A Family of Spanning-Tree Formulations for the Maximum Cut Problem

Sven Mallach

High Performance Computing & Analytics Lab
University of Bonn,
Friedrich-Hirzebruch-Allee 8, 53115 Bonn, Germany

December 29, 2023

Abstract

We present a family of integer programming formulations for the maximum cut problem. These formulations encode the incidence vectors of the cuts of a connected graph by employing a subset of the odd-cycle inequalities that relate to a spanning tree, and they require only the corresponding edge variables to be integral explicitly. They so describe sufficient restrictions of the classic integer linear program by Barahona and Mahjoub. In addition, we characterize according formulations comprising facet-defining inequalities only. Trade-offs and comparisons to prevalent formulations concerning size and relaxation strength are subject to an experimental study.

1 Introduction

Given an undirected graph $G = (V, E)$, a *cut* in G is an edge set $\delta(U) := \{ \{u, v\} \in E : u \in U, v \in V \setminus U \}$ induced by some *bi-partition* of V into $U \subseteq V$ and $V \setminus U$. Considering edge weights $c : E \to \mathbb{R}$, let $c(\delta(U)) := \sum_{e \in \delta(U)} c_e$ for any $U \subseteq V$. Then the *Maximum Cut Problem* (MaxCut) is to determine $U^* \subseteq V$ such that $c(\delta(U^*)) \geq c(\delta(U))$ for all $U \subseteq V$. Since the union of maximum cuts for the connected components of an undirected graph G give a maximum cut for G, we will assume throughout this paper that G is connected.

MaxCut is a classical combinatorial optimization problem that is $\mathscr{NP}$-hard in the general case [14], and that receives increasing interest in recent years. Besides direct applications in e.g. Image Segmentation [10], Frequency Assignment [6] and VLSI Design [1], a major reason for this interest is its direct correspondence to the Unconstrained Binary Quadratic Programming problem (UBQP) [2, 9, 11] that has itself numerous applications in research and economy.

A common approach to solve MaxCut instances to proven optimality is based on integer linear programming and polyhedral relaxations. Especially for sparse instances, such exact methods have proven increasingly effective recently [7, 18].

In this paper, we deduce a family of integer programming formulations for MaxCut that is based on spanning trees. On the one hand, these formulations constitute proper restrictions of the classical edge-variable formulation by Barahona and Mahjoub [3]. In particular, it is an immediate consequence of our results that, in order to still encode precisely the cuts in $G=(V,E)$, only $|V|-1$ of the edge variables in the classic formulation need an explicit enforcement of integrality and one may confine to an appropriate subset of its only set of non-trivial constraints, called *odd-cycle inequalities*, at the same time. We characterize natural approaches to determine such appropriate inequality subsets based on cycles related to a spanning tree of G, and we discuss the corresponding trade-offs that arise regarding formulation size and quality. In addition, we address how to obtain violated odd-cycle inequalities from these cycles by means of an efficient separation. On the other hand, the presented family of spanning-tree formulations generalizes on the so-called root-triangulated model for MaxCut from [8] as well as on the further classic formulation with node- and edge-variables that has been obtained in the light of UBQP transformations [9, 2, 17]. In our computational experiments, we compare the different formulations regarding their size and their strength in terms of their linear programming (LP) relaxations.

This paper is organized as follows: In Section 2, we briefly summarize fundaments and related work. The family of spanning-tree formulations for MaxCut is presented in Section 3 along with a correctness proof and improvements concerning the size and the strength of the formulations. In Section 4, we report on our computational experiments, and finally, a conclusion is given in Section 5.

2 Preliminaries and Related Work

As is common especially in (integer) linear programming approaches to MaxCut, we will identify edge subsets $S \subseteq E$ with their *incidence vectors* $\chi^S \in \mathbb{R}^E$ where $\chi^S_e \in \{0,1\}$ and $\chi^S_e = 1$ if and only if $e \in S$. Based on this notion, the *cut polytope* as defined by Barahona and Mahjoub in [3] is the convex hull of all the incidence vectors of cuts in $G=(V,E)$:

$$P_{\mathrm{CUT}}(G) = \mathrm{conv}\{\chi^S \mid S \subseteq E \text{ is a cut in } G\}$$

It is well-known that an edge subset $S \subseteq E$ is a cut in G if and only if S intersects with every cycle in G in an *even* number (possibly zero) of edges. Exactly this condition is established (for $S := \{e \in E : x_e = 1\}$) by the constraints (1), called *odd-cycle inequalities*, in the seminal binary linear programming formulation by Barahona and Mahjoub [3] that we refer to as the edge model (E).

