UNIVERSITE PARIS DAUPHINE

Laboratoire d'Analyse et Modélisation de Systèmes pour l'Aide à la Décision

CNRS UMR 7024

CAHIER DU LAMSADE

280

Novembre 2008

Efficient approximation of MIN COLORING by moderately exponential algorithms

Nicolas Bourjeois, Bruno Escoffier, Vangelis Th. Paschos

CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE

Efficient approximation of MIN COLORING by moderately exponential algorithms

Nicolas Bourgeois   Bruno Escoffier   Vangelis Th. Paschos
LAMSADE, CNRS UMR 7024 and Université Paris-Dauphine, France
{bourgeois,escoffier,paschos}@lamsade.dauphine.fr

November 8, 2008

Abstract

We present in this note a rather new way to cope with polynomial inapproximability of **NP**-hard problems. We study approximation of MIN COLORING by moderately exponential time algorithms, able to achieve approximation ratios unachievable in polynomial time for MIN COLORING by algorithms with provably upper complexity bounds, better than those of exact resolution.

1 Introduction

Consider a graph $G(V, E)$ of order n. In the minimum graph-coloring problem, denoted by MIN COLORING, we wish to color V with as few colors as possible so that no two adjacent vertices receive the same color. The input-graph G is said to be k-colorable if there exists a coloring of V with k colors. The chromatic number $\chi(G)$ of G is the minimum k such that G is k-colorable (in other words, $\chi(G)$ is the value of an optimal coloring of G). By its definition, a color is an independent set; so, finding a minimum coloring amounts to determining a minimum partition of V into independent sets.

We present in this paper a rather new way to cope with polynomial inapproximability of **NP**-hard problems, namely by devising moderately exponential time algorithms, able to achieve approximation ratios unachievable in polynomial time for them by algorithms with provably upper complexity bounds, better than those of exact resolution.

Let $\mathtt{A}$ be an approximation algorithm for an **NP**-hard minimization (resp., maximization) problem Π, let $m(I, S)$ be the value of the solution S provided by $\mathtt{A}$ on an instance I of Π, and $\mathrm{opt}(I)$ be the value of the optimal solution for I. Finally, let $\omega(I)$ be the value of a worst solution of Π defined as the value of an optimal solution for $\bar{\Pi}$, the combinatorial problem having the same constraints as Π but instead of minimizing (resp., maximizing) the objective function of Π we wish to maximize (resp., minimize) it. For MIN COLORING, $\omega(G) = n$, since, coloring any vertex of G with its own color produces a feasible coloring. Then, if a feasible coloring of size k is produced for G, the quantity $n - k$ is exactly the number of colors left unused. Hence, MIN COLORING has a natural interpretation as the problem of maximizing the number of unused colors. This formulation is obviously equivalent from an exact viewpoint, but leads to different approximation properties. The *standard-approximation ratio* $\rho_{\mathtt{A}}(I)$ of $\mathtt{A}$ is defined by $\rho_{\mathtt{A}}(I) = m(I, S)/\mathrm{opt}(I)$, while the *differential-approximation ratio* $\delta_{\mathtt{A}}(I)$ of $\mathtt{A}$ on I is defined as $\delta_{\mathtt{A}}(I) = (\omega(I) - m(I, S))/(\omega(I) - \mathrm{opt}(I))$. In other words, studying MIN COLORING under the differential ratio is exactly the same as studying the problem of maximizing the number of unused colors (with the standard ratio). For both standard and differential ratios, the closer to 1, the better the approximability quality of an algorithm. In general, no apparent links exist between standard and differential approximations in the case of minimization problems, in the sense that there is no evident transfer of a positive, or negative, result from one paradigm to the other. Hence a "good" differential approximation result does not signify anything for the behavior of the approximation algorithm studied when dealing with the standard framework and vice-versa.

For the standard approximation, MIN COLORING is approximable within approximation ratio $O(n \log^2 \log n / \log^3 n)$ ([9]), while it cannot be approximated within ratio $n^{1-\varepsilon}$, for any $\varepsilon > 0$, assuming $\mathbf{P} \neq \mathbf{NP}$ ([13]). On the other hand, in the differential paradigm, MIN COLORING is approximable in polynomial time within differential ratio 289/360 ([10]) but it is **DAPX**-complete ([1]), so no polynomial time differential approximation schema can solve it, unless $\mathbf{P} = \mathbf{NP}$ (**DAPX** is the class of problems approximable in polynomial time within constant differential ratios).

