, it has been showed that Global SEMO achieves a $\log n$ -approximation in expected polynomial time.

In the following, we show a bound on the approximation quality depending on the value of OPT that Global SEMO or Global SEMO_{alt} can achieve in polynomial time. Furthermore we prove that under this secondary measure the expected number of iterations until Global SEMO_{alt} finds a minimum vertex cover is bounded by

$$O(\text{OPT} \cdot n^4 + n \cdot 2^{\text{OPT}^2 + \text{OPT}}).$$

A central idea in our proofs is to consider a solution $x \in P$ where the set of vertices is a subset of a minimum vertex cover and such that $G(x)$ does not contain vertices of degree greater than OPT. The following lemma shows that Global SEMO and Global SEMO_{alt} spend an expected number of $O(\text{OPT} \cdot n^4)$ steps on producing such solutions during the run of the algorithm.

LEMMA 2. *The expected number of iterations of Global SEMO and Global SEMO_{alt} where the population does not contain a solution x that fullfills the properties that*

1. *the vertices chosen by x constitute a subset of a minimum vertex cover of G and that*
2. *$G(x)$ contains no vertex of degree greater than OPT,*

is upper bounded by $O(\text{OPT} \cdot n^4)$.

Proof. We know that the search point 0^n is introduced into the population after an expected number of $O(n^2 \log n)$ iterations. Assuming that the search point 0^n has already been introduced into the population, we show that an expected number of $O(\text{OPT} \cdot n^4)$ iterations occur where the population does not contain a solution with the desired properties.

We denote by $V' \subseteq V$ the set of vertices that have degree larger than OPT in G . Clearly every vertex cover of cardinality OPT must contain V' , since otherwise all the neighbors of a vertex $v \in V'$ would have to be chosen. We assume that $V' \neq \emptyset$ as otherwise 0^n has the desired properties.

The idea to prove the lemma is to investigate a potential taking on $O(|E| \cdot \text{OPT})$ different values. If the population does not contain a solution with properties 1 and 2 this potential is decreased with probability $\Omega(1/n^2)$ which leads to the stated upper bound on the number of steps that have a population where each solution does not fullfill the desired properties.

Let $s_0, s_1, \dots, s_{\text{OPT}}$ be integer values such that s_j is the smallest value of $u(x)$ for any search point x in P choosing at most j vertices, i.e. $|x|_1 \leq j$. Note, that each s_j cannot increase during the run of the algorithm as only non-dominated solutions are accepted.

We investigate the potential of a population P given by

$$\text{pot}(P) = \sum_{j=1}^{\text{OPT}} s_j \leq |E| \cdot \text{OPT}.$$

Let i be the largest integer such that P contains solutions x_0, \dots, x_i with fitness $(0, s_0), \dots, (i, s_i)$ that select only vertices of V' . We will now consider different cases to show that either x_i has the desired properties or that, with probability $\Omega(1/n^2)$, a solution is generated that improves at least one of the s_j .

1. If the graph $G(x_i)$ contains no vertex of degree larger than OPT then x_i fulfills properties 1 and 2 by selection of i . For the other cases we assume that $G(x_i)$ contains a vertex of degree greater than OPT, say v .
2. If $s_i - s_{i+1} \leq \text{OPT}$ (note: this includes the case that P does not contain any solution x with $|x|_1 = i + 1$, implying that $s_{i+1} = s_i$) then with probability $\Omega(1/n^2)$ Global SEMO or Global SEMO_{alt} chooses the search point x_i and mutates it into a point x'_{i+1} that additionally selects v . Clearly

$$u(x'_{i+1}) = u(x_i) - \text{deg}_{G(x_i)}(v) < s_i - \text{OPT}.$$

Thus $u(x'_{i+1}) < s_{i+1}$, implying that s_{i+1} is decreased by at least one.

3. If $s_i - s_{i+1} > \text{OPT}$ then P contains a solution x_{i+1} of fitness $(i + 1, s_{i+1})$ and x_{i+1} selects at least one vertex $u \in V \setminus V'$ by choice of i . With probability at least $\Omega(1/n^2)$ the search point x_{i+1} is chosen and is mutated into a solution x'_i by flipping only the bit corresponding to u . Thus

$$u(x'_i) = u(x_{i+1}) + \text{deg}_{G(x'_i)}(u) \leq s_{i+1} + \text{OPT}.$$

Therefore $u(x'_i) < s_i$, i.e. s_i is improved by at least one.

