Coordinated cutting plane generation via multi-objective separation

Edoardo Amaldi · Stefano Coniglio · Stefano Gualandi

Received: 23 June 2011 / Accepted: 14 September 2012 / Published online: 18 October 2012

E. Amaldi · S. Coniglio (✉) · S. Gualandi
Dipartimento di Elettronica ed Informazione, Politecnico di Milano,
Piazza L. da Vinci 32, 20133 Milano, Italy
e-mail: stefano.coniglio@mail.polimi.it

E. Amaldi
e-mail: amaldi@elet.polimi.it

Present Address:
S. Gualandi
Dipartimento di Matematica, Università degli Studi di Pavia, Via Ferrata 1, 27100 Pavia, Italy
e-mail: stefano.gualandi@unipv.it

1 Introduction

In cutting plane methods for integer programming, one of the most critical and crucial issues is that of generating the "best possible" set of cuts.

An iteration of state-of-the-art cutting plane algorithms, see, for instance, [1,4,3], typically consists of a first phase where a large number of cuts is generated according to a given criterion (e.g., cut violation) and a second phase where a subset of promising cuts is selected according to a *cut selection procedure* based on different cut quality measures. Frequently used measures include, among others, the orthogonal distance between the hyperplane associated to the cut and the optimal solution of the current relaxation, the cut sparsity, and a measure of parallelism between cuts. The latter measure aims at discarding cuts that are similar to those that were previously generated and, therefore, favors a form of diversity among the cutting planes.

Recently, diversification strategies have also been used at the cut generation stage for some families of valid inequalities. In [17], where a mixed-integer program (MIP) is proposed to separate rank-1 Chvátal–Gomory cuts, larger bound improvements are obtained by dropping the upper bounds on the multipliers. This additional freedom produces a beneficial diversification effect when breaking the ties between equivalent solutions of the separation problem. In [8], when optimizing over the rank-1 Split Closure by solving a MIP with a single parameter via bisection, the set of disjunctions is diversified by enforcing their partial orthogonality. In [7], the authors apply the lexi-cographic dual simplex method (rather than the standard dual one) when reoptimizing the linear programming (LP) relaxations in Gomory's cutting plane algorithm [19]. Their method produces sequences of solutions which are further away from each other (in Euclidean distance) than those obtained with the standard method and allows to close a larger fraction of the duality gap.

In this paper, we propose a lexicographic multi-objective scheme for cutting plane generation in which the cut violation and a suitable measure of diversity between cuts are simultaneously optimized. Specifically, we propose a separation problem where, among all the maximally violated valid inequalities of a given family, we generate a cut that is also undominated and maximally diverse w.r.t. the cuts that were pre-viously found. Since new cuts explicitly depend on the previous ones, we obtain a *coordinated cutting plane generation* scheme. The focus in this work is on valid inequalities with 0–1 coefficients in the left-hand side and a constant right-hand side, which encompasses families of inequalities such as clique and cut set inequalities (see [25] and the references therein) that are valid for many combinatorial optimization problems.

The paper is organized as follows. In Sect. 2, we review the main cut quality measures used in the literature and point out their advantages and disadvantages. In Sect. 3, we describe our choice of diversity measure and our lexicographic multi-objective cutting plane generation scheme. In Sect. 4, we address the critical issue of generating undominated cuts and propose a revised scheme that guarantees the generation of cuts which are as strong as possible. In Sect. 5, we report computational results obtained for the separation of stable set and cut set inequalities for, respectively, the max clique and min Steiner tree problems. We address a pure cutting plane setting for both problems and also a cut-and-branch one for max clique, comparing the results obtained with our revised separation scheme to those for the standard separation of maximally violated cuts which are undominated. Section 6 contains some concluding remarks and directions for future work.

A preliminary version of this work appeared in [2]. See also the PhD thesis [12] and its extended abstract [13].

2 Cut quality measures and cut selection procedures

Consider an integer program (IP)

$$\begin{aligned} \min\ & cx \\ \text{s.t.}\ & Ax \leq b \\ & x \in \mathbb{Z}^n_+ \end{aligned}$$

and the usual dual cutting plane algorithm where the integrality restrictions on x are dropped and a separation problem is solved to generate a valid inequality $\alpha x \leq \alpha_0$, with $\alpha \in \mathbb{Z}^n$ and $\alpha_0 \in \mathbb{Z}$ (for the sake of space, we will explicitly mention the case of valid inequalities $\alpha x \geq \alpha_0$ only when the direction of the inequality makes a difference). Let $\mathcal{C}$ be the family of valid inequalities under consideration, in short $(\alpha, \alpha_0) \in \mathcal{C}$. At each iteration, the current continuous relaxation of the problem, tightened with all the valid inequalities generated so far, is solved via linear programming. Let x^* be a corresponding optimal solution.

Assume that a set of valid inequalities has been generated but not yet introduced into the relaxation. Let $H = \{x \in \mathbb{R}^n : \alpha x = \alpha_0\}$ be the hyperplane corresponding to $\alpha x \leq \alpha_0$. A typical cut selection procedure works as follows. First, the cutting planes are sorted w.r.t. a measure of distance from x^* to H, often the Euclidean distance from x^* to its orthogonal projection onto H. Then, the cuts are added, according to that order, only if they meet a prescribed cut quality measure requirement and if they are not dominated by the previously added cuts. Recall the usual definition of cut domination [25]: for an arbitrary polyhedron $P \subseteq \mathbb{R}^n_+$, a cut $\alpha' x \leq \alpha'_0$ dominates another cut $\alpha'' x \leq \alpha''_0$ if there exists a scalar $u > 0$ such that $\alpha' \geq u\alpha''$ and $\alpha'_0 \leq u\alpha''_0$ and $(\alpha', \alpha'_0) \neq (u\alpha'', u\alpha''_0)$. A cut in some family $\mathcal{C}$ is said to be *strong* if there is no other cut in $\mathcal{C}$ that dominates it. Conversely, a cut is said to be *weak* if there is at least a cut in $\mathcal{C}$ by which it is dominated.

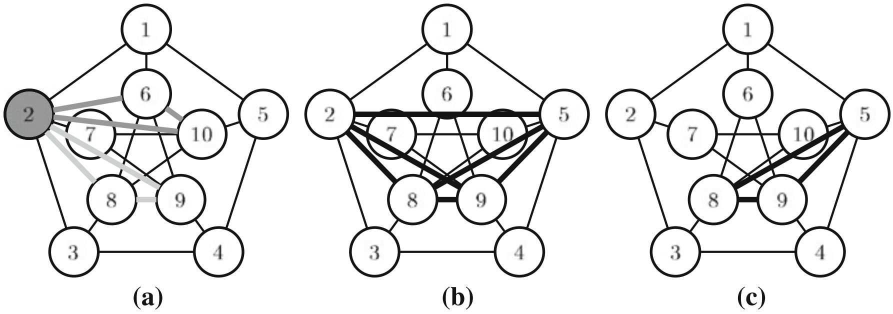

Fig. 1 Illustration of Example 1

2.1 Cut quality measures

The most relevant cut quality measures considered in the literature are based on distance, angle, and cut sparsity. We briefly recall them, pointing out their main advantages and disadvantages.

2.1.1 Distance measures

Definition 1 *(Cut violation)* Given a relaxation $P \subseteq \mathbb{R}^n_+$ with an optimal solution $x^* \in P$ and a cut $\alpha x \leq \alpha_0$, the *cut violation* is the quantity $\alpha x^* - \alpha_0$.

Definition 2 *(Cut depth)* Given a relaxation $P \subseteq \mathbb{R}^n_+$ with an optimal solution $x^* \in P$ and a cut $\alpha x \leq \alpha_0$, the *cut depth* is the Euclidean distance between x^* and its orthogonal projection onto H, namely, $\dfrac{\alpha x^* - \alpha_0}{\|\alpha\|_2}$, where $\|\alpha\|_2 = \sqrt{\sum_{j=1}^n \alpha_j^2}$.

The cut depth, which appears in [4] and is also referred to as "geometric distance" in [5] or as "efficacy" in [3] and [1], suffers from a serious drawback. Indeed, as illustrated in the following example for the max clique problem, it may favor the selection of strong or weak cuts, depending on the direction of the inequality.

Example 1 Consider the fractional clique polytope for a Petersen graph with 10 vertices, tightened with the stable set inequalities $x_2 + x_6 + x_{10} \leq 1$ and $x_2 + x_8 + x_9 \leq 1$. In Fig. 1a, the nonedges related to the stable sets involved in the two inequalities are indicated, respectively, in dark and light gray. The unique optimal solution of maximizing $\sum_{j=1}^n x_j$ over the polytope is $x^* = (\frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$. Consider the two maximally violated stable set inequalities $x_2 + x_5 + x_8 + x_9 \leq 1$ and $x_5 + x_8 + x_9 \leq 1$, both with a violation of $\frac{1}{2}$. The corresponding nonedges are highlighted in black in Fig. 1b, c. The first inequality dominates the second one, but it has a smaller cut depth:

Cut	Cut violation	$\|\alpha\|_2$	Cut depth
$x_2 + x_5 + x_8 + x_9 \leq 1$	$\frac{1}{2}$	$\sqrt{4}$	$\frac{1}{2\sqrt{4}}$
$x_5 + x_8 + x_9 \leq 1$	$\frac{1}{2}$	$\sqrt{3}$	$\frac{1}{2\sqrt{3}}$

Definition 3 *(Cut depth variant)* Given a relaxation $P \subseteq \mathbb{R}^n_+$ with an optimal solution $x^* \in P$ and a cut $\alpha x \leq \alpha_0$, the *cut depth variant* is the quantity $\dfrac{\alpha x^* - \alpha_0}{\sqrt{\sum_{j=1:x_j^* \neq 0}^{n} \alpha_j^2 + 1}}$.