Dealing with the exact resolution of MIN COLORING by (exponential) algorithms with provably upper running-time bounds, to the best of our knowledge, the fastest known algorithms have running times $O^*(2.2461^n)$ in general graphs, if only polynomial space is allowed ([3, 12]), while, if exponential space $O^*(2^n)$ is allowed, time-complexity becomes $O^*(2^n)$. These results improve the $O^*(2.415^n)$ result by [7]. Improved worst case bounds have been achieved for determining whether a graph is 3-colorable or not (3-COLORING), and whether is it 4-colorable or not (4-COLORING). The best current known such bounds are respectively $O^*(1.329^n)$ ([6]) and $O^*(1.751^n)$ ([5]). Notation $O^*(\cdot)$ is as $O(\cdot)$ where polynomial terms are omitted.

In what follows, we devise approximation algorithms for MIN COLORING achieving approximation ratios "forbidden" in polynomial time, for both the standard and the differential approximation paradigms. We propose interesting tradeoffs between running time and approximation ratios. Let us note that devising exponential approximation algorithms for MIN COLORING has been marginally handled by [3]. Note also that, in the same setting, we handle in [4] MAX INDEPENDENT SET, MIN VERTEX COVER, MAX CLIQUE, MAX BIPARTITE SUBGRAPH and MAX SET PACKING. Finally, we provide a worst case complexity analysis for the exact algorithm by [2]. This is one of the oldest algorithms for MIN COLORING and, to the best of our knowledge its complexity has never been analyzed.

Before concluding this section, let us make some remarks on what kind of results can be expected in the area of low-complexity non-polynomial approximation.

First, it is worth noticing that, since a resolution of MIN COLORING within a ratio smaller than 4/3 would allow to solve the 3-COLORING problem, the existence of subexponential approximation algorithms (within ratio better than 4/3) is quite improbable since it would imply that 3-COLORING can be solved in subexponential time, contradicting so the "exponential time hypothesis". We conjecture that the same holds for any *constant* ratio. On the other hand, we will see that *any* unbounded ratio can be achieved in subexponential time, leading to an interesting dichotomy.

Secondly, dealing with good approximation ratios, if one is able to optimally solve a problem in time $O^*(\gamma^n)$, ideally one should try to find a kind of "exponential approximation schema", i.e., for any $\varepsilon > 0$ a $(1+\varepsilon)$-approximation algorithm (for a minimization problem) working in time $O^*(\gamma_\varepsilon^n)$, with $\gamma_\varepsilon < \gamma$. This kind of schemata are easily achievable for classical problems such as MAX INDEPENDENT SET or MIN VERTEX COVER, see [4]. For instance, for MAX INDEPENDENT SET, one can find a r-approximate solution in time $O^*(\gamma^{rn})$, for any $r \leqslant 1$. For MIN COLORING, however, the situation seems to be different. Indeed, solution of MIN COLORING within standard ratio smaller than $(k+1)/k$ allows to solve the problem of determining whether the input graph is k-colorable or not. Hence, such an approximation schema would imply the existence, for *any* k, of an exact algorithm running in time $O^*(\gamma_k^n)$, where $\gamma_k < 2$ (exponential space) or smaller than 2.2461 (polynomial space). Until now, this is far from being true for any k (it holds only for small values of k). Hence, it seems quite hard (if not unreasonable) to obtain such a schema.

Interestingly enough, when tackling the MAX UNUSED COLORS problem (or equivalently MIN COLORING with the differential ratio), we will see in Section 3 that we are able to obtain two nice properties:

1. dealing with exponential space algorithms, we obtain such a schema, i.e., a $(1-\varepsilon)$-differential approximation algorithm in time (and space) $O^*(2^{ne^{-0.12\varepsilon}})$ (Proposition 7);

2. dealing with polynomial space, we obtain even better: a $(1-\varepsilon)$-differential approximation algorithm in time $O^*(2^n)$, for any $\varepsilon > 0$ (Proposition 6).