In each case we get that either P contains a solution as claimed in the lemma or with probability $\Omega(1/n^2)$ the potential decreases by at least one. The potential can take on only $O(\text{OPT} \cdot |E|)$ different values which completes the proof. \square

We have seen that in all but expected $O(\text{OPT} \cdot n^4)$ iterations of Global SEMO or Global SEMO_{alt} the population contains a solution x that is a subset of some minimum vertex cover and such that $G(x)$ has maximum degree OPT. Such partial solutions will be useful in the proving an upper bound on the expected number of iterations of Global SEMO_{alt} to generate a minimum vertex cover, while also implying that an OPT-approximate vertex cover is produced in expected polynomial number of iterations of Global SEMO or Global SEMO_{alt}. One can easily see that $G(x)$ has at most $(\text{OPT} - |x|_1) \cdot \text{OPT}$ uncovered edges, since $(\text{OPT} - |x|_1)$ vertices of degree at most OPT suffice to cover all of them.

Though these partial solutions are obtained in a randomized fashion aiming to cover as many edges as possible with few vertices, they are strongly related to deterministic preprocessing for the parameterized VERTEX COVER problem. To decide whether a given graph has a vertex cover of size at most k one may select all vertices of degree larger than k . In fact, if v is a vertex of degree larger than k then G has a vertex cover of cardinality k if and only if $G - v$ has a vertex cover of cardinality $k - 1$. In conjunction with deleting isolated vertices this leads to an equivalent reduced instance

with at most $O(k^2)$ vertices, this technique being known as Buss' kernelization (cf. [6]).

These structural insights can be used to show that our algorithms achieve an OPT-approximation in expected polynomial time when using the fitness function f_1

THEOREM 1. *The expected number of iterations of Global SEMO or Global SEMO_{alt} until an OPT-approximation is computed is $O(\text{OPT} \cdot n^4)$.*

Proof. According to Lemma 1 and Lemma 2, we already know that the expected number of steps where the population does not contain a solution with the properties stated in Lemma 2 is $O(\text{OPT} \cdot n^4)$. In the following, we consider only steps where such a solution exists.

Thus it is ensured that there is a solution x in the population for which $|x|_1 \leq \text{OPT}$ and the maximum degree of $G(x)$ is at most OPT. This implies $u(x) \leq (\text{OPT} - |x|_1) \cdot \text{OPT}$ and $|x|_1 + u(x) \leq \text{OPT}^2$. If x is dominated by any solution x' then clearly $|x'|_1 + u(x') \leq \text{OPT}^2$. Therefore, in all later steps the population contains at least one solution y with $|y|_1 + u(y) \leq \text{OPT}^2$.

Let u denote the minimum value of $u(x)$ among solutions $x \in P$ with $|x|_1 + u(x) \leq \text{OPT}^2$. Let $y \in P$ be a solution with $|y|_1 + u(y) \leq \text{OPT}^2$ and $u(y) = u$. If $u(y) = 0$ it follows that y selects at most OPT^2 vertices which are a vertex cover. Otherwise at least one vertex v of $G(y)$ is incident with an (uncovered) edge.

The probability that y is selected and that it is mutated into a solution y' that additionally selects v is $\Omega(1/n^2)$ for Global SEMO and Global SEMO_{alt}. Clearly the solution y' fullfills $|y'|_1 + u(y') \leq |y|_1 + u(y)$ and $u(y') < u(y)$. Observe that y' cannot be dominated by any solution in P due to $|y'|_1 + u(y') \leq |y|_1 + u(y)$ and by choice of y , implying that it is added to P , decreasing u by at least 1.

If the solution y with $u(y) = u$ and $|y|_1 + u(y) \leq \text{OPT}^2$ is removed from the population then there must be a solution, say z , that dominates it. By $u(z) \leq u(y)$ and $|z|_1 \leq |y|_1$ this cannot increase the value of u . Clearly $0 \leq u \leq \text{OPT}^2$, i.e. it can be decreased at most OPT^2 times.

Thus after expected $O(\text{OPT}^2 \cdot n^2 + \text{OPT} \cdot n^4)$ iterations of Global SEMO or Global SEMO_{alt} a solution with fitness $(S, 0)$ with $S \leq \text{OPT}^2$ is obtained. \square

After having shown that both algorithms achieve an OPT-approximation in expected polynomial time, we will bound the time until Global SEMO_{alt} achieves an optimal solution.