$$\text{maximize } \sum_{e\in E} c_e x_e \qquad \text{(E)}$$

$$\sum_{e\in S} x_e - \sum_{e\in C\setminus S} x_e \le |S|-1 \qquad \text{for all cycles } C \subseteq E \text{ and all } S \subseteq C,\ |S| \text{ odd} \qquad (1)$$

$$0 \le x_e \le 1 \qquad \text{for all } e \in E \qquad (2)$$

$$x_e \in \{0,1\} \qquad \text{for all } e \in E$$

Given a cycle $C \subseteq E$, an edge $e \in E \setminus C$ that is incident to two nodes of C is called a *chord of C*. An odd-cycle inequality defines a facet of $P_{\mathrm{CUT}}(G)$ if and only if the corresponding cycle C is chordless [3]. We also recall from the same reference that, although we do not explicitly state this in (E), the lower and upper bounds on the variables (2) are implied by (11) for those edges which are part of a triangle in G. When removing the integrality restrictions, we obtain

$$P_{\mathrm{OC}} := \{x \in \mathbb{R}^E : x \text{ satisfies (1), (2)}\}$$

as the feasible set, giving rise to the LP relaxation $\max\{c^\top x : x \in P_{\mathrm{OC}}\}$ of (E) which provides an upper bound on the value of a maximum cut in G.

An alternative and also well-known integer linear programming formulation for MaxCut [9, 2, 17] is given by the following node-edge model (NE):

$$\text{maximize} \sum_{\{i,j\}\in E} c_{ij}x_{ij} \qquad \text{(NE)}$$

$$x_{ij} + z_i + z_j \leq 2 \quad \text{for all } \{i,j\} \in E \tag{3}$$

$$x_{ij} - z_i - z_j \leq 0 \quad \text{for all } \{i,j\} \in E \tag{4}$$

$$-x_{ij} + z_i - z_j \leq 0 \quad \text{for all } \{i,j\} \in E \tag{5}$$

$$-x_{ij} - z_i + z_j \leq 0 \quad \text{for all } \{i,j\} \in E \tag{6}$$

$$z_i \in \{0,1\} \quad \text{for all } i \in V$$

In (NE), the integrality restrictions are imposed only on the additional node-variables $z \in \mathbb{R}^V$ which then imply the integrality of $x \in \mathbb{R}^E$ via the constraints (3)–(6). These inequalities also imply $0 \leq z \leq 1$ and $0 \leq x \leq 1$.

The node variables z_i, $i \in V$, in (NE) may also be regarded as edge variables in terms of i and an additional auxiliary node that is connected to all original ones. In [8], Charfreitag et al. take on this perspective and propose to refine (NE) by choosing an original node $r \in V$ to adopt the role of the auxiliary one. Consequently, in the resulting root-triangulated model (RT), the original edge set is (only) extended to $E' := E \cup \{\{r,v\} : v \in V,\ \{r,v\} \notin E\}$.

$$\text{maximize} \sum_{\{i,j\}\in E} c_{ij}x_{ij} \qquad \text{(RT)}$$

$$x_{ij} + x_{ri} + x_{rj} \leq 2 \quad \text{for all } \{i,j\} \in E, r \notin \{i,j\} \tag{7}$$

$$x_{ij} - x_{ri} - x_{rj} \leq 0 \quad \text{for all } \{i,j\} \in E, r \notin \{i,j\} \tag{8}$$

$$-x_{ij} + x_{ri} - x_{rj} \leq 0 \quad \text{for all } \{i,j\} \in E, r \notin \{i,j\} \tag{9}$$

$$-x_{ij} - x_{ri} + x_{rj} \leq 0 \quad \text{for all } \{i,j\} \in E, r \notin \{i,j\} \tag{10}$$

$$x_{ri} \in \{0,1\} \quad \text{for all } \{r,i\} \in E'$$

In (RT), the inequalities (7)–(10), substituting for (3)–(6), now appear as triangle inequalities (i.e., odd-cycle inequalities for $|C| = 3$) for each $\{i,j\} \in E$, $r \notin \{i,j\}$, and the corresponding unique "root-triangle" in E'. These triangle inequalities again imply $0 \leq x_{ij} \leq 1$ for all $\{i,j\} \in E'$, and their presence combined with the integrality of the variables x_{ri}, for all $i \in V \setminus \{r\}$, suffices to establish the integrality of all variables. Indeed if $\bar{x}_{ri}, \bar{x}_{rj} \in \{0,1\}$ one has $\bar{x}_{ij} \in \{0,1\}$ because the corresponding instances of (7)–(10) then reduce to $\bar{x}_{ij} = 0$ (if $\bar{x}_{ri} = \bar{x}_{rj}$) or $\bar{x}_{ij} = 1$ (if $\bar{x}_{ri} = 1 - \bar{x}_{rj}$) [16, 8].

Compared to (E), (NE) and (RT) are extended by at most $|V|$ variables, and they contain only a small subset of the odd-cycle or, more precisely, triangle inequalities. As such, their LP relaxations provide only a weak bound on the value of a maximum cut, but they may serve as a starting point to be successively enriched by further odd-cycle inequalities, and thus promoted to be as strong as the one of (E) [8]. From a converse perspective, the strength of P_{OC} can principally be retained even solely by triangle inequalities at the expense of augmenting G with (zero-weight) edges, respectively variables, to a chordal graph, like e.g. in [16]. This is because then the resulting triangle inequalities become the only facet-defining, i.e., irredundant, odd-cycle inequalities.