This latter result is quite surprising since one should expect that when $\varepsilon \to 0$, the exponential basis of the running time should tend to that of the exact computation; here, the exponential basis *does not* depend on ε, and is much lower than the one of exact computation!

2 Standard approximation

2.1 Splitting the input graph

We give in this section a simple approximation algorithm achieving constant approximation ratio. This algorithm, denoted by `C1` works as follows: arbitrarily partition the vertices of the input-graph G into q subsets of size (except, eventually, for the last one) n/q, run some exact coloring-algorithm `EC` on any of the induced subgraphs $G_1, \ldots, G_q$ using a new set of colors for each of them and take the union of the colors used as solution for G. Then, the following result holds.

Proposition 1. *Assume that an algorithm* `EC` *computes an optimal solution for* MIN COLORING *with running time $O^*(\alpha^n)$ and space requirements $O^*(\beta^n)$. Then, for any $q \in \mathbb{N}$ it is possible to compute a q-approximation for* MIN COLORING *with running time $O^*(\alpha^{n/q})$ and space requirements $O^*(\beta^{n/q})$.*

Proof. Obviously $\chi(G_i) \leqslant \chi(G)$. If we use distinct colors for each subgraph, then the size of the final coloring produced by `C1` is bounded above by $\sum_{i=1}^{q} \chi(G_i) \leqslant q\chi(G)$, qed. ▮

Table 1 gives examples of running times of `C1` for some values of q.

q	1	2	3	5	10
Polynomial space	2.246^n	1.499^n	1.310^n	1.176^n	1.084^n
Exponential space	2^n	1.414^n	1.260^n	1.149^n	1.072^n
4-colorable graphs	1.751^n	1.324^n	1.206^n	1.119^n	1.058^n
3-colorable graphs	1.329^n	1.153^n	1.099^n	1.074^n	1.029^n

Table 1: Running times of Algorithm `C1` for some values of q.

2.2 Excavating independent sets

Assume that we have an exact algorithm `EC` for MIN COLORING, an exact algorithm `ES` for MAX INDEPENDENT SET, and consider the following classical excavation algorithm (see, for instance [11]), denoted by `C2`, parameterized by a nonnegative integer k:

1. run k times Algorithm `ES`, at each time color the vertices of the excavated independent set with an unused color and remove them from G;

2. run EC in the surviving graph;

3. output the set of colors computed at steps 1 and 2.

The following proposition holds about Algorithm C2.

Proposition 2. *([3]) Assume we use an exact algorithm for* MAX INDEPENDENT SET *that runs in time $O^*(\gamma^n)$ and an exact algorithm for* MIN COLORING *that runs in time $O^*(\alpha^n)$. Then, denoting $\mu = k/\chi(G)$, C2 is a $(1+\mu)$-approximation algorithm for* MIN COLORING *that works in polynomial space and in time $O^*(\gamma^n + \alpha^{ne^{-\mu}})$.*

As it is pointed out in [3] Proposition 2 also holds when one wishes to determine a coloring verifying its statement. Finally, for its proof, just remark that the number of colors used by Algorithm C2 is bounded above by $k + \chi(G)$, and that the size of the remaining graph is at most $(1 - 1/\chi(G))^k n < e^{-k/\chi(G)} n$.

In Table 2, running times derived from Proposition 2 are given, for some ratios's values (using an exact algorithm in $O^*(1.22^n)$ for MAX INDEPENDENT SET by [8] and the exact coloring algorithms given in the introduction).

$1+\mu$	4/3	3/2	2	3	5
Polynomial space	1.786^n	1.634^n	1.347^n	1.22^n	1.22^n
Exponential space	1.643^n	1.523^n	1.290^n	1.22^n	1.22^n
4-colorable graphs		1.405^n	1.22^n	1.22^n	1.22^n
3-colorable graphs	1.226^n		1.22^n	1.22^n	1.22^n

Table 2: Running times of Algorithm C2 as derived from Proposition 2.