THEOREM 2. *The expected number of iterations of Global SEMO_{alt} until it has computed a minimum vertex cover is $O(\text{OPT} \cdot n^4 + n \cdot 2^{\text{OPT} + \text{OPT}^2})$.*

Proof. As in the proof of Theorem 1, we assume that P contains a solution x such that $G(x)$ has maximum degree at most OPT and there exists a minimum vertex cover S that contains the vertices selected by x . Due to Lemma 2 the expected number of iterations where Global SEMO_{alt} does not fulfill the properties is $O(\text{OPT} \cdot n^4)$, i.e. adding this term to the obtained bound covers the assumption.

The probability of choosing x in the next mutation step is $\Omega(1/n)$. Choosing all the remaining vertices of S and not flipping any other bit in x leads to a minimum vertex cover. The graph $G(x)$ has maximum degree OPT and it

has a vertex cover of size $(\text{OPT} - |x|_1)$. Each vertex in such a vertex cover can be adjacent to at most OPT non-isolated vertices (and all edges are incident with the cover), implying that $G(x)$ has at most $(\text{OPT} - |x|_1) + (\text{OPT} - |x|_1) \cdot \text{OPT} \leq \text{OPT} + \text{OPT}^2$ non-isolated vertices.

We consider the mutation of x which flips vertices adjacent to non-covered edges with probability 1/2. Note that with probability $(1 - 1/n)^{n'} \in \Omega(1)$ no bit corresponding to any of the $n' \leq n$ isolated vertices of $G(x)$ is flipped. The probability of flipping only the bits corresponding to the missing vertices of S is therefore $\Omega(2^{-(\text{OPT} + \text{OPT}^2)})$, since there are at most $\text{OPT} + \text{OPT}^2$ non-isolated vertices. Hence, the expected time until a minimum vertex cover has been computed is upper bounded by $O(\text{OPT} \cdot n^4 + n \cdot 2^{\text{OPT} + \text{OPT}^2})$. \square

4. FRACTIONAL VERTEX COVERS

In this section, we use the minimum cost of a fractional vertex cover for the uncovered edges as the second criteria. For every search point x this gives an estimate on how many vertices are needed to cover $G(x)$. We denote this cost by $LP(x)$, as it is the optimal cost of solutions to the VERTEX COVER ILP with relaxed integrality constraints, i.e. $0 \leq x_i \leq 1$ in place of $x_i \in \{0, 1\}$. Balinski [1] showed that all basic feasible solutions (or extremal points) of the FRACTIONAL VERTEX COVER LP are half-integral.

LEMMA 3. *Every basic feasible solution x of the relaxed VERTEX COVER ILP is half-integral, i.e. $x \in \{0, 1/2, 1\}^n$.*

Due to this lemma, optimal fractional vertex covers can be computed very efficiently via a maximum matching of an auxiliary bipartite graph (cf. [2]). Throughout the section we will implicitly assume that chosen fractional vertex covers are half-integral.

Nemhauser and Trotter [11] proved a very strong relation between optimal fractional vertex covers and minimum vertex covers.

THEOREM 3. *Let x^* be an optimal fractional vertex cover and let $P_0, P_1 \subseteq V$ be the vertices whose corresponding components of x^* are 0 or 1 respectively, then there exists a minimum vertex cover that contains P_1 and no vertex of P_0 .*

We start with a simple lemma that gives insights into the structure of the objective space.

LEMMA 4. *For every $x \in \{0, 1\}^n$ it holds that*

1. $|x|_1 + LP(x) \geq LP(0^n)$.
2. $|x|_1 + 2 \cdot LP(x) \geq \text{OPT}$.

Proof. Let y be an optimal fractional vertex cover of $G(x)$ of cost $LP(x)$.

1.) One can obtain a fractional vertex cover of G from y by adding the vertices that are selected by x . The cost of this cover, i.e. $|x|_1 + LP(x)$, cannot be smaller than the minimum cost of a fractional vertex cover, i.e. $LP(0^n)$.