3 A Family of Spanning-Tree Formulations for MaxCut

In this section, we characterize a family of integer programming formulations for Max-Cut that is based on the selection of a spanning edge subset of the original graph G, and whose sets of constraints and variables with an explicit integrality requirement are (usually strict) subsets of those in (E). In contrast to the further models addressed in Sect. 2, the spanning-tree formulations do not involve any further variables and they do not restrict to triangle inequalities. Instead, other appropriate subsets of the odd-cycle inequalities (1) are identified that prove sufficient to restrict the feasible set to the incidence vectors of cuts in G, provided that (only) the variables corresponding to the spanning edge set are integral. More precisely, they consistently transfer the integrality of these variables to the others such that effectively *all* odd-cycle inequalities for G are satisfied, exactly as this is achieved by the triangle inequalities in (RT) as well.

We will use the following basic terminology with respect to spanning trees and extend it appropriately during the course of the discussion.

Definition 1. *Let $T = (V, E_T)$ be a spanning tree of a connected undirected graph $G = (V, E)$, i.e., $E_T \subseteq E$, $|E_T| = |V| - 1$, and T is connected. Let $e = \{i, j\} \in E \setminus E_T$, and let T_e be the unique i-j-path in T. Then, we call $C_e = T_e \cup \{e\} \subseteq E$ an* elementary cycle *(w.r.t. T).*

3.1 Basic Spanning-Tree Formulations

Given a graph $G = (V, E)$ and some spanning tree $T = (V, E_T)$ with $E_T \subseteq E$, we obtain a first family of new edge-variable formulations (ST) by restricting to the odd-cycle inequalities associated with the elementary cycles C_e, $e \in E \setminus E_T$.

$$\text{maximize} \sum_{e \in E} c_e x_e \qquad \text{(ST)}$$

$$\sum_{e \in S} x_e - \sum_{e \in C_f \setminus S} x_e \leq |S| - 1 \qquad \text{for all elementary cycles } C_f \subseteq E,\ f \in E \setminus E_T, \text{ and all } S \subseteq C_f,\ |S| \text{ odd} \tag{11}$$

$$0 \leq x_e \leq 1 \qquad \text{for all } e \in E$$

$$x_e \in \{0, 1\} \qquad \text{for all } e \in E_T$$

Before we show that (ST) indeed encodes the incidence vectors of cuts in G, we remark that, in analogy to (E), the lower and upper bounds on the variables in (ST) are implied by (11) for those edges which take part in a triangle that serves as one of the corresponding elementary cycles $C_f \subseteq E$, $f \in E \setminus E_T$.

Theorem 1. *Let $\bar{x} \in \mathbb{R}^E$ be a solution to (an instance of) (ST) associated with $G = (V,E)$. Then $\bar{x}$ is integral and the incidence vector of a cut in G.*

Proof. Concerning integrality, since $\bar{x}_e \in \{0,1\}$ is enforced explicitly in (ST) for all $e \in E_T$, it suffices to show that $\bar{x}_e \in \{0,1\}$ for all $e \in E \setminus E_T$. For any such edge $e \in E \setminus E_T$, consider the corresponding elementary cycle C_e which consists of $|C_e| - 1$ edges from E_T and e. The odd-cycle inequalities for C_e are satisfied by $\bar{x}$. Each of them can be written as $\sum_{f \in S}(1 - x_f) + \sum_{f \in C_e \setminus S} x_f \geq 1$ for all $S \subseteq C_e$, $|S|$ odd. Let $O := \{f \in C_e \setminus \{e\} : \bar{x}_f = 1\}$ and let $k := |O|$. If k is odd, consider that $\sum_{f \in O}(1 - \bar{x}_f) + \sum_{f \in C_e \setminus O} \bar{x}_f \geq 1$ reduces to $\bar{x}_e \geq 1$. If k is even, consider that $\sum_{f \in O \cup e}(1 - \bar{x}_f) + \sum_{f \in C_e \setminus (O \cup e)} \bar{x}_f \geq 1$ reduces to $(1 - \bar{x}_e) \geq 1 \Leftrightarrow -\bar{x}_e \geq 0 \Leftrightarrow \bar{x}_e \leq 0$. Since $0 \leq \bar{x}_e \leq 1$ is enforced explicitly in (ST), we have $\bar{x}_e = 1$ in the first, and $\bar{x}_e = 0$ in the second case. Thus, $\bar{x}_e$ is integral for all $e \in E \setminus E_T$. Moreover, since the edge set $C_e \setminus \{e\} \subseteq E_T$ corresponds to the unique simple path between the two endpoints of each $e = \{i,j\}$ in T, the previous arguments show that $\bar{x}_e = 1$ ($\bar{x}_e = 0$) if, when traversing this simple path, the partitions of the nodes change an odd (even) number of times. i.e., if i and j belong to different (the same) partition(s). Thus, the contradiction-free partitioning of $V(T)$ given by the restriction of $\bar{x}$ to the components for E_T[1] is consistently imposed on the components for $E \setminus E_T$ by (11). In other words, $\bar{x}$ is the incidence vector of a cut. $\square$