We now show how one can decrease running time claimed in Proposition 2, especially as μ grows. For this, we use the same algorithm modulo the fact that instead of an exact algorithm ES a low (exponential) complexity approximation algorithm is called for MAX INDEPENDENT SET in step 1 of Algorithm C2. Let us denote C2′ this new version, also parameterized by k in the first step. Such an algorithm can be, for instance, the algorithm by [4] that guarantees approximation ratio $r \leqslant 1$ for MAX INDEPENDENT SET with complexity $O^*(\gamma^{rn})$ (assuming that MAX INDEPENDENT SET can be optimally solved in $O^*(\gamma^n)$). This modification allows to achieve a tradeoff between the running time of step 1 (faster in C2′, since only an approximate solution is computed) and of step 2 (slower in C2′, since the size of the remaining graph is larger).

Proposition 3. *Denoting $\mu = k/\chi(G)$, for every $1 \geqslant r > 0$, C2′ is a $(1+\mu)$-approximation algorithm for* MIN COLORING *that works in polynomial space and with running-time $O^*(\gamma^{rn} + \alpha^{ne^{-r\mu}})$.*

Proof. Remark that each time a subgraph is removed the size of the graph is reduced by a multiplicative factor $(1-r/\chi)$. Hence, after k steps, the size of the graph is at most $(1-r/\chi)^k n < e^{-rk/\chi} n$. The running time follows. For the approximation ratio, just remark that the bound $k + \chi(G)$ on the size of the solution built still holds. ▮

In Table 3, complexities derived from Proposition 3 are shown. For every value of the ratio $1+\mu$, the optimal approximation ratio r used to solve MAX INDEPENDENT SET is computed. For instance the 5-approximation algorithm for 4-COLORING uses a 0.456-approximation algorithm for MAX INDEPENDENT SET.

Note that the results of Propositions 2 and 3 can be restated as follows (by fixing a priori the size of the surviving graph after execution of step 1 of Algorithm C2):

$1+\mu$	$4/3$	$3/2$	2	3	5
Polynomial space	1.786^n	1.634^n	1.347^n	1.174^n	1.108^n
Exponential space	1.643^n	1.523^n	1.290^n	1.163^n	1.102^n
4-colorable graphs		1.405^n	1.22^n	1.149^n	1.095^n
3-colorable graphs	1.226^n		1.151^n	1.107^n	1.072^n

Table 3: Running times of Algorithm `C2` as derived from Proposition 3.

- *Proposition 2:* for any $\rho \geqslant 1$ an approximation ratio $1 + \lceil \rho\chi(G) \rceil / \chi(G)$ is achieved in $O^*(\gamma^n + 2^{e^{-\rho}n})$;
- *Proposition 3:* for any $\rho \geqslant 1$ and any $r \leqslant 1$, an approximation ratio $1 + \lceil \rho\chi(G) \rceil / \chi(G)$ is achieved in $O^*(\gamma^{rn} + 2^{e^{-r\rho}n})$.

Let us now compare Algorithms `C2` and `C2`$'$, i.e., let us determine the range of ratios for which it is interesting to use an r_μ-approximation algorithm for MAX INDEPENDENT SET instead of an exact one. This holds for $\mu > \mu^* = \log\log\alpha - \log\log\gamma$. Indeed, it is easy to see that γ^{rn} increases with r while $\alpha^{ne^{-r\mu}}$ decreases. So, it is interesting to bound beyond the complexity of the first term if and only if it is the dominating term for $r = 1$, i.e., if:

$$\gamma \geqslant \alpha^{e^{-\mu}} \Leftrightarrow e^{\mu}\log\gamma \geqslant \log\alpha \Leftrightarrow \mu \geqslant \log\log\alpha - \log\log\gamma$$

For instance, when only polynomial space is allowed, $\mu^* = 1.403$; hence, for $1+\mu = 4/3, 3/2, 2, \ldots$ ($\mu \leqslant \mu^*$), the values are the same in Tables 1 and 2, while for $1+\mu = 3, 5, \ldots$ ($\mu > \mu^*$), the values are better in Table 2. When exponential space is allowed, $\mu^* = 1.249$. For 3-colorable graphs, $\mu^* = 1.009$, while for those that are 4-colorable, $\mu^* = 0.358$.

We conclude this section by showing that any unbounded ratio can be reached in subexponential time (as mentioned in Section 1).