2.) Similarly a vertex cover of G can be obtained by adding all vertices that have value 1/2 or 1 in y to the vertices selected by x , since each edge of $G(x)$ must be incident with vertices of total value of at least one. The cardinality of this vertex cover is bounded by $2 \cdot LP(x)$ (i.e. the maximum number of vertices with value 1/2 or 1) plus $|x|_1$. Clearly this vertex cover cannot be smaller than a minimum vertex cover (with cardinality OPT). \square

Hence, each solution for which equality holds in one of the inequalities stated in Lemma 4 is Pareto optimal. The following lemma relates a search point $x \in \{0, 1\}^n$ to an optimal fractional solution $x^* \in [0, 1]^n$. For $x, y \in [0, 1]^n$, we denote by $x \leq y$ the fact that $x_i \leq y_i$, $1 \leq i \leq n$.

LEMMA 5. *Let y be an optimal fractional vertex cover of G . Every $x \in \{0, 1\}^n$ with $x \leq y$, is a Pareto optimal solution.*

Proof. Let y' be obtained from y by setting the value of all vertices that are selected by x to 0. The graph $G(x)$ contains all edges that are not incident to any vertex that is selected by x . Thus y' is a fractional vertex cover of $G(x)$. Therefore we have $|y|_1 - |x|_1 = |y'|_1 \geq LP(x)$, implying that $LP(0^n) = |y|_1 \geq LP(x) + |x|_1$. Thus, by Lemma 4, we can conclude that $|x|_1 + LP(x) = LP(0^n)$ and that x is a Pareto optimal solution. \square

We state a simple property that describes search points that are subsets of a minimum vertex cover. Such solutions are of particular interest as they can be turned into a minimum vertex cover by adding vertices.

LEMMA 6. *Let x be a solution with $LP(x) = LP(0^n) - |x|_1$, then there exists a minimum vertex cover $y \in \{0, 1\}^n$ with $x \leq y$ (i.e. every vertex selected by x is also selected by y).*

Proof. Consider an optimal fractional vertex cover y of $G(x)$ of cost $LP(0^n) - |x|_1$. We can obtain a fractional vertex cover y' of G by also selecting the $|x|_1$ vertices that are selected by x (i.e. setting the corresponding components of y to 1). Hence y' is a fractional vertex cover of G of cost $LP(0^n)$, implying that it is optimal. By Theorem 3 it follows that there exists a minimum vertex cover of G that contains all vertices with value 1 in y' which includes all vertices that are selected by x . \square

After having pointed out some basic properties about fractional vertex covers and Pareto optimal solutions, we can now analyze our algorithms with respect to the approximation that they can achieve in expected polynomial time. It is easy to see that, for every optimal fractional vertex cover, the vertices of value 1/2 and 1 form a 2-approximate vertex cover, since the fractional vertex cover has cost at most OPT.

THEOREM 4. *The expected number of iterations of Global SEMO or Global SEMO_{alt} until the population P contains a 2-approximate vertex cover is $O(n^2 \log n + \text{OPT} \cdot n^2)$.*

Proof. The expected number of iterations until the search point 0^n is added to the population is $O(n^2 \log n)$ due to Lemma 1.

Let $x \in P$ be a solution that minimizes $LP(x)$ under the constraint that $|x|_1 + 2 \cdot LP(x) \leq 2 \cdot LP(0^n) \leq 2 \cdot \text{OPT}$. Note, that 0^n fulfills the constraint. If $LP(x) = 0$ then x is a vertex cover of G and $|x|_1 \leq 2 \cdot LP(0^n) \leq 2 \cdot \text{OPT}$ as claimed. Otherwise every optimal fractional vertex cover of $G(x)$ assigns at least 1/2 to some vertex, say v . Therefore, $LP(x') \leq LP(x) - 1/2$ where x' is obtained from x by additionally selecting v . With probability at least $\Omega(1/n^2)$ the solution x is picked in the mutation step and exactly

the bit corresponding to v is flipped, leading to the solution x' . Clearly, $|x'|_1 = |x|_1 + 1$ and $LP(x') \leq LP(x) - 1/2$. Thus $|x'|_1 + 2 \cdot LP(x') \leq |x|_1 + 2 \cdot LP(x) \leq 2 \cdot LP(0^n)$, implying that x' fulfills the constraint while having a smaller value $LP(x')$. Thus, x' is added to the population since no solution in P dominates it, by selection of x .