Each choice of the spanning edge set $E_T \subseteq E$ for (ST) gives rise to a concrete integer program which leads to different variables with an explicit integrality requirement, and to different elementary cycles whose lengths influence the size of the formulation as well as the upper bound on the value of a maximum cut provided by its LP relaxation[2]. Thereby, the number of odd-cycle inequalities associated with each of the $|E| - |V| + 1$ elementary cycles C_f in (11) is the respective sum of $\binom{|C_f|}{k}$ over all $k \leq |C_f|$, k odd, and thus strongly increases with their length. Moreover, from a general perspective, the quality of the upper bound is expected to be weak (irrespective of the choice of E_T), as is also visible from the experiments in Sect. 4. In particular, the selection (11) of odd-cycle inequalities is not necessarily facet-defining since the associated cycles need not be chordless. This can in turn be used to reduce the size of the formulation and to strengthen its LP relaxation at the same time, as is exposed in the following.

3.2 Improved Spanning-Tree Formulations

We discuss two strategies to improve over (ST) by focusing on facet-defining odd-cycle inequalities deduced from a given spanning tree $T = (V, E_T)$ of $G = (V,E)$. A first idea could be to replace (11) by all the chordless cycles that are composed from

[1] Indeed, one may choose an arbitrary root $r \in V$, assign it to any of the two partitions, and then determine the partition of every $v \in V \setminus \{r\}$ based on the $\bar{x}_e$ for the edges e on the unique r-v-path in T.

[2] Analogously, (RT) defines a family of integer programs as well, with varying edges and associated variables to be added and different triangle inequality sets based on the choice of the root node $r \in V$. However, the corresponding number of triangle inequalities is always $4 \cdot (|E| - d(r))$ where $d(r)$ is the degree of r in G.

the edges of C_e and its (possibly empty set of) chords. Indeed, the corresponding set $\mathscr{C}_e$ of chordless cycles is of relevance and thus formalized in the first part of Definition 2 below. However, the corresponding properties may apply to a single chordless cycle for more than one $e \in E \setminus E_T$. In particular, each chord $f \in E$ of an elementary cycle C_e, $e \in E \setminus E_T$, gives rise to an elementary cycle C_f (with $C_f \cap E_T \subsetneq C_e \cap E_T$) itself, and some chords of C_e may also be chords of C_f (this is illustrated in Fig. 1). As opposed to that, the particular chordless cycles in $\mathscr{C}_e$ *that involve* e, formalized as the set $\mathscr{C}_e^e$ in the subsequent Definition 2 uniquely relate to the elementary cycles C_e, $e \in E \setminus E_T$.

Definition 2. *Let $C_e \subseteq E$ be the unique elementary cycle of $G = (V,E)$ w.r.t. the spanning tree $T = (V, E_T)$ and $e \in E \setminus E_T$. Moreover, let $D_e \subseteq E \setminus C_e$ be the set of chords of C_e. Then the* set of chordless cycles associated with C_e *is the set* $\mathscr{C}_e := \{F \subseteq C_e \cup D_e :$ *F chordless cycle, $F \cap C_e \neq \emptyset\}$. Further, the subset of these cycles containing e is referred to as $\mathscr{C}_e^e := \{F \subseteq C_e \cup D_e : F$ chordless cycle, $e \in F \cap C_e\}$.*

The sets $\mathscr{C}_e^e$ facilitate to generate chordless cycles and associated odd-cycle inequalities only once during a step-wise consideration of elementary cycles. Even more, it turns out that they are sufficient to replace the inequalities (11) of (ST).

The first ingredient to see this is that the odd-cycle inequalities for C_e are satisfied by $x \in \mathbb{R}^E$ as soon as x satisfies the odd-cycle inequalities corresponding to a set of (chordless) cycles whose union contains C_e [3, 16, 13]. We formalize such cycle sets in the present context as follows.

Definition 3. *Let $\mathscr{C}_e$ be the set of chordless cycles associated with the unique elementary cycle $C_e \subseteq E$ of $G = (V,E)$ w.r.t. the spanning tree $T = (V, E_T)$ and $e \in E \setminus E_T$. Then a subset $\mathscr{C}_e^* \subseteq \mathscr{C}_e$ such that $C_e \subseteq \bigcup_{C \in \mathscr{C}_e^*} C$ is called a* chordless composition of C_e. *Moreover, if $\mathscr{C}_e^*$ is a chordless composition of C_e but $\mathscr{C}_e^* \setminus C$ is not for any $C \in \mathscr{C}_e^*$, then $\mathscr{C}_e^*$ is called* irreducible.