Proposition 4. *For any $\rho_n \to \infty$, it is possible to compute a k-coloring whose size is bounded below by $\rho_n\chi(G)$, with infra-exponential running time $O^*(\gamma^{n/\sqrt{\rho_n}} + \alpha^{ne^{-\sqrt{\rho_n}}})$.*

Proof. Fix $q = \sqrt{\rho_n}$. Compute an $1/q$-approximation for MAX INDEPENDENT SET and use a new color for the independent set computed. As before, the size of the surviving graph G' is at most $(1 - 1/(q\chi(G)))n$. Iterate the process until $|G'| < ne^{-\sqrt{\rho_n}}$, and use some exact algorithm to color (the finally surviving) G'.

Complexity bound is obvious. Let us consider the size of the solution so-computed. If j independent sets have been excavated,

$$\frac{|G'|}{n} \leqslant \left(1 - \frac{1}{q\chi(G)}\right)^j < e^{-\frac{j}{q\chi(G)}}$$

Consequently, since excavation stops at the first time this bound is reached, the following holds:

$$\begin{aligned} \frac{j}{q\chi(G)} &\leqslant \sqrt{\rho_n} \\ k \leqslant j + \chi(G) &\leqslant \chi(G)\left(1 + \rho_n\right) \simeq \rho_n\chi(G) \end{aligned}$$

as $\rho_n \to \infty$. ∎

3 Differential approximation

As stated informally in Section 1, we will show in what follows that any differential ratio $1-\varepsilon$ is achievable within non-trivial running time, both in polynomial and exponential space. More precisely:

- in Proposition 6, we devise a $(1-\varepsilon)$-differential approximation algorithm running in time $O^*(2^n)$, for any $\varepsilon > 0$ (while the best known exact algorithm using polynomial space has time-complexity $O^*(2.2461^n)$);
- when exponential space is allowed, we obtain a $(1-\varepsilon)$-differential approximation algorithm in time (and space) $O^*(2^{ne^{-0.12\varepsilon}})$, for any $\varepsilon > 0$ (Proposition 7).

Consider first the following proposition that establishes an exact computation result for MIN COLORING in a particular class of graphs. In what follows, using standard notations ([2]), we denote by $\alpha(G)$ the cardinality of a maximum independent set of a graph G.

Proposition 5. *Consider a graph G with $\alpha(G) \leqslant k$, for some bounded constant k. Then, it is possible to solve* MIN COLORING *with polynomial space and with running time $O(2^n n^k)$.*

Proof. Note first that MIN COLORING is equivalent to compute, for any $X \subset V$, the number $a(X)$ of independent sets that do not intersect X ([12]). If we denote by $\iota(S, G)$ the number of independent sets of $G[S]$, running time of such a computation is bounded above by:

$$\sum_{S \subset V} \iota(S, G) \leqslant \sum_{j \leqslant n} \binom{n}{j} n^{\alpha(G)} \leqslant 2^n n^k$$

qed. ∎

Based upon Proposition 5, we show in the following result how one can approximate MIN COLORING within any differential approximation ratio strictly less than 1 in time $O^*(2^n)$ (that is the same as that of the exact computation for MIN COLORING), but with polynomial space (recall that exact computation of in time $O^*(2^n)$ requires exponential space ([3, 12])).

Proposition 6. *For any $\varepsilon > 0$, it is possible to compute a $(1-\varepsilon)$-differential approximation of* MIN COLORING *with polynomial space and with running time $O(2^n n^{10/\varepsilon}) = O^*(2^n)$.*

Proof. Consider the following algorithm, denoted by `DC1` in what follows:

1. fix some μ (we will see below that optimal value for μ is 0.107); fix $k = (1-2\mu)n$ and produce a coloring of size greater than k using the following procedure:
   (a) fix any subset $K \subset V$ of size k and color each vertex of K with an unused color;
   (b) run an exact polynomial space algorithm to color $V \setminus K$;
   (c) store one among the best solutions produced by steps 1a and 1b, denoted by C_1;
2. fix an $\varepsilon > 0$ and, as long as possible, successively excavate independent sets of size at least $1/\mu\varepsilon$, color the vertices of each of them by a new color and remove them from the graph;
3. use the algorithm from Proposition 5 ([12]) to compute some solution of MIN COLORING on the remaining graph; let C_2 be the coloring of G produced by steps 2 and 3;
4. output the best solution among C_1 and C_2.