As $LP(x) \leq \text{OPT}$ this can happen at most $2 \cdot \text{OPT}$ times since each time the smallest value of $LP(x)$ among solutions x that fulfill $|x|_1 + 2 \cdot LP(x) \leq 2 \cdot \text{OPT}$ is reduced by at least 1/2. Therefore, the expected number of steps until the population contains a 2-approximate vertex cover is at most $O(n^2 \log n + \text{OPT} \cdot n^2)$. \square

After having shown that using the minimum cost of a fractional vertex cover as the second criteria leads to a 2-approximation, we will now examine the number of iterations until Global SEMO_{alt} has obtained an optimal solution.

To prove an upper bound on that number we consider solutions choosing r vertices such that the subgraph consisting of the non-covered edges has at most $2 \cdot (LP(0^n) - r)$ non-isolated vertices. Therefore we are interested in solutions x of fitness $(|x|_1, LP(0^n) - |x|_1)$ such that optimal fractional vertex covers of $G(x)$ assign 1/2 to each non-isolated vertex, implying that there are exactly $2 \cdot (LP(0^n) - |x|_1)$ non-isolated vertices in $G(x)$.

LEMMA 7. *The expected number of iterations during the run of Global SEMO and Global SEMO_{alt} where the population does not contain a solution x that fulfills the properties*

1. *that $LP(x) = LP(0^n) - |x|_1$ and*
2. *that each optimal fractional vertex cover assigns 1/2 to each non-isolated vertex of $G(x)$*

is upper bounded by $O(n^2 \cdot \log n + \text{OPT} \cdot n^2)$.

Proof. After expected $O(n^2 \log n)$ iterations the population contains the solution 0^n of fitness $(0, LP(0^n))$.

Let r be the largest integer such that P contains solutions of fitness values $(0, LP(0^n)), \dots, (r, LP(0^n) - r)$ and let $x \in P$ be the solution of fitness $(r, LP(0^n))$. There are two possible cases:

1. Every optimal fractional vertex cover of assigns 1/2 to each non-isolated vertex of $G(x)$.
2. There is an optimal fractional vertex cover z of $G(x)$ which assigns 1 to at least one non-isolated vertex of $G(x)$, say v . With probability at least $\Omega(1/n^2)$ Global SEMO or Global SEMO_{alt} chooses the solution x for mutation and flips exactly the bit corresponding to v , obtaining a solution x' .

Observe that $LP(x') \leq LP(x) - 1$ since z' , i.e. the same as z but assigning 0 to v , is a fractional vertex cover of $G(x')$. Clearly, x' is added to the population since solutions of fitness $(i, LP(0^n) - i)$ are Pareto optimal, according to Lemma 4. This increases the value of r by 1.

Since $0 \leq r \leq LP(0^n) \leq \text{OPT}$ its value can be increased at most OPT times. Therefore the expected number of steps in which case 2 happens is at most $O(n^2 \cdot \log n + \text{OPT} \cdot n^2)$. \square

Both algorithms generate a search point x that selects a subset of a minimum vertex cover and such that $G(x)$ has at most $2 \cdot (\text{OPT} - |x|_1)$ non-isolated vertices in expected polynomial time and, similar to Lemma 2, the population contains such a solution in all but expected $O(n^2 \cdot \log n + \text{OPT} \cdot n^2)$ iterations.

In the following, we show that Global SEMO_{alt} is able to produce from such a solution an optimal one in expected time $O(n^2 \cdot \log n + \text{OPT} \cdot n^2 + n \cdot 4^{\text{OPT}})$ which implies that it is an evolutionary FPT algorithm for the VERTEX COVER problem.

THEOREM 5. *The expected number of iterations of Global SEMO_{alt} until it has computed a minimum vertex cover is $O(n^2 \cdot \log n + \text{OPT} \cdot n^2 + n \cdot 4^{\text{OPT}})$*

Proof. We consider iterations of Global SEMO_{alt} where the population contains a solution x with $LP(x) = LP(0^n) - |x|_1$ such that each optimal fractional vertex cover assigns 1/2 to each non-isolated vertex of $G(x)$. By Lemma 7 the expected number of iterations where this is not the case is at most $O(n^2 \cdot \log n + \text{OPT} \cdot n^2)$.