This measure, proposed in [28], is similar to the cut depth, but the denominator only depends on the components of α corresponding to nonzero components of x^*. Although it copes with the issue of favoring dominated cuts when some components of x^* are zero (as in Example 1), it does not avoid the drawback when some components are nonzero but very small.

Example 2 Consider the LP relaxation of a generic 0–1 IP in which we maximize $\sum_{j=1}^{n} x_j$ over some polyhedron $P \subseteq \mathbb{R}^n_+$. Let $x^* = (1, 1, \frac{1}{100}, \frac{1}{100}, \frac{1}{100}, \frac{1}{100}, \frac{1}{100}, \frac{1}{100}, \frac{1}{2})$ be an optimal solution and let $x_1 + x_2 \leq 1$ and $\sum_{j=1}^{9} x_j \leq 1$ be two valid inequalities. Although the first one has a larger cut depth and cut depth variant (see the following table, where $\|\alpha\|_2^{'} = \sqrt{\sum_{j=1:x_j^* \neq 0}^{n} \alpha_j^2 + 1}$), it is dominated by the second one.

Cut	Violation	$\|\alpha\|_2$	$\|\alpha\|_2^{'}$	Depth	Depth Variant
$x_1 + x_2 \leq 1$	1	$\sqrt{2}$	$\sqrt{2} + 1$	$\frac{1}{\sqrt{2}} = 0.57$	$\frac{1}{\sqrt{2}+1} = 0.41$
$\sum_{j=1}^{9} x_j \leq 1$	1.56	$\sqrt{9}$	$\sqrt{9} + 1$	$\frac{1.56}{\sqrt{9}} = 0.52$	$\frac{1.56}{\sqrt{9}+1} = 0.39$

As pointed out in [5], cut depth may be unreliable for nonfull-dimensional polyhedra. Indeed, for a polyhedron $P \subset \mathbb{R}^n$ of dimension $d < n$, a cut that is uniquely defined in $\mathbb{R}^d$ can be represented in $\mathbb{R}^n$ with $n - d$ degrees of freedom, with different Euclidean distances from x^*. To cope with this issue, the hyperplane corresponding to the cut can be rotated to make it orthogonal to the affine hull of P. For more detail, the reader is referred to [14] for two alternative measures, namely, "rotated steepness" and "steepness with bounds".

2.1.2 Angle measures

Definition 4 (*Objective function parallelism*) Given a valid inequality $\alpha x \leq \alpha_0$ and an objective function cx, the *objective function parallelism* is defined as the cosine of the angle between α and c, namely $\dfrac{\alpha c}{\|\alpha\|_2 \; \|c\|_2}$.

Given a maximization problem and a cut $\alpha x \leq \alpha_0$, if the objective function parallelism takes its maximum value of 1, we have $\alpha = \lambda c$, for some $\lambda > 0$, and then $cx \leq \frac{\alpha_0}{\lambda}$. Similarly, for a minimization problem and a cut $\alpha x \geq \alpha_0$, we have $cx \geq \frac{\alpha_0}{\lambda}$. Therefore, cuts that are parallel to c directly imply a bound on the objective function. An important drawback is that, by favoring cuts whose normal vector α is almost parallel to c, we tend to favor cuts that are parallel to one another.

Definition 5 *(Cut parallelism)* Given two valid inequalities $\alpha x \leq \alpha_0$ and $\alpha' x \leq \alpha_0'$, the *cut parallelism* is defined as the cosine of the angle between α and α', namely $\dfrac{\alpha\alpha'}{\|\alpha\|_2 \, \|\alpha'\|_2}$.

Fig. 2 Two valid inequalities with the same cut depth, but with (**a**) a large cut parallelism and (**b**) a small cut parallelism. The polyhedra corresponding to the relaxations are highlighted in gray. In (**b**), the polyhedron of the relaxation has a smaller volume than that in (**a**)

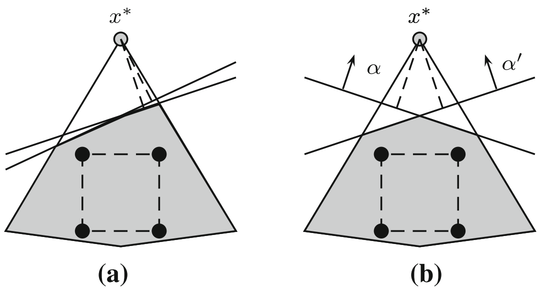

Figure 2 illustrates the simple geometrical intuition for favoring cuts with a small cut parallelism, that is, with large angles between their normal vectors. If we add to the feasible region of the current relaxation two cuts $\alpha x \leq \alpha_0$ and $\alpha' x \leq \alpha'_0$ with the same cut depth, the larger the angle between α and α', the tighter the relaxation is likely to be.

Cut parallelism is adopted in most state-of-the-art cut selection procedures. As remarked in [5], discarding cuts that are close to parallel to previously added ones allows to discard duplicates. See [28] for, among others, a computational study on how often a cut is discarded because it is almost parallel to or dominated by a previously generated one. When experimenting with lift-and-project cuts, the authors of [5] observe that a larger fraction of the duality gap is closed when generating "cuts that improve the polyhedron in diverse directions". Many papers confirm the effectiveness of this measure also for other types of cuts. See, e.g., [1,3,6,28].

2.1.3 Cut sparsity

The *density* of a cut is the number of nonzero components of its normal vector α. Cut sparsity is important for two main reasons. On the one hand, the introduction of dense cuts is sometimes discouraged in cutting plane methods, see, e.g., [8], because they lead to denser linear programs that are harder to solve and are possibly affected by larger numerical errors. On the other hand, the density of a cut is often related to its strength. Indeed, for a given cut violation and a given right-hand side α_0, assuming that $x \geq 0$ and $\alpha \geq 0$, undominated cuts of the form $\alpha x \leq \alpha_0$ are obtained by looking for dense cuts, whereas undominated cuts of the form $\alpha x \geq \alpha_0$ are obtained by looking for sparse ones.

2.2 Cut selection procedures

As previously mentioned, in a typical cut selection procedure the candidate cutting planes are sorted w.r.t. a distance measure and then considered according to that sorting order and added only if the chosen cut quality measures meet some given requirements.

In [4] and [3], sorting is based on, respectively, cut violation and cut depth, and cuts are added only if their parallelism is below 0.999. In [1] and in the solver SCIP, the cuts are sorted w.r.t. an aggregate of cut depth, cut parallelism, and objective function parallelism and then added to the relaxation only if their parallelism is at most 0.5. Although an aggregate measure without the objective function parallelism leads to better average results, all three terms are considered in [1], possibly because of the better performance on some structured problems.

According to the experiments in [14], where different distance measures are compared, the methods based on cut depth perform substantially better than those based on cut violation. Similar experiments in [28] suggest to sort the cuts w.r.t. an aggregate measure based on variants of cut depth and cut density, and on objective function parallelism, and to introduce them only if their cut parallelism is at most 0.1.

3 Coordinated cutting plane generation

In this paper, we propose a multi-objective cutting plane generation scheme in which two cut quality measures are lexicographically optimized in the separation problem. We consider valid inequalities $\alpha x \leq \alpha_0$ or $\alpha x \geq \alpha_0$, where $\alpha \in \{0, 1\}^n$ and α_0 is a constant.

3.1 Choice of two suitable cut quality measures

Based on our observations in the previous section, we choose a measure of distance between the cut that we are about to generate and the optimal solution of the current relaxation as well as a measure of diversity w.r.t. the cutting planes that were previously introduced. Objective function parallelism is not considered because it tends to favor the generation of cuts that are parallel to one another, disfavoring their diversity. We address cut density and sparsity, which are directly related to cut domination for cuts with binary left-hand side coefficients and a constant right-hand side, in Sect. 4, where we propose a revised version of our coordinated cutting plane generation scheme which produces cuts that are guaranteed to be undominated.

As distance measure, we choose cut violation rather than cut depth. Besides its linearity, which makes the separation problem easier to solve, this measure does not suffer from the cut domination issue mentioned in Sect. 2.

As diversity measure between two cutting planes $\alpha x \leq \alpha_0$ and $\alpha' x \leq \alpha'_0$, we adopt the 1-norm distance between their normal vectors. Note that, when α and α' are binary vectors, $\|\alpha - \alpha'\|_1 = \sum_{j=1}^{n} |\alpha_j - \alpha'_j|$ amounts to the Hamming distance between α and α'. The following example illustrates the relevance of this choice.

Example 3 Consider the fractional clique polytope for the Petersen graph shown in Fig. 3. The unique optimal solution of maximizing $\sum_{j=1}^{n} x_j$ over the polytope is $x^* = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$, of value 5. If we add the two stable set inequalities $x_5+x_8+x_9 \leq 1$ and $x_2+x_8+x_9 \leq 1$ (in gray in Fig. 3a) with 1-norm distance equal to 1, the new optimal solution of the relaxation is $x^* = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2})$, of value 4.5. If we add the two stable set inequalities $x_5 + x_8 + x_9 \leq 1$ and $x_4 + x_6 + x_7 \leq 1$ (in gray and light gray in Fig. 3b), with 1-norm distance equal to 3, we get a better bound since the new optimal solution of the relaxation is $x^* = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2})$, of value 4.