Step 1 of Algorithm DC1 has running time (using Stirling's formula):

$$\binom{n}{k} \times 2.246^{n-k} = O^*\left(\left(\frac{2.246^{2\mu}}{(2\mu)^{2\mu}(1-2\mu)^{1-2\mu}}\right)^n\right)$$

If $\mu \leqslant 0.107$, this term is $o(2^n)$.

On the other hand, complexity of step 2 is at most $O^*(n^{(1/\mu\varepsilon)+1})$ and of step 3 $O^*(2^n n^{1/\mu\varepsilon})$.

Assume first $\chi(G) \geqslant (1-\mu)n$. Then, at least $(1-2\mu)n$ colors are singletons. This means that step 1 of DC1 determines an optimal solution.

On the other hand, if $\chi(G) \leqslant (1-\mu)n$, then C_2 has size at most $n\mu\varepsilon + \chi(G)$, since at most $n\mu\varepsilon$ independent sets (of size $\mu\varepsilon$) are removed. Then, the following holds:

$$\frac{n-|C_2|}{n-\chi(G)} = 1 - \frac{|C_2|-\chi(G)}{n-\chi(G)} \geqslant 1 - \frac{\mu\varepsilon n}{\mu n} = 1-\varepsilon$$

and the proof of the proposition is completed. ▮

We now conclude this section by showing that, if exponential space is allowed, running time claimed in Proposition 6 can be improved.

Proposition 7. *For any $\varepsilon > 0$, it is possible to compute a $(1-\varepsilon)$-differential approximation to* MIN COLORING *with running time and space $O^*(2^{ne^{-0.12\varepsilon}})$.*

Proof. Consider the following algorithm:

1. fix $k = (1-2\mu)n$ and produce a coloring of size greater than k using steps 1a and 1b of Algorithm DC1 except that, at step 1b we use the algorithm by [12] instead of some polynomial space one; let C_1 be some coloring of minimum cardinality so produced;
2. use the excavation algorithm C2 of Section 2.2 to compute some coloring C_2 that is a $\rho = (\varepsilon\mu/(1-\mu))$-standard approximation for MIN COLORING;
3. output the best among solutions C_1 and C_2.

Recall that, as discussed after Proposition 3, the ratio we get in time $O^*(\gamma^n + \alpha^{ne^{-\rho}})$ is:

$$1 + \frac{\lceil \rho\chi(G) \rceil}{\chi(G)} \leqslant 1 + \rho + \frac{1}{\chi(G)}$$

Step 1 has running time and space requirements:

$$\binom{n}{k} \times 2^{n-k} \leqslant \left(\frac{4^{\mu}}{(2\mu)^{2\mu}(1-2\mu)^{1-2\mu}}\right)^n$$

If $\mu \leqslant 0.107$, this term is $o(2^n)$.

On the other hand, complexity of step 2 is at most (with $\mu \leqslant 0.107$) $O^*(2^{ne^{-\mu\varepsilon/(1-\mu)}}) = O^*(2^{ne^{-0.12\varepsilon}})$.

As in the proof of Proposition 6, if $\chi(G) \geqslant (1-\mu)n$, at least $(1-2\mu)n$ colors are singletons and step 1 of the algorithm works optimally. Else ($\chi(G) \leqslant (1-\mu)n$):

$$\frac{n-|C_2|}{n-\chi(G)} = 1 - \frac{|C_2|-\chi(G)}{n-\chi(G)} \geqslant 1 - \frac{\chi\frac{\varepsilon\mu}{1-\mu}+1}{n-\chi(G)} \geqslant 1 - \frac{\mu\varepsilon n + 1}{\mu n} = 1 - \varepsilon + \frac{1}{\mu n}$$

qed. ▮

4 Analysis of Berge algorithm

The MIN COLORING exhaustive algorithm of Berge ([2]) is one of the oldest exact algorithms for this problem that works as follows. Fix an optimal coloring C^* of G and a pair (a, b) of non-adjacent vertices of G. We branch on (a, b): either they are in different colors of C^*, in this case we may add, w.l.o.g., (a, b) to E; or they are in the same color of C^* and, in this case, we can contract both a and b into a single vertex, namely ab, whose neighbors verify $\Gamma(ab) = \Gamma(a) \cup \Gamma(b)$. Leaves of the so-built search-tree are cliques, and smallest ones derive optimal colorings.