While it is clear from Definition 3 that $\mathscr{C}_e$ is by itself a chordless composition of C_e, it is instructive to observe that one can obtain a (typically smaller) chordless composition of C_e by unifying (and thus restricting to) the sets $\mathscr{C}_f^f$ where either $f = e$ or $f \in D_e$ according to Definition 2. This is due to the aforementioned fact that each chord of C_e defines an elementary cycle - which also needs to be covered by a chordless decomposition - itself. That is, after generating the chordless cycles in $\mathscr{C}_e^e$, we can rely on the sets $\mathscr{C}_f^f$, $f \in D_e$, to gradually and jointly obtain a chordless composition for C_e, as is exemplified in Fig. 1.

The first improved formulation, referred to as ($\mathrm{ST_{cl}}$), is then naturally obtained from (ST) by replacing the odd-cycle inequalities (11) for the elementary cycles C_e, $e \in E \setminus E_T$, with those for the cycles contained in the sets $\mathscr{C}_e^e$.

$$\text{maximize} \sum_{e \in E} c_e x_e \qquad (\mathrm{ST_{cl}})$$

$$\sum_{e \in S} x_e - \sum_{e \in C \setminus S} x_e \leq |S| - 1 \qquad \text{for all } C \in \mathscr{C}_f^f,\ f \in E \setminus E_T, \text{ and all } S \subseteq C,\ |S| \text{ odd} \tag{12}$$

$$0 \leq x_e \leq 1 \qquad \text{for all } e \in E$$

$$x_e \in \{0,1\} \qquad \text{for all } e \in E_T$$

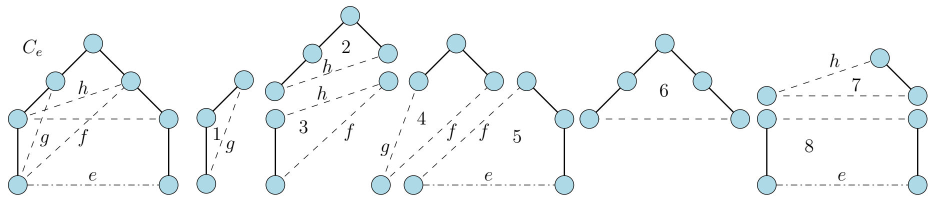

Figure 1: An elementary cycle C_e (left) with $C_e \cap E_T$ solid, e dash-dotted, and chords D_e dashed, three of which are labeled (f, g, and h). Several chordless cycles, labeled 1-8 and building $\mathscr{C}_e$ as of Definition 2, could be extracted from their union. In turn, any composition (union) of these chordless cycles covering all edges of C_e, like e.g. 1-4-5, is referred to as a *chordless composition of* C_e according to Definition 3. To obtain one for C_e, it suffices to consider the union of (even each time a single element of) the *chordless cycle sets* $\mathscr{C}_d^d$ according to Definition 2 where either $d = e$ or $d \in D_e$. For example, for C_e itself one may choose from $\mathscr{C}_e^e$ either the cycle 5 or 8. When choosing cycle 5, one may then rely on a chordless cycle from $\mathscr{C}_f^f$ (either 3 or 4), and then on either cycle 1 (from $\mathscr{C}_g^g$) or 2 (from $\mathscr{C}_h^h$), respectively, to jointly obtain a complete composition for C_e.

($\mathrm{ST_{cl}}$) replaces each C_e by possibly multiple chordless cycles $\mathscr{C}_e^e$ which are then however shorter (at most two chordless cycles of length $|C_e|-1$ involving e are possible, each of which could be substituted for by at most two such cycles of length $|C_e|-2$, and so on). Due to their binomial nature, ($\mathrm{ST_{cl}}$) has thus (typically strictly) less inequalities than (ST) while its LP relaxation provides a (strictly) better upper bound on the value of a maximum cut in G, as is also demonstrated in Sect. 4.

In practice, given an elementary cycle C_e, $e \in E \setminus E_T$, the set $\mathscr{C}_e^e$ can be derived e.g. by recursive split operations based on the chords of C_e as described in [13] combined with backtracking. Since only cycles containing e are of interest, one may thereby neglect any chord of C_e that is also a chord of a shorter elementary cycle, and truncate the search using the node numbering described below.

At the expense of a weakening of the LP relaxation compared to ($\mathrm{ST_{cl}}$) but still improving over (ST), a further reduction in size is possible, since a chordless composition $\mathscr{C}_e^*$ of each C_e, $e \in E \setminus E_T$, is already obtained by taking only one element (chordless cycle) from each set $\mathscr{C}_f^f$ where either $f = e$ or $f \in D_e$. This way, one is guaranteed to obtain a minimum *total number* of chordless cycles (by the necessity to have one per C_e, $e \in E \setminus E_T$) while these cycles may still, and possibly inevitably, induce a chordless composition for (larger) elementary cycles that is not irreducible in the sense of Definition 3.