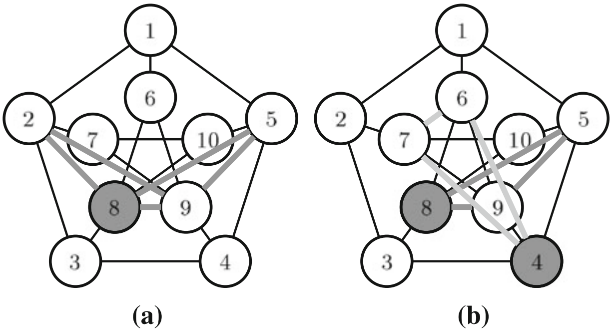

Fig. 3 Illustration of Example 3: the nodes in gray have value 0, whereas those in white have value $\frac{1}{2}$

Assume that $\alpha x \leq \alpha_0$ is the cut to be generated and $\alpha' \leq \alpha'_0$ is a previously generated one. A straightforward property of the 1-norm distance between binary vectors is that $\left\|\alpha - \alpha'\right\|_1$ is a linear function of α when α' is constant. Indeed, such quantity can be represented as a sum of simple disjunctions taking value $1 - \alpha'_j$ if $\alpha_j = 1$ and α'_j if $\alpha_j = 0$. Therefore, we have

$$\begin{aligned}\sum_{j=1}^{n} \mid \alpha_j - \alpha'_j \mid &= \sum_{j=1}^{n} \alpha_j(1 - \alpha'_j) + \sum_{j=1}^{n}(1 - \alpha_j)\alpha'_j \\ &= \sum_{j=1}^{n} \alpha_j - 2\sum_{j=1}^{n} \alpha_j\alpha'_j + \sum_{j=1}^{n} \alpha'_j = (e - 2\alpha')\alpha + e\alpha', \qquad (1)\end{aligned}$$

where e denotes the all-one vector.

Assume that k cuts have been added to the relaxation. As diversity measure between a cutting plane $\alpha x \leq \alpha_0$ that we are about to generate and the whole set of the previously introduced cuts, we consider the 1-norm distance between the normal vector α of that cut and a weighted combination $\bar{\alpha}^k$ of the normal vectors of the previously generated ones. In this work, we define $\bar{\alpha}^k$ as the arithmetic mean of those normal vectors, namely $\bar{\alpha}^k := \frac{1}{k}\sum_{l=1}^{k} \alpha^l$. Note that Eq. (1) holds for any $\alpha' \in [0, 1]$ and, therefore, also for $\bar{\alpha}^k$. Other versions, including that where the diversity measure only considers pairs of successive cuts, i.e., where $\bar{\alpha}^k := \alpha^k$, turned out to be less effective.

3.2 Multi-objective separation problem

In many problems, such as max clique (see also Examples 1 and 3) and min Steiner tree, the standard separation problem aiming at maximizing the cut violation admits multiple optimal solutions. We exploit this property and, among all the maximally violated cuts of a given family, we look for a cut that is also maximally diverse w.r.t. the previously generated cuts with coefficients $(\alpha^1, \alpha_0^1), \ldots, (\alpha^k, \alpha_0^k)$. For cuts of the form $\alpha x \leq \alpha_0$, the new separation problem can be stated as the following lexicographic bi-objective optimization problem:

$$\begin{array}{ll} \max & \left\|\alpha-\bar{\alpha}^{k}\right\|_{1} \\ \text{s.t.} & \alpha=\operatorname{argmax}\{\alpha x^{*}-\alpha_{0}\} \\ & (\alpha, \alpha_{0}) \in \mathcal{C}, \end{array}$$

which is equivalent to the following single objective problem:

$$\begin{array}{ll} \max & \alpha x^{*}-\alpha_{0}+\epsilon\left\|\alpha-\bar{\alpha}^{k}\right\|_{1} \\ \text{s.t.} & (\alpha, \alpha_{0}) \in \mathcal{C} \end{array}$$

for a finite, small enough, $\epsilon > 0$ (see below).

According to (1) with $\bar{\alpha}^k$ replacing α', the objective function becomes $\alpha x^* - \alpha_0 + \epsilon(e - 2\bar{\alpha}^k)\alpha + \epsilon e\bar{\alpha}^k$. By collecting α and dropping the constant terms, we have the *Coordinated Separation problem*:

$$\begin{array}{ll} \max & (x^{*}+\epsilon(e-2\bar{\alpha}^{k}))\alpha \\ \text{s.t.} & (\alpha, \alpha_{0}) \in \mathcal{C}. \end{array} \tag{2}$$

The effect of the 1-norm diversity is of adding, to each coefficient x_j^* of α_j, a term which is strictly positive if $\bar{\alpha}_j^k < \frac{1}{2}$ and strictly negative if $\bar{\alpha}_j^k > \frac{1}{2}$. Thus, the generation of a cut with $\alpha_j = 1$ is favored for components that have value 1 in less than half of the previous k cuts, and disfavored otherwise.

For cuts of the form $\alpha x \geq \alpha_0$, for which the cut violation is $\alpha_0 - \alpha x^*$, we maximize $\alpha_0 - \alpha x^* + \epsilon \left\|\alpha - \bar{\alpha}^k\right\|_1 = \alpha_0 - \alpha x^* + \epsilon(e - 2\bar{\alpha}^k)\alpha + \epsilon e\bar{\alpha}^k$. In this case, the coordinated separation problem, stated as a minimization problem, becomes:

$$\begin{array}{ll} \min & (x^{*}-\epsilon(e-2\bar{\alpha}^{k}))\alpha \\ \text{s.t.} & (\alpha, \alpha_{0}) \in \mathcal{C}, \end{array} \tag{3}$$

where the effect of the 1-norm diversity is the same as in the previous case.

Let us now comment on the choice of an appropriate value for ϵ. Given two functions $f_1, f_2 : \mathbb{R}^n \to \mathbb{R}$, the value of ϵ must be such that, when optimizing $f_1 + \epsilon f_2$, we obtain a solution which is optimal for f_1 and which, among all such solutions, is also optimal for f_2. Let Δ_2 be the difference between the maximum and minimum of f_2 and let δ_1 be the smallest variation between any two values that f_1 can take. A simple sufficient condition is $\epsilon\Delta_2 < \delta_1$, see for instance [27]. In our case, Δ_2 is finite and amounts to $2n$. Note that $f_1 = \alpha x^*$ because α_0 is a constant and can be dropped. Since α takes discrete values, the image of f_1 is a finite set and a finite δ_1 exists. Although finding the exact value of δ_1 may be difficult, any lower bound yields a valid value for ϵ. For rational polyhedra, $x^* \in \mathbb{Q}^n$ and δ_1 can be bounded from below by the reciprocal of any multiple of all the denominators of the components of x^*. See the Appendix for more details on how we choose the value of ϵ and on how we proceed to avoid numerical errors.

In the case of cuts with 0–1 coefficients, our coordinated cutting plane generation scheme exploits the linearity of the 1-norm diversity function and the existence of a finite ϵ. The approach can be extended to cuts with general integer coefficients at the cost of introducing extra variables (both binary and continuous) accounting for a reformulation of the absolute value (see [29] for the standard one). An appropriate value for ϵ requires both Δ_2 and δ_1 to be finite. For cuts with continuous or general integer coefficents, a value for Δ_2 is usually easy to find if we assume that the coefficients are bounded, because this implies the boundedness of the range of the 1-norm distance. For cuts with general integer coefficients, the value for δ_1 is clearly finite as it is equivalent to that for the 0–1 case. Unfortunately, for continuous coefficients this value is not bounded and hence we cannot cast our lexicographic separation problem as in (2) or (3). A possible extension of our approach to this case amounts to solving the coordinated separation problem in two steps, first maximizing the cut violation and then maximizing the cut diversity subject to a constraint on the cut violation.

4 Revised separation for undominated cuts

In this section, we show how our coordinated separation problem can be modified to guarantee the generation of undominated cuts.

We shall say that a cut $\alpha x \leq \alpha_0$ is *maximal* ($\alpha x \geq \alpha_0$ is *minimal*) if it becomes invalid when any component of α is increased from 0 to 1 (decreased from 1 to 0). Since $\alpha \in \{0, 1\}^n$ and α_0 is a constant, the maximality (minimality) of α is necessary and sufficient to have a cut that is not dominated by other cuts of the same family.

4.1 Revised standard separation problem

When solving the standard separation problem, if $x^* > 0$ any maximally violated cut of the form $\alpha x \leq \alpha_0$ is maximal. This is because if, given a cut, another valid inequality can be obtained by setting any $\alpha_j = 0$ to 1, then the cut violation $\alpha x^* - \alpha_0$ can be strictly increased. Similarly, when $x^* > 0$, any maximally violated cut of the form $\alpha x \geq \alpha_0$ is minimal. In the general case where x^* might contain at least a component $x^*_j = 0$, the cut violation will be unmodified by setting either $\alpha_j = 0$ or $\alpha_j = 1$, thus allowing for the generation of dominated cuts.

A way to generate undominated cuts for any (unrestricted) x^* is the following one. Maximal cuts of the form $\alpha x \leq \alpha_0$ can be obtained by looking for maximally violated cuts with a maximum number of nonzero components of α. This can be achieved by modifying the standard separation problem as follows:

$$\begin{array}{ll} \max & \alpha x^* - \alpha_0 + \epsilon \, \|\alpha\|_1 \\ \text{s.t.} & (\alpha, \alpha_0) \in \mathcal{C}, \end{array}$$

for an appropriate $\epsilon > 0$, usually larger than that used in the coordinated separation problem. By collecting α, dropping the constant term, and rewriting $\|\alpha\|_1$ as $e\alpha$, we have the *Revised Standard Separation problem*:

$$\begin{array}{ll} \max & (x^* + \epsilon e)\alpha \\ \text{s.t.} & (\alpha, \alpha_0) \in \mathcal{C}, \end{array} \tag{4}$$

where the only difference w.r.t. the standard one is that x^* is substituted with $x^* + \epsilon e$. For cuts of the form $\alpha x \geq \alpha_0$, we maximize $\alpha_0 - \alpha x^* - \epsilon \|\alpha\|_1$, thus discarding, among all the maximally violated cuts, those that are not minimal. This amounts to minimizing $(x^* + \epsilon e)\alpha$.