Berge algorithm produces any feasible coloring of G exactly once. Assume a contrario that some coloring is represented by two distinct leaves A and B of the search-tree. Then, there exists a node R that is the root of the minimal subtree A and B belong to. At step R, some edge is examined, and only one choice may lead to A and B, that is a contradiction with minimality of the subtree considered.

The complexity of Berge Algorithm is the number of feasible colorings, that cannot exceed the number of partitions of V, i.e., the Bell number of G. This number is $B_n = n!/log(n)^n R(n)$ where $R(n)$ is infra-exponential. This bound is reached if G is an independent set on n vertices.

Let us note that, for any $k \in \mathbb{N}$, Berge Algorithm can be used to verify if G is k-colorable, with running time $O^*(k^n)$. Indeed, in order to do this, one can verify at each step if there exists a $(k+1)$-clique formed. If yes, one does not have to examine the subtree rooted at this clique. Complexity is then bounded above by:

$$\binom{n}{k+1} \times \sum_{i=1}^{k} \frac{i^n}{i!} \leqslant n^{k+2} k^n$$

Remark 1. If $a, b \in V$ are such that $b \notin \Gamma(a)$ and $\Gamma(b) \subset \Gamma(a)$, then, there exists an optimal solution to MIN COLORING such that a and b have the same color. ▮

Proposition 8. *Berge Algorithms allows us to find an optimal solution to* MIN COLORING *with running time* $O^*(1.466^{|\bar{E}|})$*, where* $\bar{G}(V, \bar{E})$ *is the complement of* $G(V, E)$*.*

Proof. Any time we branch on an edge (a, b), we have the choice either to add a and b to colors of the optimal solution C^*, i.e., to decrease $|\bar{E}|$ by 1, or to contract a and b into a single vertex ab. In this case, denoting by E' the edges of the graph resulting by the contraction and by $\bar{E}'$ the corresponding anti-edges, the following holds for any other pair of vertices v and w:

- $(v, w) \in \bar{E} \Leftrightarrow (v, w) \in \bar{E}'$;
- $(v, a) \notin \bar{E}$ and $(v, b) \notin \bar{E} \Rightarrow (v, ab) \notin \bar{E}'$ (no change);
- $(v, a) \notin \bar{E}$ but $(v, b) \in \bar{E} \Rightarrow (v, ab) \notin \bar{E}'$ (one edge is deleted);
- $(v, a) \in \bar{E}$ and $(v, b) \in \bar{E} \Rightarrow (v, ab) \in \bar{E}'$ (two edges are replaced by a single one).

Consequently:

$$\left|\bar{E}'\right| = \left|\bar{E}\right| - \left|\Gamma_{\bar{G}}(a) \cup \Gamma_{\bar{G}}(b)\right| (+1) = \left|\bar{E}\right| - n + \left|\Gamma_G(a) \cap \Gamma_G(b)\right| + 1$$

Thanks to Remark 1, there exist two other distinct vertices c and d such that $c \notin \Gamma_G(a)$ and $d \notin \Gamma_G(b)$. Then, $|\Gamma_G(a) \cap \Gamma_G(b)| \leqslant n - 4$. Hence, we get:

$$T\left(\left|\bar{E}\right|\right) \leqslant T\left(\left|\bar{E}\right| - 1\right) + T\left(\left|\bar{E}\right| - 3\right)$$

that leads to the result claimed. ▮

Proposition 9. *Assume that an algorithm computes an optimal solution to* MIN COLORING *with running time $O^*(\alpha^{|\bar{E}|})$ and that an algorithm computes a maximum independent set with complexity $O^*(\gamma^n)$. Then, it is possible to find a $\chi(G) + 1$ coloring of G with running time $O^*(\alpha^{|\bar{E}|-(n(n+1)/2\chi^2)} + \gamma^n)$*