Consequently, we obtain another sufficient formulation that we refer to as ($\mathrm{ST_{cl}^{co}}$) by replacing the odd-cycle inequalities (11) for the elementary cycles C_e, $e \in E \setminus E_T$ in (ST), by the odd-cycle inequalities for exactly one cycle $C \in \mathscr{C}_e^e$.

$$
\begin{aligned}
&\text{maximize} \sum_{e \in E} c_e x_e \quad (\mathrm{ST}_{\mathrm{cl}}^{\mathrm{co}}) && \\
&\sum_{e \in S} x_e - \sum_{e \in C \setminus S} x_e \le |S| - 1 && \text{for } \textbf{one } C \in \mathscr{C}_f^f,\ f \in E \setminus E_T, \\
& && \text{and all } S \subseteq C,\ |S| \text{ odd} \qquad (13) \\
&0 \le x_e \le 1 && \text{for all } e \in E \\
&x_e \in \{0,1\} && \text{for all } e \in E_T
\end{aligned}
$$

In order to determine the desired chordless cycles for each elementary cycle efficiently, no backtracking is required anymore and one may proceed as follows: For each $e = \{i, j\} \in E \setminus E_T$, consider the unique i-j-path T_e in T that builds C_e with e. Enumerate the nodes of T_e increasingly, starting at i. Then construct a path $R_e \subseteq T_e \cup D_e$, where D_e are the chords of C_e, as follows. Initialize $v \in V(T_e)$ with i. Determine $w \in V(T_e)$ among the nodes $T_e \cup D_e$-adjacent to v such that w has the maximum index w.r.t. to the created numbering. Add the respective edge to R_e, and continue with replacing v by w unless $w = j$. Finally, create the chordless cycle $C = R_e \cup e \in \mathscr{C}_e^e$.

3.3 Separation of Odd-Cycle Inequalities for a Given Cycle

Given $x \in \mathbb{R}^E$, $0 \le x \le 1$, the general separation problem for the odd-cycle inequalities asks for either an odd-cycle inequality that is violated by x, or for a proof that none exists. Barahona and Mahjoub [3] showed that it can be solved in polynomial time. Decades later, the algorithm has been refined to produce only facet-defining inequalities in terms of an a posteriori extraction described in [12] and [13]. Further ideas on such an extraction have been described in [18].

Besides that, the special separation problem associated with a *given* cycle $C \subseteq E$ is not well addressed by the general separation approach, but of interest (in the present context and beyond, e.g., for spin-glass problems [15]) as well. Here, one may exploit that at most one of the odd-cycle inequalities associated with C can be violated at a time [19, 8]. Based on this, given $x \in \mathbb{R}^E$, $0 \le x \le 1$, the separation problem w.r.t. C can be solved in linear time as follows:

1. If $\sum_{e \in C: x_e > \frac{1}{2}} (1 - x_e) + \sum_{e \in C: x_e \le \frac{1}{2}} x_e \ge 1$, there is no odd-cycle inequality for C violated by x. STOP
2. (Otherwise:) If $\left\{e \in C : x_e > \frac{1}{2}\right\}$ is odd, return the violated inequality. STOP
3. Otherwise, switch the role of $e^* = \arg\min_{e \in C} \left|\frac{1}{2} - x_e\right|$. If violated, return the inequality obtained, otherwise no odd-cycle inequality for C is violated by x.

Thereby, the condition in the first statement is a necessary one for the cycle C to admit a violated odd-cycle inequality. If $\left\{e \in C : x_e > \frac{1}{2}\right\}$ is odd, it is also sufficient. Otherwise, the increase of the left-hand side when switching the role of any edge $e \in C$ is precisely $2\left|\frac{1}{2} - \bar{x}_e\right|$. Thus, selecting for this purpose (any) edge that minimizes this increase gives another necessary and sufficient condition [19].

In particular, one may employ this procedure to obtain a cutting plane algorithm for (ST), ($\mathrm{ST}_{\mathrm{cl}}$), and ($\mathrm{ST}_{\mathrm{cl}}^{\mathrm{co}}$) after pre-computing the corresponding cycles defining (11), (12), and (13), respectively. Moreover, in case of (ST), and ($\mathrm{ST}_{\mathrm{cl}}^{\mathrm{co}}$), at most $|E|-|V|+1$ calls to the separation procedure are necessary per iteration.

4 Computational Experiments

The family of spanning-tree-based MaxCut formulations presented in Sect. 3 naturally relates to the - in terms of odd-cycle inequalities - complete edge-based integer program (E), and the model (RT) that potentially introduces additional variables but restricts to triangle inequalities. None of these formulations suits for the solution of a broader class of (sparse) MaxCut instances "per se", but requires a sophisticated branch-and-cut approach involving at least the dynamic separation of odd-cycle inequalities and further ingredients like e.g. in [7, 8, 18].

A particularly interesting question in this respect as well as in the focus and context of this paper is how the qualities of the linear programming relaxations of (ST), ($\mathrm{ST}_{\mathrm{cl}}$), and ($\mathrm{ST}_{\mathrm{cl}}^{\mathrm{co}}$) relate to each other and to those of (E) and (RT). That is, we strive to quantify the impact of the respective odd-cycle inequality selection in terms of the obtained upper bounds on the value of a maximum cut, and to relate this quality to the actual size of the respective linear program.