A similar technique is used in [23] when separating cut set inequalities for the min Steiner tree problem. In [23] however, ϵ is taken as a fixed value, with no guarantee of respecting the lexicographic priority between $\alpha x^* - \alpha_0$ and $\|\alpha\|_1$. Thus, the separation problem may not yield a violated cut even if such a cut exists.

4.2 Revised coordinated cut separation problem

In the coordinated separation problem, it may happen that, among all the cuts which are maximally violated and also maximally diverse w.r.t. the previous cuts, none is undominated.

Example 4 Consider a generic 0–1 IP where the cut $x_1 + x_2 \leq 1$ has been added to the relaxation and $x^* = (0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ is the current optimal solution. Consider the two valid inequalities $x_2 + x_3 + x_4 \leq 1$ and $x_1 + x_2 + x_3 + x_4 \leq 1$, which have both a violation of $\frac{1}{2}$ but a diversity w.r.t. the previous cut of, respectively, 3 and 2. The coordinated separation problem yields the first inequality, which is dominated by the second one.

To enforce the generation of undominated (maximal) cuts of the form $\alpha x \leq \alpha_0$, we include the extra term $2\epsilon \|\alpha\|_1$ in the coordinated separation problem. Note that undominated cuts are obtained when adding $\gamma\epsilon \|\alpha\|_1$ for any $\gamma > 1$. Here, we take $\gamma = 2$, which is the smallest integer value satisfying the condition. We obtain:

$$\begin{array}{ll} \max & \alpha x^* - \alpha_0 + \epsilon \left\|\alpha - \bar{\alpha}^k\right\|_1 + 2\epsilon \|\alpha\|_1 \\ \text{s.t.} & (\alpha, \alpha_0) \in \mathcal{C}. \end{array}$$

The objective function can be rewritten as $\alpha x^* - \alpha_0 + \epsilon(3e - 2\bar{\alpha}^k)\alpha + \epsilon e \bar{\alpha}^k$. After collecting α and dropping the constant terms, we obtain the *Revised (max) Coordinated Separation problem*:

$$\begin{array}{ll} \max & (x^* + \epsilon(3e - 2\bar{\alpha}^k))\alpha \\ \text{s.t.} & (\alpha, \alpha_0) \in \mathcal{C}. \end{array} \tag{5}$$

Since $x^* \geq 0$ and $0 \leq \bar{\alpha}^k \leq 1$, all the components of $\epsilon(3 - 2\bar{\alpha}^k_j)$ take values in $[\epsilon, 3\epsilon]$ and all the objective function coefficients $\hat{x}_j := x^*_j + \epsilon(3 - 2\bar{\alpha}^k_j)$ are strictly positive. Thus (5) is guaranteed to yield undominated (maximal) cuts.

Similarly, for cuts of the form $\alpha x \geq \alpha_0$, we subtract the term $2\epsilon \|\alpha\|_1$. The objective function of the minimization problem becomes $\alpha x^* - \alpha_0 - \epsilon(-e - 2\bar{\alpha}^k)\alpha - \epsilon e \bar{\alpha}^k$, and the corresponding *Revised (min) Coordinated Separation problem* is:

$$\begin{array}{ll} \min & (x^* + \epsilon(e + 2\bar{\alpha}^k))\alpha \\ \text{s.t.} & (\alpha, \alpha_0) \in \mathcal{C}, \end{array} \tag{6}$$

where all the coefficients $\hat{x}_j := x^*_j + \epsilon(1 + 2\bar{\alpha}^k_j)$ are strictly positive.

It is worth pointing out that our revised coordinated separation problem amounts to the standard one when adopting the objective function coefficient vector $\hat{x}$ instead of x^*. Since $\hat{x}$ is nonnegative if x^* is nonnegative, any algorithm for standard separation which is applicable to objective functions with nonnegative coefficients can also be used to solve the revised coordinated separation problem. This is the case of stable set and cut set inequalities, which we consider in Sect. 5.

The substitution of x^* with $\hat{x}$ in the separation problem suggests some similarities with stabilization techniques used in column generation where dual multipliers are appropriately modified in the pricing subproblem, see, e.g., [15], and with the in-out search strategy for cutting plane methods [9,18].

Finally, let us emphasize that, unlike in previous approaches where cut diversity and cut strength are only implicitly favored with heuristic techniques, as in [17], in our revised coordinated separation scheme we precisely optimize a cut diversity measure over all the maximally violated cuts, with the guarantee of generating an inequality which is always undominated.

5 Computational experiments

We assess the impact of our coordinated cutting plane scheme when separating stable set and cut set inequalities for, respectively, the max clique and min Steiner tree problems. For max clique, given a graph $G = (V, E)$, we consider the following LP relaxation:

$$\min_{x_i \geq 0, i \in V} \left\{ \sum_{i \in V} x_i : \sum_{i \in S} x_i \leq 1 \quad \text{for } S \in \mathcal{S} \right\}, \tag{7}$$

where $\mathcal{S}$ is the collection of all the stable sets of G. When only maximal stable sets are considered, the corresponding inequalities are facet defining. Solving (7) is equivalent to computing the so-called fractional coloring number of G. The separation problem amounts to finding a maximum weight stable set of G with weights x_i^*.

For the min Steiner tree problem, given a graph $G = (V, E)$ with a subset $T \subset V$ of terminals and a cost function $c : E \to \mathbb{R}^+$, we adopt the directed formulation [10,11], which is tighter than the undirected one. Let $G' = (V, A)$ be the directed version of G containing a pair of arcs (i, j) and (j, i) for each edge $\{i, j\} \in E$, with the same cost as $\{i, j\}$. Let $r \in T$ be an arbitrary root node. We have the following LP relaxation:

$$\min_{0 \leq x_{ij} \leq 1, (i,j) \in A} \left\{ \sum_{(i,j) \in A} c_{ij} x_{ij} : \begin{array}{l} \displaystyle\sum_{(i,j) \in \delta^+(S)} x_{ij} \geq 1 \quad \text{for } S \subset V \\ \text{s.t. } r \in S \text{ and } V \backslash S \cap T \neq \emptyset \end{array} \right\},$$

which is based on the so-called $s - t$ (for $s = r$) cut set inequalities. The separation problem amounts to finding an $s - t$ cut set in G' of minimum total cost, for $s = r$ and for each terminal $t \in T \backslash \{r\}$, where the values x_{ij}^* are used as arc capacities. The problem is polynomially solvable for nonnegative arc capacities (as a max flow problem), which is the case of both the revised standard and the revised coordinated separation problems.

5.1 Results

The main focus of this paper is on a pure cutting plane setting, but we also consider a cut-and-branch one. In the pure cutting plane setting, we compare the results obtained with the revised coordinated separation and the revised standard separation for max clique and min Steiner tree. In the cut-and-branch setting, we compare for max clique the overall performance of the cut-and-branch method when cuts are generated (only at the root node) either according to the revised standard or the revised coordinated separation schemes. In both settings, the cutting plane algorithms are stopped when no more violated inequalities are found, thus closing the same fraction of the duality gap.

The algorithms are implemented in C++ using the ILOG CPLEX Concert Library, compiled with GNU-g++-4.3. The graphs are represented via the `adjacency_list` structure available in the Boost Graph Library. The experiments are carried out on a Dell PowerEdge Quad Core Xeon 2.0 Ghz, with 4 GB of RAM. All the LP relaxations are solved with CPLEX 12.2 using the dual simplex method. For the sake of reproducibility and comparability, we disable preprocessing by adopting the parameter settings `PreInd=0, AdvInd=0, and Reduce=0`.

5.1.1 Pure cutting plane setting

In the tables reporting the results for max clique and min Steiner tree, we consider the following figures:

- Time: total computing time (in seconds) spent for solving the relaxation and separation problems.
- Rnds: total number of cutting plane rounds.
- Cuts: total number of generated cuts.
- Dupl: number of duplicated cuts which are discarded (only for min Steiner tree, see below).
- Cond: arithmetic mean of the condition number of the optimal basis matrices of the last 20 linear programming relaxations. We use the CPLEX 12.2 function `getQuality(IloCplex::ExactKappa)`. The average mitigates the natural oscillations of the number.
- ReIt: arithmetic mean of the number of dual simplex iterations carried out to reoptimize the LP relaxations.
- ReT: arithmetic mean of the computing time (in seconds) needed to reoptimize the LP relaxations.
- SepTime: arithmetic mean of the computing time (in seconds) taken to solve the separation problem.

The figures corresponding to the best cutting plane generation scheme are highlighted in bold. For comparison purposes, in the last line of each table we also report the percentage aggregate saving for each figure when adopting our coordinated cutting plane generation scheme, evaluated over all the instances. Specifically, for each figure and for each instance we compute the ratio between the values obtained with revised coordinated separation and revised standard separation, and return the geometric mean of those ratios (using a shifted geometric mean with a shift of 0.01 for the computing time). The aggregate saving for each figure amounts to one minus the geometric mean.