According to Lemma 6 there exists a minimum vertex cover y with $x \leq y$, i.e. y contains the vertices that are selected by x . Let V' be the set of vertices that are selected by y but not by x . Observe that every vertex of V' is non-isolated in $G(x)$, i.e. incident with an uncovered edge, since y is a minimum vertex cover. With probability at least $1/n + 1$ the solution x is picked in the mutation step. The probability that y is obtained in that case can be easily lower bounded:

- With probability 1/2 Global SEMO_{alt} chooses the mutation proves that flips every bit that corresponds to a non-isolated vertex of $G(x)$ with probability 1/2.
- In that case, the probability that exactly the bits corresponding to V' are flipped (to 1) is $\Omega(2^{2 \cdot (\text{OPT} - |x|_1)})$ since there are at most $2 \cdot (\text{OPT} - |x|_1)$ vertices that are incident with uncovered edges in $G(x)$. This includes a factor of $\Omega(1)$ for the probability that Global SEMO_{alt} does not flip bits corresponding to isolated vertices of $G(x)$, which is $(1 - 1/n)^{n'}$ for $n' \leq n$ isolated vertices.

Thus with probability at least $1/n \cdot 1/2 \cdot (1/4)^{\text{OPT}}$ the solution y of fitness $(\text{OPT}, 0)$ is obtained. Therefore, the expected number of iterations of Global SEMO_{alt} until the population contains a minimum vertex cover is bounded by $O(n^2 \cdot \log n + \text{OPT} \cdot n^2 + n \cdot 4^{\text{OPT}})$. \square

In the final theorem of this section we prove that the expected number of iterations until Global SEMO_{alt} has generated a $(1 + \epsilon)$ -approximate vertex cover is bounded by $O(n^2 \cdot \log n + \text{OPT} \cdot n^2 + n \cdot 4^{(1 + \epsilon) \cdot \text{OPT}})$. This implies that the expected approximation ratio of the vertex cover generated by Global SEMO_{alt} improves over time (that is to say, the upper bound on that ratio decreases) to the point where it reaches 1 at time $O(n^2 \cdot \log n + \text{OPT} \cdot n^2 + n \cdot 4^{\text{OPT}})$.

THEOREM 6. *The expected number of iterations of Global SEMO_{alt} until it has generated a $(1 + \epsilon)$ -approximate vertex cover, i.e. a solution of fitness $(r, 0)$ with $r \leq (1 + \epsilon) \cdot \text{OPT}$, is $O(n^2 \cdot \log n + \text{OPT} \cdot n^2 + n \cdot 4^{(1 + \epsilon) \cdot \text{OPT}})$.*

Proof. Again we consider iterations where the population of Global SEMO_{alt} contains a solution x with $LP(x) = LP(0^n) - |x|_1$ such that each optimal fractional vertex cover assigns 1/2 to each non-isolated vertex of $G(x)$.

Let X denote the set of non-isolated vertices in $G(x)$, let $S \subseteq X$ be any minimum vertex cover of $G(x)$, and let $T = X \setminus S$. Observe that T is an independent set and that $|T| < |S|$, otherwise assigning 1 to each vertex of S and 0 to each vertex of T would yield a fractional vertex cover of cost less than $1/2 \cdot |X|$. Let $\text{OPT}' = \text{OPT} - |x|_1$, i.e. the size of minimum vertex covers of $G(x)$. Let $s_1, \dots, s_{\text{OPT}'}$ and $t_1, \dots, t_{|T|}$ be any two numberings of the vertices in S and T , respectively.

With probability $\Omega(1/n)$ Global SEMO_{alt} selects the solution x and applies the mutation that flips bits corresponding to non-isolated vertices of $G(x)$ with probability 1/2. With probability $\Omega((1/4)^{(1 - \epsilon) \cdot \text{OPT}'})$ all bits corresponding to $s_1, \dots, s_{\lceil (1 - \epsilon) \cdot \text{OPT}' \rceil}$ are flipped and those corresponding to t_1, \dots, t_α , with $\alpha = \min\{|T|, \lceil (1 - \epsilon) \cdot \text{OPT}' \rceil\}$, are not flipped. With probability greater than 1/2 the mutation flips bits of at least as many of the remaining vertices of S as of the remaining vertices of T , since $|T| < |S|$. Thus with probability $\Omega(1/n \cdot (1/4)^{(1 - \epsilon) \cdot \text{OPT}'})$ the solution x is mutated into a solution x' that additionally selects subsets $S' \subseteq S$ and $T' \subseteq T$ with $|S'| \geq (1 - \epsilon) \cdot \text{OPT}' + |T'|$. Again this includes a factor of $\Omega(1)$ accounting for the probability that Global SEMO_{alt} does not flip bits corresponding to isolated vertices of $G(x)$.