Proof. Assume, w.l.o.g., that G connected. Construct a maximum independent set S, whose size is, of course, at least $n/\chi(G)$, color its vertices with a new color, remove it from G and then run the exact algorithm claimed for MIN COLORING in the surviving graph. Since the complementary of S is a clique, we drastically decrease $\bar{E}$:

$$\begin{aligned} |\bar{E}'| &= |\bar{E}| - \bigcup_{s \in S} \Gamma_{\bar{G}}(s) = |\bar{E}| - \frac{|S|(|S|-1)}{2} - \bigcup_{s \in S} \Gamma_{\bar{G} \setminus S}(s) \\ &\leqslant |\bar{E}| - \frac{|S|(|S|+1)}{2} \end{aligned}$$

that leads to result claimed. ∎

Remark finally that it is possible, for any $k \in \mathbb{N}$, to compute a $1 + (k/\chi(G))$-approximation for MIN COLORING with running time $O^*(\alpha^{|\bar{E}|-(n^2(1-e^{-2(k+1)/\chi})/4\chi}) + \gamma^n)$.

Indeed, iterate k times the computation of a maximum independent set in the surviving graph. With some easy algebra one gets:

$$\sum_{i=0}^{k} \frac{(n^2\left(1 - \frac{1}{\chi(G)}\right)^{2i})}{\chi^2} = \frac{n^2}{2\chi^2} \times \frac{1 - \left(1 - \frac{1}{\chi(G)}\right)^{2k+2}}{1 - \left(1 - \frac{1}{\chi(G)}\right)^2} \geqslant \frac{n^2}{2} \times \frac{1 - e^{-\frac{2(k+1)}{\chi(G)}}}{2\chi(G) - 1}$$

References

[1] C. Bazgan, B. Escoffier, and V. Th. Paschos. Completeness in standard and differential approximation classes: Poly-(D)APX- and (D)PTAS-completeness. *Theoret. Comput. Sci.*, 339:272–292, 2005.

[2] C. Berge. *Graphs and hypergraphs.* North Holland, Amsterdam, 1973.

[3] A. Björklund and T. Husfeldt. Inclusion-exclusion algorithms for counting set partitions. In *Proc. FOCS'06*, pages 575–582, 2006.

[4] N. Bourgeois, B. Escoffier, and V. Th. Paschos. Efficient approximation by "low-complexity" exponential algorithms. Cahier du LAMSADE 271, LAMSADE, Université Paris-Dauphine, 2008. Available at `http://www.lamsade.dauphine.fr/cahiers/PDF/cahierLamsade271.pdf`.

[5] J. M. Byskov. Enumerating maximal independent sets with applications to graph colouring. *Oper. Res. Lett.*, 32(6):547–556, 2004.

[6] D. Eppstein. Improved algorithms for 3-coloring, 3-edge-coloring, and constraint satisfaction. In *Proc. Symposium on Discrete Algorithms, SODA'01*, pages 329–337, 2001.

[7] D. Eppstein. Small maximal independents sets and faster exact graph coloring. In Frank Dehne, Jörg-Rüdiger Sack, and Roberto Tamassia, editors, *Proc. Workshop on Algorithms and Data Structures, WADS'01*, volume 2125 of *Lecture Notes in Computer Science*, pages 462–470. Springer-Verlag, 2001.

[8] F. V. Fomin, F. Grandoni, and D. Kratsch. Measure and conquer: a simple $O(2^{0.288n})$ independent set algorithm. In *Proc. Symposium on Discrete Algorithms, SODA'06*, pages 18–25, 2006.

[9] M. M. Halldórsson. A still better performance guarantee for approximate graph coloring. *Inform. Process. Lett.*, 45(1):19–23, 1993.

[10] R. Hassin and S. Khuller. z-approximations. *J. Algorithms*, 41:429–442, 2001.

[11] D. S. Johnson. Approximation algorithms for combinatorial problems. *J. Comput. System Sci.*, 9:256–278, 1974.

[12] M. Koivisto. An $O^*(2^n)$ algorithm for graph coloring and other partitioning problems via inclusion-exclusion. In *Proc. FOCS'06*, 2006.

[13] D. Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. In *Proc. STOC'06*, pages 681–690, 2006.