For max clique, we consider a subset of the instances from the second DIMACS implementation challenge on max clique, graph coloring, and satisfiability [22]. The initial relaxation only contains the bounds on the variables. We formulate the revised standard and revised coordinated separation problems adopting a simple 0–1 IP with a constraint for each nonedge of the graph and solve them with CPLEX. To handle as precisely as possible the small differences among the objective function coefficients which are due to the parameter ϵ, we set `NumericalEmphasis=1`, `EpAGap=0`, `EpGap=0`, `EpInt=0`, and `EpOpt=1e-09`, to have the tightest precision on absolute and relative duality gap, integrality gap, and reduced cost tolerance.

The results for max clique are reported in Table 1. On average, our coordinated cutting plane generation scheme allows to save 37 % of the computing time, to generate 23 % less cuts, and to obtain relaxations with an average condition number that is reduced by 47 % w.r.t. the revised standard scheme. Note that the average computing time needed to solve our coordinated separation problems not only does not exceed that for the revised standard one, but is also slightly smaller, by 3 % on average. Since with coordinated cutting plane generation we introduce 23 % less cuts, which are only 1 % denser (we do not report the density figure due to lack of space), the LP relaxations are substantially smaller than those obtained with the revised standard separation. This is likely to determine the substantial reduction in the average number of iterations and computing time needed for the reoptimizations by, respectively, 30 and 24 %.

Note that our cutting plane method requires a higher computing time only for 5 instances out of 24. According to Table 1, the improvement with coordinated cutting plane generation can be as large as on instance `c-fat500-10`. In that case, the algorithm terminates in less than 20 s (as opposed to 70.53 s), generates 128 cuts (instead of 413), and yields a final relaxation with a condition number that is smaller by more than 3 orders of magnitude.

For min Steiner tree, we consider five data sets taken from the SteinLib [24], namely B, C, D, I640, and PUC. Cut coordination is achieved by considering the cut diversity w.r.t. all the cuts that were previously generated, also within the same round, independently of the terminal. Other options where cut diversity only considers cuts generated when separating w.r.t. the same terminal yielded not as good results. We solve the two separation problems with the Boost Graph Library implementation of the $O(|V|\,|A|^2)$ Edmonds–Karp algorithm [16]. The root node r is chosen as the terminal with the largest degree. We observe that this choice allows to close a larger fraction of the duality gap, regardless of the cutting plane algorithm that is adopted. For each instance, we derive an initial pool of inequalities by solving, for each pair of source $s = r$ and terminal t, a min $s - t$ cut set problem with unit capacity on every arc that is still uncovered. Cuts are generated in rounds by solving a separation problem for each terminal (except for the root node). Since a cut set can be found more than once

Table 1 Comparison between revised standard separation and revised coordinated separation on max clique instances in a pure cutting plane setting

	Revised standard separation						Revised coordinated separation					
	Time	Cuts	Cond	ReIt	ReT	SepT	Time	Cuts	Cond	ReIt	ReT	SepT
c-fat200-1	5.20	192	2.0E+3	279.89	0.02	**0.01**	**3.07**	**158**	**1.9E+3**	**261.22**	**0.01**	**0.01**
c-fat200-2	1.04	74	1.9E+2	81.62	**0.00**	**0.01**	**0.52**	**44**	**2.5E+1**	**27.39**	**0.00**	**0.01**
c-fat200-5	8.64	254	1.0E+3	172.19	**0.00**	**0.02**	**6.14**	**201**	**5.3E+2**	**139.56**	**0.00**	**0.02**
c-fat500-10	70.53	413	1.2E+3	364.38	0.01	0.12	**19.75**	**128**	**1.3E+0**	**64.00**	**0.00**	**0.11**
c-fat500-1	54.59	345	8.2E+3	804.99	0.13	**0.01**	**7.39**	**194**	**4.2E+3**	**261.88**	**0.02**	**0.01**
c-fat500-2	200.11	512	1.7E+4	1,129.18	0.34	**0.02**	**1.60**	**51**	**3.6E+1**	**30.72**	**0.00**	**0.02**
c-fat500-5	22.21	244	**1.4E+3**	329.56	**0.01**	**0.05**	**18.18**	**217**	2.1E+3	**250.24**	**0.01**	**0.05**
hamming6-2	1.84	95	5.1E+1	41.86	**0.00**	**0.02**	**1.06**	**59**	**3.2E+1**	**29.50**	**0.00**	**0.02**
hamming6-4	7.64	60	5.8E+2	64.02	**0.00**	0.12	**5.99**	**58**	**4.0E+2**	**62.93**	**0.00**	**0.10**
hamming8-2	254.73	545	**1.0E+2**	189.03	**0.00**	0.43	**128.72**	**287**	2.3E+2	**142.90**	**0.00**	**0.42**
johnson16-2-4	13.98	18	**3.6E+0**	28.00	**0.00**	0.67	**13.01**	**17**	4.3E+0	**27.06**	**0.00**	**0.60**
johnson8-2-4	**0.04**	**8**	**3.6E+0**	**7.14**	**0.00**	**0.00**	**0.04**	**8**	**3.6E+0**	**7.14**	**0.00**	**0.00**
johnson8-4-4	6.24	**55**	1.6E+2	51.54	**0.00**	0.11	**4.76**	**55**	**1.1E+2**	**47.94**	**0.00**	**0.08**
MANN_a9	**1.44**	**48**	**4.8E+1**	27.21	**0.00**	**0.02**	**1.44**	**48**	7.3E+1	**25.79**	**0.00**	**0.02**
myciel3	**0.06**	**13**	**1.1E+1**	7.17	**0.00**	**0.00**	**0.06**	**13**	**1.1E+1**	**6.58**	**0.00**	**0.00**
myciel4	**0.79**	27	**6.6E+1**	**18.85**	**0.00**	**0.03**	1.13	32	1.0E+2	22.55	**0.00**	**0.03**
myciel5	**12.85**	**72**	5.3E+2	49.97	**0.00**	**0.17**	12.96	**72**	**4.5E+2**	**49.17**	**0.00**	**0.17**

Table 1 continued

	Revised standard separation						Revised coordinated separation					
	Time	Cuts	Cond	ReIt	ReT	SepT	Time	Cuts	Cond	ReIt	ReT	SepT
queen10_10	**46.09**	**318**	4.4E+3	**222.13**	**0.01**	**0.13**	55.64	327	**4.2E+3**	229.01	**0.01**	0.15
queen11_11	**53.19**	**333**	**3.4E+3**	**272.72**	**0.02**	**0.14**	55.65	337	**3.4E+3**	289.35	**0.02**	**0.14**
queen12_12	**73.62**	**400**	**3.7E+3**	**346.63**	**0.03**	**0.15**	75.86	413	4.6E+3	365.57	**0.03**	**0.15**
queen13_13	99.91	447	5.4E+3	**422.71**	**0.05**	**0.17**	**98.58**	**444**	**4.6E+3**	438.66	**0.05**	**0.17**
queen14_14	177.40	544	6.2E+3	**549.37**	**0.08**	**0.24**	**172.73**	**524**	**5.4E+3**	553.32	**0.08**	**0.24**
queen15_15	276.65	603	**6.3E+3**	658.98	0.12	0.33	**257.42**	**595**	8.2E+3	**657.45**	**0.11**	**0.30**
queen16_16	409.47	666	**8.6E+3**	**781.49**	**0.17**	0.42	**369.35**	**656**	8.9E+3	795.16	**0.17**	**0.38**
Aggr. saving %							37	23	47	30	24	3

during a cutting plane iteration as the solution to separation problems for different terminals, we only add nonduplicate cuts.

Table 2 reports the results obtained for min Steiner tree. Coordinated cutting plane generation yields, on average, a substantial reduction in the number of rounds and cuts, by 16 % in both cases, in the number of duplicate cuts, by 33 %, and in the condition number, by 20 %. When focusing on the nonduplicated cuts which are actually added to the LP relaxation, their number is reduced by 8 % (rather than 16 %), but this figure is not reported in the tables for a matter of space. Moreover, those cuts are also 1 % denser, on average, than those obtained with the revised standard separation problem. Therefore, although the LP relaxations that we solve with coordinated cut generation are smaller than those obtained with the revised standard separation, their size is not reduced as much as for max clique. This is likely to justify the reduction of only 2 % of the average time spent to reoptimize the LP relaxations. Note also that the average separation time is increased, on average, by 14 %. This could be because the vector of capacities $\hat{x}$ is less sparse and more diversified yielding max flow problems which are more difficult to solve. Nevertheless, since the number of rounds is largely reduced the total computing time is only increased by 7 % on average.

According to Table 2, coordinated cutting plane generation can be as effective as on instance `b15`, where the number of rounds is only 6 instead of 12 and the number of cuts is 243 instead of 452, with only 97 duplicates instead of 296. Note that the number of rounds, cuts, and duplicates only increases for, respectively, 7, 8, and 8 instances out of the 64, while all those figures are simultaneously improved on the other 51 instances.

Finally, for both problems we have also experimented with a variant of our approach where the cut diversity is enforced w.r.t. the average of only the previously generated cuts that are binding at the solution of the current LP relaxation. Since only n inequalities suffice to describe the vertex of the relaxation corresponding to its solution, this variant may seem interesting. Overall, though, we obtain substantially inferior improvements compared to the case where all the cuts are considered. This is likely due to the fact that all the inequalities that we generate are facet defining. Indeed, this guarantees that no previously introduced cuts can be dominated by a combination of the new ones. Since a cut which is nonbinding at a current iteration may be binding in a future one, we should favor the generation of a cut which is diverse not only from the cuts which are currently binding at the LP optimal solution, but also w.r.t. those that might (or will) be binding in the future iterations. This is precisely what we obtain by enforcing the 1-norm diversity w.r.t. the average of all the previous cuts when also including those that are currently nonbinding.