We will now prove an upper bound of $(1 + \epsilon) \cdot \text{OPT}$ on the value of $|x'|_1 + 2 \cdot LP(x')$. Observe that $LP(x') \leq \text{OPT}' - |S'|$ since $S \setminus S'$ is a vertex cover of $G(x')$. We also use the fact that $|T'| \leq |S'| - (1 - \epsilon) \cdot \text{OPT}'$.

$$\begin{aligned} & |x'|_1 + 2 \cdot LP(x') \\ &= |x|_1 + |S'| + |T'| + 2 \cdot LP(x') \\ &\leq |x|_1 + |S'| + |T'| + 2 \cdot (\text{OPT}' - |S'|) \\ &\leq |x|_1 + |S'| + |S'| - (1 - \epsilon) \cdot \text{OPT}' + 2 \cdot \text{OPT}' - 2 \cdot |S'| \\ &= |x|_1 + (1 + \epsilon) \cdot \text{OPT}' \\ &= |x|_1 + (1 + \epsilon) \cdot (\text{OPT} - |x|_1) \\ &\leq (1 + \epsilon) \cdot \text{OPT} \end{aligned}$$

Should a solution $y \in P$ dominate x' then this would imply $|y|_1 + 2 \cdot LP(y) \leq |x'|_1 + 2 \cdot LP(x')$. Thus after expected $O(n^3 + n \cdot 4^{(1 + \epsilon) \cdot \text{OPT}'})$ steps the population contains a solution x' with $|x'|_1 + 2 \cdot LP(x') \leq (1 + \epsilon) \cdot \text{OPT}$.

Ending the proof we show that such a solution leads to a $(1 + \epsilon)$ -approximate vertex cover in expected polynomial time. Let $y \in P$ be a solution with minimum value of $LP(y)$ under the constraint that $|y|_1 + 2 \cdot LP(y) \leq (1 + \epsilon) \cdot \text{OPT}$. If $LP(y) = 0$ then y is a $(1 + \epsilon)$ -approximate vertex cover. Otherwise there exists at least one vertex v that has value at least 1/2 in some optimal fractional vertex cover of $G(y)$. With probability $\Omega(1/n^2)$ the solution y is selected for mutation and exactly the bit corresponding to v is flipped, producing the solution y' .

Clearly $|y'|_1 = |y|_1 + 1$ and $LP(y') \leq LP(y) - 1/2$. Thus

$$|y'|_1 + 2 \cdot LP(y') \leq |y|_1 + 2 \cdot LP(y) \leq (1 + \epsilon) \cdot \text{OPT}.$$

Since y' fulfills the constraint and $LP(y') < LP(y)$ no solution in P can dominate y' since that solution would have been chosen in place of y . Thus with probability $\Omega(1/n^2)$ the minimum value of $LP(y)$ among solutions y that fulfill $|y|_1 + 2 \cdot LP(y) \leq (1 + \epsilon) \cdot \text{OPT}$ is decreased by at

least $1/2$. Since $0 \leq LP(y) \leq OPT$ the expected number of steps (from the point that x' was introduced) until the population contains a $(1 + \epsilon)$ -approximate vertex cover is bounded by $O(OPT \cdot n^2)$. Hence the total expected number of iterations of Global SEMO_{alt} until the population contains a $(1 + \epsilon)$ -approximate vertex cover is bounded by $O(n^2 \cdot \log n + OPT \cdot n^2 + n \cdot 4^{(1-\epsilon) \cdot OPT})$. \square

5. CONCLUSION

We have introduced the notion of evolutionary FPT algorithms to examine how the runtime of search heuristics depend on structural properties of a given problem. Using this approach we have examined the runtime and approximation behavior of evolutionary algorithms with respect to the value of an optimal solution. Our analyses on different multi-objective models show that additional criteria such as minimizing the number of uncovered edges or the value of a fractional solution for the uncovered part of the graph lead to a kernelization of the problem. Adding a random search component to the evolutionary algorithm by using the alternative mutation operator, we have shown that this gives evolutionary FPT algorithms.