To carry out such an experiment, a reasonable choice needs to be made for the several degrees of freedom provided by the different formulations. To this end, we adopt the rule from [8] to choose a node of maximum degree as the root node for (RT), and we sort the edges non-increasingly w.r.t. their absolute weight in order to iteratively select them if suitable to build a spanning tree for (ST), ($\mathrm{ST}_{\mathrm{cl}}$) or ($\mathrm{ST}_{\mathrm{cl}}^{\mathrm{co}}$). Finally, for ($\mathrm{ST}_{\mathrm{cl}}^{\mathrm{co}}$), we compute the chordless compositions exactly by means of the procedure described in Sect. 3.2.

For these representants of each formulation family, the experiments are carried out with the following established instances from the Biq Mac Library [5]:

- g05_60, 80, 100: Graphs with 60, 80, or 100 nodes, unit edge weights.
- pm1d_80, 100, pm1s_100: Graphs with 80 or 100 nodes, ± 1 edge weights.
- pw01_100, pw05_100, pw09_100: Graphs with 100 nodes, and integer edge weights from $[1,10]$.
- be200.8: Graphs corresponding to the transformation of UBQP instances with $n=200$ generated by Billionnet and Elloumi [4] where the entries of the original cost matrices are integer from $[-100,100]$ for the diagonal entries and $[-50,50]$ for off-diagonal entries, and their non-zero density is about 0.8.

Each set consists of ten instances. Their selection is based on combining different sizes, densities, and edge weight ranges, among graphs with known and varying maximum chordless cycle length, denoted as $\max|C_{\mathrm{cl}}|$ in Table 1. We remark that the latter is only an upper bound on the maximal chordless cycle length observed when constructing ($\mathrm{ST}_{\mathrm{cl}}$) or ($\mathrm{ST}_{\mathrm{cl}}^{\mathrm{co}}$). The first three sets comprise native MaxCut instances generated as Erdös-Renyi-Gilbert random graphs [5]. We only include one set of native UBQP instances since their transformation to MaxCut (involving a node adjacent to all others via edges of relatively high absolute weight) leads to the situation that many of the edges forming the spanning trees of (RT), and (ST), ($\mathrm{ST}_{\mathrm{cl}}$), or ($\mathrm{ST}_{\mathrm{cl}}^{\mathrm{co}}$), coincide.

Group	$\lvert V\rvert$	Dsty	max $\lvert C_{\mathrm{cl}}\rvert$	Bound Ratio to LP Rel. of (E) (ST)	($\mathrm{ST_{cl}}$)	($\mathrm{ST_{cl}^{co}}$)	(RT)	+W	Constraint-to-Edge Ratios (ST)	($\mathrm{ST_{cl}}$)	($\mathrm{ST_{cl}^{co}}$)	(RT)	$\frac{\lvert E'\rvert}{\lvert E\rvert}$ (RT)
g05	60	0.50	11	1.45	1.27	1.31	1.46	1.50	22.63	7.44	4.85	3.81	1.02
g05	60	0.50	11	1.46	1.31	1.34	1.47	1.50	78.18	11.33	5.12	3.83	1.02
g05	60	0.50	11	1.46	1.37	1.39	1.47	1.50	234.22	13.76	5.52	3.84	1.03
g05	80	0.50	12	1.47	1.32	1.34	1.47	1.50	44.44	10.14	4.79	3.87	1.02
g05	80	0.50	12	1.47	1.35	1.37	1.48	1.50	86.51	12.07	5.11	3.87	1.02
g05	80	0.50	12	1.47	1.37	1.39	1.48	1.50	221.88	14.94	5.32	3.88	1.02
g05	100	0.50	13	1.47	1.36	1.38	1.48	1.50	32.48	9.28	4.96	3.90	1.01
g05	100	0.50	13	1.47	1.37	1.39	1.48	1.50	105.67	11.85	5.33	3.90	1.01
g05	100	0.50	13	1.47	1.39	1.40	1.48	1.50	506.03	17.46	5.62	3.90	1.02
pm1d	80	0.99	4	2.80	1.85	2.52	2.80	2.87	13.93	8.82	3.90	3.90	1.00
pm1d	80	0.99	4	2.94	1.94	2.66	2.93	3.01	23.36	10.13	3.90	3.90	1.00
pm1d	80	0.99	4	3.14	2.10	2.87	3.14	3.23	65.73	12.37	3.90	3.90	1.00
pm1d	100	0.99	4	2.79	1.89	2.51	2.79	2.85	26.00	10.98	3.92	3.92	1.00
pm1d	100	0.99	4	2.91	1.98	2.62	2.91	2.97	46.12	12.79	3.92	3.92	1.00
pm1d	100	0.99	4	3.02	2.09	2.71	3.01	3.08	107.03	14.53	3.92	3.92	1.00
pm1s	100	0.10	37	1.76	1.43	1.50	1.83	1.90	541.63	54.87	30.88	3.81	1.15
pm1s	100	0.10	38.8	1.81	1.51	1.57	1.88	1.95	1167.65	69.04	46.06	3.85	1.16
pm1s	100	0.10	40	1.88	1.58	1.64	1.98	2.05	2179.39	103.19	65.23	3.88	1.17
pw01	100	0.10	37	1.25	1.11	1.14	1.28	1.32	3063.08	57.96	22.12	3.81	1.15
pw01	100	0.10	38.8	1.25	1.12	1.15	1.29	1.32	14764.12	113.81	41.99	3.85	1.16
pw01	100	0.10	40	1.26	1.13	1.16	1.30	1.33	289171.06	670.26	168.09	3.88	1.17
pw05	100	0.50	12	1.46	1.21	1.27	1.48	1.50	496.27	19.81	4.91	3.90	1.01
pw05	100	0.50	12.9	1.46	1.25	1.28	1.48	1.50	3288.65	26.12	5.15	3.90	1.02
pw05	100	0.50	13	1.46	1.28	1.30	1.48	1.50	21141.55	36.76	5.61	3.90	1.02
pw09	100	0.90	6	1.47	1.27	1.32	1.48	1.50	395.50	18.79	3.92	3.91	1.00
pw09	100	0.90	6	1.47	1.29	1.34	1.48	1.50	1499.83	22.14	3.94	3.91	1.00
pw09	100	0.90	6	1.47	1.32	1.35	1.49	1.50	6832.52	25.62	3.96	3.92	1.00
be200.8	201	0.79	9	2.56	2.28	2.42	2.56	2.93	4.67	4.49	3.95	3.95	1.00
be200.8	201	0.79	9	2.61	2.34	2.47	2.61	3.00	5.04	4.69	3.95	3.95	1.00
be200.8	201	0.79	9	2.64	2.37	2.51	2.64	3.06	5.53	4.93	3.96	3.95	1.00