5.1.2 Cut-and-branch setting

To evaluate the potential of our coordinated cutting plane generation scheme when solving an optimization problem to optimality, we also experiment with a cut-and-branch algorithm for max clique. At the root node, we run a cutting plane algorithm using either the revised standard or revised coordinated separation problem until the relaxation is solved to optimality. Then, we solve the unrelaxed problem to optimality with branch-and-bound, using CPLEX. Our initial LP relaxation contains all the

Table 2 Comparison between revised standard separation and revised coordinated separation on Steiner tree instances in a pure cutting plane setting (first part)

	Revised standard separation								Revised coordinated separation							
	Time	Rnds	Cuts	Dupl	Cond	ReIt	ReT	SepT	Time	Rnds	Cuts	Dupl	Cond	ReIt	ReT	SepT
b01	0.02	18	42	10	3.3E+1	28.37	**0.00**	**0.00**	**0.01**	**15**	**29**	**1**	**2.3E+1**	**25.44**	**0.00**	**0.00**
b02	0.02	21	143	79	5.9E+1	37.41	**0.00**	**0.00**	**0.01**	**11**	**76**	**31**	**2.5E+1**	**32.58**	**0.00**	**0.00**
b03	0.02	7	138	78	3.5E+1	45.75	**0.00**	**0.00**	**0.01**	**4**	**85**	**38**	**1.8E+1**	**37.80**	**0.00**	**0.00**
b04	**0.02**	18	78	13	7.9E+1	33.37	**0.00**	**0.00**	**0.02**	**13**	**59**	**7**	**4.6E+1**	**28.79**	**0.00**	**0.00**
b05	**0.03**	16	114	51	4.3E+1	39.23	**0.00**	**0.00**	**0.03**	**13**	**109**	**46**	**3.5E+1**	**37.64**	**0.00**	**0.00**
b06	**0.04**	11	179	63	6.3E+1	67.25	**0.00**	**0.00**	**0.04**	**9**	**158**	**45**	**6.0E+1**	**63.20**	**0.00**	**0.00**
b07	0.04	23	239	177	6.0E+1	42.12	**0.00**	**0.00**	**0.03**	**14**	**143**	**87**	**3.7E+1**	**38.07**	**0.00**	**0.00**
b08	**0.02**	14	117	50	3.2E+1	39.53	**0.00**	**0.00**	**0.02**	**9**	**96**	**34**	**3.1E+1**	**38.40**	**0.00**	**0.00**
b09	0.07	22	571	469	5.4E+1	63.39	**0.00**	**0.00**	**0.04**	**9**	**215**	**141**	**3.2E+1**	**62.20**	**0.00**	**0.00**
b10	**0.05**	17	130	33	8.0E+1	51.44	**0.00**	**0.00**	**0.05**	**13**	**104**	**14**	**6.5E+1**	**48.86**	**0.00**	**0.00**
b11	0.12	24	204	44	2.5E+2	74.48	**0.00**	**0.00**	**0.10**	**19**	**159**	**17**	**1.3E+2**	**60.90**	**0.00**	**0.00**
b12	**0.15**	17	345	86	2.2E+2	**98.39**	**0.00**	**0.00**	0.18	**16**	**289**	**35**	**2.0E+2**	100.35	**0.00**	**0.00**
b13	0.04	14	149	46	8.9E+1	**59.47**	**0.00**	**0.00**	**0.03**	**11**	**111**	**22**	**6.3E+1**	59.67	**0.00**	**0.00**
b14	0.11	26	391	227	2.1E+2	96.93	**0.00**	**0.00**	**0.09**	**18**	**238**	**90**	**1.3E+2**	**94.05**	**0.00**	**0.00**
b15	0.09	12	452	296	**7.0E+1**	100.46	**0.00**	**0.00**	**0.05**	**6**	**243**	**97**	**7.0E+1**	**96.29**	**0.00**	**0.00**
b16	**0.16**	24	230	29	3.9E+2	**90.36**	**0.00**	**0.00**	0.18	**23**	**211**	**24**	**2.1E+2**	90.58	**0.00**	**0.00**
b17	**0.15**	23	181	38	1.0E+2	87.12	**0.00**	**0.00**	0.17	**22**	**161**	**31**	**8.1E+1**	**86.91**	**0.00**	**0.00**
b18	**0.19**	13	366	110	1.4E+2	119.14	**0.00**	**0.00**	0.23	**12**	**338**	**89**	**9.9E+1**	**116.31**	**0.00**	**0.00**
c01	0.49	57	180	37	**4.9E+2**	**61.52**	**0.00**	**0.00**	**0.48**	**47**	**147**	**22**	5.0E+2	63.15	**0.00**	**0.00**
c02	2.00	85	451	96	2.7E+3	174.52	0.01	**0.00**	**1.62**	**61**	**358**	**60**	**2.3E+3**	**160.97**	**0.00**	**0.00**
c03	3.02	29	1,616	467	1.5E+3	399.40	**0.01**	**0.00**	**2.19**	**18**	**979**	**171**	**5.0E+2**	**355.21**	**0.01**	**0.00**
c04	4.76	34	2,317	1,032	1.3E+3	521.66	**0.01**	**0.00**	**4.65**	**22**	**1,785**	**664**	**7.8E+2**	**476.26**	**0.01**	**0.00**

Table 2 continued

	Revised standard separation								Revised coordinated separation							
	Time	Rnds	Cuts	Dupl	Cond	ReIt	ReT	SepT	Time	Rnds	Cuts	Dupl	Cond	ReIt	ReT	SepT
c05	**6.59**	24	4,761	3,056	2.2E+3	636.80	**0.01**	**0.00**	6.99	**15**	**3,105**	**1,553**	**1.2E+3**	**607.19**	**0.01**	**0.00**
c06	**2.23**	94	298	51	1.5E+3	126.46	**0.00**	**0.00**	2.33	**84**	**284**	**47**	**1.2E+3**	**119.27**	**0.00**	**0.00**
c07	**1.99**	55	365	27	**2.2E+3**	169.52	**0.01**	**0.00**	2.22	**49**	**321**	**13**	2.4E+3	**163.76**	**0.01**	**0.00**
c08	**12.59**	42	2,772	608	8.1E+3	583.79	**0.03**	**0.00**	15.86	**36**	**2,381**	**374**	**6.2E+3**	**582.81**	0.04	**0.00**
c09	**44.01**	62	5,518	1,293	3.4E+4	967.46	**0.12**	**0.00**	61.94	**60**	**5,307**	**854**	**2.3E+4**	**952.69**	0.13	**0.00**
c10	**13.52**	**19**	3,875	799	9.0E+2	**720.05**	**0.02**	**0.00**	30.77	22	**3,641**	**555**	**1.3E+3**	749.65	0.03	**0.00**
c11	**5.05**	**81**	**311**	70	**3.3E+3**	**141.98**	**0.01**	**0.00**	7.14	83	314	**64**	4.1E+3	143.37	**0.01**	0.01
c12	**15.07**	122	520	98	**4.6E+3**	277.13	**0.01**	**0.00**	16.15	**110**	**494**	**79**	8.3E+3	**239.30**	**0.01**	0.01
c13	**54.62**	**51**	2,755	**197**	2.8E+4	932.44	0.14	**0.00**	76.32	54	**2,629**	209	**2.6E+4**	**911.67**	**0.13**	0.01
c14	**26.19**	25	2,459	**97**	1.5E+3	568.88	**0.02**	**0.00**	38.68	**24**	**2,374**	163	**1.4E+3**	**559.80**	**0.02**	0.01
d01	4.59	128	394	47	3.2E+3	176.29	**0.01**	**0.00**	**3.73**	**101**	**317**	**32**	**2.6E+3**	**153.01**	**0.01**	**0.00**
d02	3.62	77	659	189	2.0E+3	194.64	**0.01**	**0.00**	**3.31**	**59**	**502**	**147**	**1.3E+3**	**179.93**	**0.01**	**0.00**
d03	**19.95**	36	4,823	2,086	5.3E+3	794.76	**0.02**	**0.00**	25.66	**29**	**3,666**	**1,334**	**2.7E+3**	**788.70**	**0.02**	**0.00**
d04	**29.53**	35	7,051	3,251	5.7E+3	960.22	0.04	**0.00**	31.54	**24**	**4,832**	**1,913**	**3.4E+3**	**890.92**	**0.03**	**0.00**
d05	**24.65**	23	11,053	8,023	8.2E+2	1,218.50	**0.02**	**0.00**	35.20	**19**	**9,130**	**6,539**	**4.8E+2**	**1,213.85**	**0.02**	**0.00**
d06	**29.00**	241	790	**145**	2.9E+4	322.50	**0.03**	**0.00**	31.36	**228**	**744**	147	**1.6E+4**	**321.96**	**0.03**	0.01
d07	**7.73**	105	539	153	2.8E+3	200.07	**0.01**	**0.00**	8.02	**85**	**460**	**92**	**1.8E+3**	**193.37**	**0.01**	**0.00**
d08	**72.13**	49	6,810	1,477	**3.0E+4**	**1,300.16**	**0.15**	**0.00**	97.87	**45**	**6,126**	**932**	4.7E+4	1,346.80	0.17	0.01
d10	**83.43**	**23**	**9,450**	2,060	3.7E+3	**1,514.92**	**0.09**	**0.00**	150.89	26	9,748	**1,891**	**3.6E+3**	1,565.56	0.10	0.01
d11	**40.95**	**170**	586	**139**	1.8E+4	**265.30**	**0.04**	**0.01**	57.31	172	**584**	146	**1.1E+4**	287.20	**0.04**	0.03
d12	**40.48**	127	662	89	**1.1E+4**	332.24	**0.03**	**0.01**	43.59	**113**	**607**	**84**	1.5E+4	**297.34**	**0.03**	0.02
cc3-4p	**41.71**	166	1,123	79	4.2E+4	**790.85**	**0.21**	**0.00**	44.25	**165**	**1,113**	**57**	**4.0E+4**	812.17	0.23	**0.00**
cc3-4u	75.64	211	1,424	98	**2.6E+4**	1,077.07	0.32	**0.00**	**56.53**	**196**	**1,317**	**97**	2.8E+4	**957.19**	**0.25**	**0.00**