There are several topics for future research. On the one hand, it seems to be interesting to analyze search heuristics in dependence of a given parameter on some other problems as well. The parameter can be the value of an optimal solution as considered in this paper but also a parameter which restricts the given input to certain classes of the problem. Examples include CLUSTER EDITING and 3-HITTING SET, both are FPT when parameterized by solution size, as well as MAXIMUM KNAPSACK parameterized by the capacity of the knapsack. Additionally, many graph problems, such as INDEPENDENT SET or DOMINATING SET, are FPT when parameterized by the treewidth of the input graph. Showing that an evolutionary algorithm profits from small values of treewidth might be a rather challenging problem, as the FPT algorithms for the two mentioned problems employ dynamic programming.

On the other hand, the use of the ILP relaxation as the second criteria to guide the search process may be of independent interest and we expect this criteria to be applicable for other problems as well.

References

- [1] M. L. Balinski. On maximum matching, minimum covering and their connections. In *Proceedings of the Princeton symposium on mathematical programming*, pages 434–445, 1970.
- [2] J. Chen, I. A. Kanj, and W. Jia. Vertex cover: Further observations and further improvements. *J. Algorithms*, 41(2):280–301, 2001.
- [3] J. Chen, I. A. Kanj, and G. Xia. Improved parameterized upper bounds for vertex cover. In R. Kralovic and P. Urzyczyn, editors, *MFCS*, volume 4162 of *Lecture Notes in Computer Science*, pages 238–249. Springer, 2006.
- [4] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, 2. edition, 2001.
- [5] B. Doerr, E. Happ, and C. Klein. Crossover can provably be useful in evolutionary computation. In C. Ryan and M. Keijzer, editors, *GECCO*, pages 539–546. ACM, 2008.
- [6] R. G. Downey and M. R. Fellows. *Parameterized Complexity (Monographs in Computer Science)*. Springer, November 1998.
- [7] V. Feldman. Evolvability from learning algorithms. In R. E. Ladner and C. Dwork, editors, *STOC*, pages 619–628. ACM, 2008.
- [8] T. Friedrich, J. He, N. Hebbinghaus, F. Neumann, and C. Witt. Analyses of simple hybrid algorithms for the vertex cover problem. *Evolutionary Computation*, 17(1), 2009. To appear, Conference version appeared in Proc. of IEEE CEC 2007.
- [9] T. Friedrich, N. Hebbinghaus, F. Neumann, J. He, and C. Witt. Approximating covering problems by randomized search heuristics using multi-objective models. In H. Lipson, editor, *GECCO*, pages 797–804. ACM, 2007.
- [10] O. Giel and I. Wegener. Evolutionary algorithms and the maximum matching problem. In *Proc. of STACS 2003*, volume 2607 of *LNCS*, pages 415–426, 2003.
- [11] G. L. Nemhauser and L. E. Trotter. Vertex packings: Structural properties and algorithms. *Mathematical Programming*, 8:232–248, 1975.
- [12] F. Neumann. Expected runtimes of a simple evolutionary algorithm for the multi-objective minimum spanning tree problem. *European Journal of Operational Research*, 181(3):1620–1629, 2007.
- [13] F. Neumann and I. Wegener. Randomized local search, evolutionary algorithms, and the minimum spanning tree problem. *Theor. Comput. Sci.*, 378(1):32–40, 2007.
- [14] P. S. Oliveto, J. He, and X. Yao. Evolutionary algorithms and the vertex cover problem. In *Proc. of CEC 2007*, IEEE Press, pages 1870–1877, 2007.
- [15] P. S. Oliveto, J. He, and X. Yao. Analysis of population-based evolutionary algorithms for the vertex cover problem. In *Proc. of CEC 2008*, IEEE Press, pages 1563–1570, 2008.
- [16] J. Scharnow, K. Tinnefeld, and I. Wegener. The analysis of evolutionary algorithms on sorting and shortest paths problems. *Journal of Mathematical Modelling and Algorithms*, pages 349–366, 2004.
- [17] L. G. Valiant. Evolvability. In L. Kucera and A. Kucera, editors, *MFCS*, volume 4708 of *Lecture Notes in Computer Science*, pages 22–43. Springer, 2007. Journal version appears in the Journal of ACM.
- [18] C. Witt. Worst-case and average-case approximations by simple randomized search heuristics. In *Proc. of STACS 2005*, volume 3404 of *LNCS*, pages 44–56, 2005.