Table 2 continued

	Revised standard separation								Revised coordinated separation							
	Time	Rnds	Cuts	Dupl	Cond	ReIt	ReT	SepT	Time	Rnds	Cuts	Dupl	Cond	ReIt	ReT	SepT
cc3-5p	**2,112.99**	360	2,517	97	**2.9E+5**	**3,316.22**	**5.56**	**0.00**	2,363.29	**358**	**2,516**	**87**	3.0E+5	3,523.93	6.26	0.01
cc6-2p	6.73	79	771	136	2.7E+4	**424.99**	**0.07**	**0.00**	**6.57**	**68**	**675**	**62**	**2.1E+4**	469.12	0.08	**0.00**
cc6-2u	**8.35**	83	809	144	**3.6E+4**	**524.11**	**0.08**	**0.00**	8.81	**78**	**770**	**78**	3.8E+4	596.32	0.09	**0.00**
hc6p	**0.45**	17	366	51	2.7E+3	**248.17**	**0.01**	**0.00**	0.48	**15**	**350**	**32**	**2.5E+3**	256.94	**0.01**	**0.00**
hc6u	**2.40**	39	**951**	**418**	**1.2E+4**	489.02	**0.04**	**0.00**	2.51	**38**	1,031	505	1.6E+4	**482.97**	**0.04**	**0.00**
hc7p	**2.44**	18	865	171	**1.9E+4**	**703.79**	**0.07**	**0.00**	3.30	**16**	**739**	**47**	**1.9E+4**	777.06	0.09	**0.00**
hc7u	**20.73**	**33**	**1,977**	**859**	**6.8E+4**	**1,560.09**	**0.42**	**0.00**	69.06	50	2,747	1,193	8.3E+4	2,086.35	1.03	**0.00**
hc8p	**72.16**	27	2,607	826	3.9E+5	3,409.32	1.51	**0.01**	74.84	**26**	**2,366**	**696**	**2.2E+5**	**3,290.78**	**1.19**	**0.01**
hc8u	**927.21**	**57**	**6,699**	**2,975**	**6.5E+5**	**6,707.88**	**13.99**	**0.01**	2,238.91	82	9,794	4,755	8.7E+5	8,545.57	23.79	0.02
hc9p	**145.36**	**19**	**3,520**	361	1.0E+6	**7,580.50**	**4.36**	**0.01**	178.03	**19**	3,521	**360**	**9.7E+5**	7,777.80	4.59	**0.01**
i640-001	3.10	85	446	27	8.4E+3	195.13	**0.01**	**0.00**	**2.68**	**69**	**369**	**18**	**3.3E+3**	**184.34**	**0.01**	**0.00**
i640-011	**13.94**	**91**	**582**	131	3.0E+4	**205.73**	**0.02**	**0.01**	17.84	**91**	586	**123**	**1.4E+4**	208.81	**0.02**	**0.01**
i640-031	**4.36**	80	492	43	**1.3E+4**	**222.83**	**0.01**	**0.00**	5.05	**74**	**423**	**39**	2.9E+4	248.60	0.02	**0.00**
i640-101	14.87	57	1,179	114	5.1E+4	814.97	0.14	**0.00**	**13.96**	**49**	**1,009**	**96**	**4.7E+4**	**748.16**	**0.11**	**0.00**
i640-131	**20.24**	63	1,457	**170**	8.9E+4	724.56	0.13	**0.00**	21.39	**53**	**1,243**	178	**7.6E+4**	**700.80**	**0.11**	0.01
i640-201	**9.83**	47	1,570	310	2.5E+4	692.06	**0.07**	**0.00**	11.02	**40**	**1,409**	**227**	**1.9E+4**	**675.88**	**0.07**	**0.00**
i640-231	192.42	69	3,279	283	**3.2E+5**	3,212.54	2.10	**0.01**	**179.38**	**62**	**2,946**	**277**	3.6E+5	**2,933.60**	**1.78**	0.02
i640-301	39.45	38	5,483	2,383	5.4E+4	1,344.69	0.29	**0.00**	**32.90**	**24**	**3,723**	**1,577**	**3.2E+4**	**1,143.96**	**0.14**	0.01
i640-331	**552.52**	84	**13,072**	**6,062**	7.7E+5	3,565.32	4.40	**0.01**	671.41	**84**	13,280	6,379	**4.6E+5**	**3,379.21**	**4.03**	0.02
Aggr. saving (%)									−7	16	16	33	20	2	2	14

nonedge inequalities (that is, the stable set inequalities of cardinality 2), introduced as lazy cuts which are added by CPLEX only if violated. Thus, all the stable set inequalities that we introduce act as cutting planes which tighten the original formulation. Note that a similar cut-and-branch approach is not applicable to min Steiner tree in the x_{ij} variable space. Indeed, since any correct formulation for this problem requires all the $s-t$ cut set inequalities, a branch-and-cut approach is needed.

Since, in these experiments, we aim at computational efficiency, rather than generating the stable set inequalities with CPLEX, we use `Cliquer-1.21` [26], which is among the most efficient exact solvers for the max weight clique problem on sparse graphs. It is based on a combinatorial branch-and-bound, and is used, for instance, in state-of-the-art solvers for the graph coloring problem [20,21]. We solve the max weight stable set separation problem by looking for a max weight clique on the complement graph. At each iteration, the separation problem is solved to optimality and a single cut is added.

We consider a set of instances taken from the second DIMACS implementation challenge on max clique, graph coloring, and satisfiability [22]. A time limit of 1 h (3,600 s) is set for both cutting plane generation at the root node and for the subsequent branch-and-bound application.

In Table 3, for each instance and for both the revised standard and revised coordinated generation scheme, we report: the number of cuts generated (Cuts), the corresponding computing time at the root node (Root time), the number of branch-and-bound nodes (B&B nodes), and the overall computing time (Total time).

As far as the root node is concerned, the results are in line with those of the pure cutting plane setting and show on average, a reduction of 13 % in the number of cuts and of 22 % in the computing time. As to the overall results, we observe a substantial reduction of 54 % in the number of branch-and-bound nodes and of 24 % in the total computing time. The number of cuts and total computing time are simultaneously reduced on 13 instances out of 19 and, for the other 6 instances, at least one of the two figures is improved. Finally, the number of branch-and-bound nodes is reduced for 7 instances out of the 10 that are not solved to optimality at the root node.

6 Concluding remarks

We have proposed a new cutting plane generation scheme in which, among all the maximally violated valid inequalities of a given family, we generate one that is also undominated and maximally diverse w.r.t. the cuts that were previously found. For inequalities with binary left-hand-side coefficients and a constant right-hand side, our revised coordinated separation problem is equivalent to the standard separation problem of finding a maximally violated cut with different objective function coefficients. Computational results obtained in a pure cutting plane setting when separating stable set and cut set inequalities for the max clique and min Steiner tree problems indicate that we can close the same fraction of the duality gap in a considerably smaller number of cuts or rounds, obtaining a final LP relaxation which is numerically more stable. For max clique instances, the computing time is also substantially reduced. Experiments

Table 3 Comparison between revised standard separation and revised coordinated separation on max clique instances in a cut-and-branch setting

	Revised standard separation				Revised coordinated separation			
	Cuts	Root time	B&B nodes	Total time	Cuts	Root time	B&B nodes	Total time
C125.9	200	0.49	**24,244**	**25.27**	**180**	**0.37**	33,702	26.35
gen200_p0.9_44	1,093	14.63	2,161	40.09	**992**	**13.61**	**813**	**32.51**
gen200_p0.9_55	**310**	2.29	0	4.69	311	**2.24**	0	**4.6**
hamming10-2	**1,524**	**12.96**	0	**28.53**	**1,524**	14.31	0	29.24
hamming10-4	1,133	3,785.52	**16,132**	11,207.9	**795**	**2,034.75**	24,633	**7,701.49**
hamming8-2	369	**0.54**	0	1.13	**320**	**0.45**	0	**0.93**
hamming8-4	258	2.89	0	6.77	**137**	**1.19**	0	**3.54**
keller4	521	15.23	**89,529**	213.55	**408**	**9.75**	102,579	**205.45**
MANN_a27	**428**	**1.11**	42,268	20.81	436	1.17	**14,774**	**11.39**
MANN_a45	**1,175**	**13.91**	**289,485**	490.4	**1,175**	**13.94**	**289,485**	**490.01**
san200_0.7_1	217	1.21	0	2.73	**165**	**0.64**	0	**1.57**
san200_0.7_2	1,258	71.60	759	150.31	**1,211**	**60.62**	**319**	**125.8**
san200_0.9_1	190	**0.24**	0	0.58	**154**	**0.16**	0	**0.39**
san200_0.9_2	263	0.75	0	1.59	**217**	**0.46**	0	**1.01**
san200_0.9_3	739	4.21	4,676	20.01	**687**	**3.97**	**1,239**	**11.57**
san400_0.7_1	1,844	712.79	0	1,436.08	**1,639**	**537.47**	0	**1,081.88**
san400_0.7_2	**3,313**	**1,232.96**	70,023	4,410	3,551	1,453.15	**1,919**	**3,001.58**
san400_0.7_3	6,436	**3,661.14**	106,201	**10,921.16**	**6,145**	3,678.22	**52,200**	10,964.50
san400_0.9_1	752	27.03	0	54.86	**658**	**19.81**	0	**40.35**
Aggr. saving %					13	22	54	24

in a cut-and-branch setting for max clique indicate the potential of our scheme also when solving a problem to optimality.

Future developments include the investigation of alternative diversity measures and of different ways to enforce diversity when dealing with several families of cuts (diversity within each family or w.r.t. all the previously generated cuts), as well as the extension of the proposed approach to the case of inequalities with general integer coefficients.

Acknowledgments We would like to thank two anonymous referees for useful comments that helped to improve the quality of the paper.

Appendix

For the sake of completeness and reproducibility, we describe how we select an appropriate value for the parameter ϵ, as well as how we proceed to avoid numerical issues.

As stated in Sect. 3, when lexicographically optimizing $f_1, f_2 : \mathbb{R}^n \rightarrow \mathbb{R}$ by optimizing $f_1 + \epsilon f_2$, we select ϵ so as to satisfy the condition $\epsilon\Delta_2 < \delta_1$, where

Δ_2 is the difference between the maximum and minimum of f_2 and δ_1 is the smallest variation between any two values of f_1. For rational polyhedra where $x^* \in \mathbb{Q}^n$, δ_1 can be bounded from below by the reciprocal of any multiple of all the denominators of the components of x^*. On a computer, where x^* is usually represented as a floating point number, if p denotes the position of the least significant digit among all the components of x^*, then 10^{-p} is a valid lower bound on δ_1. Since only the first few digits are usually numerically significant, in our computational experiments we truncate the components of x^* to the 3rd decimal digit, thus bounding δ_1 from below by 10^{-3}.

As to the specific values for ϵ, for the revised standard separation problem where $f_2 = e\alpha$ takes values in $[0, n]$ we choose $\epsilon = \frac{10^{-3}}{n}$. In the revised coordinated separation problem, depending on the direction of the cut, we have either $f_2 = (3e - 2\bar{\alpha}^k)\alpha$ or $f_2 = (e + 2\bar{\alpha}^k)\alpha$, both taking values in $[0, 3n]$. Therefore, we choose $\epsilon = \frac{10^{-3}}{3n}$.

For problems with a large number of variables n, the parameter ϵ in the revised coordinated cutting plane generation scheme may be quite small. Since in the aggregate vector $\hat{x}$ the information concerning the cut diversity is entirely contained in the least significative digits, $\hat{x}$ must be handled with appropriate numerical precision. We proceed as follows. First, when truncating x^* with a precision of 10^{-3} we round it down for max clique and up for min Steiner tree. This amounts to slightly underestimating the violation of the cuts that we generate, thus avoiding the introduction of cuts which might appear to be violated only due to numerical issues. We also discard any cut with a violation smaller than 10^{-3}. Remember that, in our revised coordinated separation problem, the cut diversity is multiplied by $\epsilon = \frac{10^{-3}}{3n}$, where n is the number of nodes for max clique and of arcs for min Steiner tree. For the two problems, the smallest ϵ is obtained for the largest instances with, respectively, 500 and 10,000 variables, and amounts to $\frac{10^{-3}}{1{,}500} = 6.\bar{6}\ 10^{-7}$ and $\frac{10^{-3}}{30{,}000} = 3.\bar{3}\ 10^{-8}$, respectively. Note that for the revised standard case we have a larger $\epsilon = \frac{10^{-3}}{n}$ which amounts to $\frac{10^{-3}}{1{,}500} = 2\ 10^{-6}$ for max clique and to $\frac{10^{-3}}{30{,}000} = 2\ 10^{-7}$ for min Steiner tree. Then, we construct $\hat{x}$ by adding the diversity term multiplied by ϵ. The final vector $\hat{x}$ is then truncated with a precision of 10^{-8} for max clique problem and of 10^{-9} for min Steiner tree, so as to guarantee that at least one significant digit is preserved to represent the cut diversity in the revised coordinated separation problem. Finally, we turn $\hat{x}$ into an integer vector, multiplying it by 10^8 for max clique and by 10^9 for min Steiner tree.

References

1. Achterberg, T.: SCIP: solving constraint integer programs. Math. Program. C **1**, 1–41 (2009)
2. Amaldi, E., Coniglio, S., Gualandi, S.: Improving cutting plane generation with 0–1 inequalities by bi-criteria separation. In: Festa, P. (ed.) Experimental Algorithms, Lecture Notes in Computer Science, pp. 266–275. Springer, Berlin (2010)
3. Andreello, G., Caprara, A., Fischetti, M.: Embedding {0, 1/2}-cuts in a branch-and-cut framework: a computational study. INFORMS J. Comput. **19**(2), 229–238 (2007)
4. Balas, E., Ceria, S., Cornuéjols, G.: A lift-and-project cutting plane algorithm for mixed 0–1 programs. Math. Program. **58**(1–3), 295–324 (1993)
5. Balas, E., Ceria, S., Cornuéjols, G.: Mixed 0–1 programming by lift-and-project in a branch-and-cut framework. Manage. Sci. **42**(9), 1229–1246 (1996)

6. Balas, E., Ceria, S., Cornuèjols, G., Natraj, N.: Gomory cuts revised. Oper. Res. Lett. **19**, 1–9 (1996)
7. Balas, E., Fischetti, M., Zanette, A.: Lexicography and degeneracy: can a pure cutting plane algorithm work? Math. Program. A **1**, 153–176 (2011)
8. Balas, E., Saxena, A.: Optimizing over the split closure. Math. Program. A **113**, 219–240 (2008)
9. Ben-Ameur, W., Neto, J.: Acceleration of cutting-plane and column generation algorithms: applications to network design. Networks **49**, 3–17 (2007)
10. Chopra, S., Rao, M.R.: The Steiner tree problem I: formulations, compositions and extension of facets. Math. Program. **64**, 209–229 (1994)
11. Chopra, S., Rao, M.R.: The Steiner tree problem II: properties and classes of facets. Math. Program. **64**, 231–246 (1994)
12. Coniglio, S.: On coordinated cutting plane generation and mixed-integer programs with nonconvex 2-norm constraints. Ph.D. thesis, Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy (2011)
13. Coniglio, S.: On coordinated cutting plane generation and mixed-integer programs with nonconvex 2-norm constraints. 4OR: Quart. J. Oper. Res. 1–2 (2012). doi:10.1007/s10288-012-0199-7
14. Cook, W., Fukusawa, R., Goycoolea, M.: Choosing the best cuts. In: 3rd Mixed Integer Programming, Workshop (2006)
15. Desrosiers, J., Lübbeke, M.: A primer in column generation. In: Desaulniers, G., Desrosiers, J., Solomon, M.M. (eds.) Column Generation, GERAD 25th Anniversary, chap. 1, pp. 1–32. Springer, Berlin (2005)
16. Edmonds, J., Karp, R.: Theoretical improvements in algorithmic efficiency for network flow problems. J. ACM **19**(2), 248–264 (1972)
17. Fischetti, M., Lodi, A.: Optimizing over the first Chvàtal closure. Math. Program. A **110**(1), 3–20 (2006)
18. Fischetti, M., Salvagnin, D.: An in-out approach to disjunctive optimization. In: Lodi, A., Toth, P. (eds.) Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, Lecture Notes in Computer Science, pp. 136–140. Springer, Berlin (2010)
19. Gomory, R.: An algorithm for integer solutions to linear programs. In: Graves, R., Wolfe, P. (eds.) Recent Advances in Mathematical Programming, pp. 269–302. McGraw-Hill, New York (1963)
20. Gualandi, S., Malucelli, F.: Exact solution of graph coloring problems via constraint programming and column generation. INFORMS J. Comput. **24**, 81–100 (2012)
21. Held, S., Cook, W., Sewell, E.: Safe lower bounds for graph coloring. In: Günlük, O., Woeginger, G. (eds.) Integer Programming and Combinatorial Optimization, Lecture Notes in Computer Science, pp. 261–273. Springer, Berlin (2011)
22. Johnson, D., Trick, M.: Cliques, Coloring, and Satisfiability, DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 26. American Mathematical Society, Providence (1996)
23. Koch, T., Martin, A.: Solving Steiner tree problems in graphs to optimality. Networks **32**(3), 207–232 (1998)
24. Koch, T., Martin, A., Voß, S.: SteinLib: an updated library on steiner tree problems in graphs. In: Cheng, X., Du, D. (eds.) Steiner Trees in Industry, vol. 11, pp. 285–326. Kluwer, Dordrecht (2001)
25. Nemhauser, G., Wolsey, L.: Integer Programming and Combinatorial Optimization. Wiley, New York (1980)
26. Ostergard, P.: A fast algorithm for the maximum clique problem. Disc. Appl. Math. **120**(1–3), 197–207 (2002)
27. Phillips, N.: A weighting function for pre-emptive multicriteria assignment problems. J. Oper. Res. Soc. **38**, 797–802 (1987)
28. Wesselmann, F., Suhl, U.: Implementation techniques for cutting plane management and selection. Technical report, University of Paderborn, Paderborn (2007)
29. Williams, H.: Model Building in Mathematical Programming. Wiley, New York (1999)