Table 1: Model Size and LP Relaxation Comparison: For each instance set, the rows show minimum, mean, and maximum values, respectively in this order, rounded to two decimal digits. The density (column "Dsty") shown is defined as $|E|/\binom{|V|}{2}$. The upper bounds provided by the LP relaxations of (ST), ($\mathrm{ST_{cl}}$), ($\mathrm{ST_{cl}^{co}}$), and (RT), as well as the sum of positive edge weights (+W), are shown as a factor in terms of the bound provided by P_{OC}. The number of constraints of each of the formulations is displayed as a factor of the number of edges of the instances (Constraint-to-Edge Ratios).

The results in Table 1 demonstrate well the trade-offs arising between the (chosen representants) of the different formulation families. The formulation (RT) and the largest formulation (ST), which however typically comprises a large fraction of odd-cycle inequalities that are not facet-defining, provide the weakest LP relaxations, with an upper bound often only slightly below the sum of positive edge weights. Constructing instead all facet-defining odd-cycle inequalities involving the non-tree edge closing an elementary cycle lets ($\mathrm{ST_{cl}}$) admit the strongest LP relaxation while its size is considerably reduced over (ST) but often still rather large, especially for the sparse instance sets. The quality gap of the relaxation of ($\mathrm{ST_{cl}}$) compared to the one of (E), i.e., P_{OC}, can still be significant. Finally, $\mathrm{ST_{cl}^{co}}$ provides an effective reduction to only one chordless cycle per non-tree edge. Its number of odd-cycle inequalities typically only slightly exceeds the one of (RT) for the dense instances while providing a significantly better upper bound. At the same time, $\mathrm{ST_{cl}^{co}}$ can still become quite large (pm1s, pw01) and lose strength (pm1d) compared to ($\mathrm{ST_{cl}}$). Fortunately, for all the other instances, the size of $\mathrm{ST_{cl}^{co}}$ as well as the loss in terms of the upper bound compared to ($\mathrm{ST_{cl}}$) is moderate.

5 Conclusion

We have presented families of spanning-tree formulations for the Maximum Cut Problem that encode incidence vectors of cuts via a subset of the odd-cycle inequalities associated with a connected graph $G = (V, E)$, and that require only $|V| - 1$ edge variables to be integral explicitly. Further, two variants have been described which are reduced in size and consist of facet-defining inequalities only. In an experimental study, it has been shown that for rather dense problem instances, one may obtain a formulation that moderately exceeds the size of a common integer program comprising only triangle inequalities while providing a stronger linear programming relaxation. For sparse instances, respectively for graphs with long chordless cycles, even the reduced spanning-tree formulations can become quite large. Generally, the experimental results show an interesting trade-off in terms of size and relaxation quality which motivates further research regarding the relevance of certain subsets of the odd-cycle inequalities. Moreover, they demonstrate the importance of selecting facet-defining inequalities which lead to a significantly better and more compact relaxation at the same time.

References

[1] Francisco Barahona, Martin Grötschel, Michael Jünger, and Gerhard Reinelt. An application of combinatorial optimization to statistical physics and circuit layout design. *Operations Research*, 36(3):493–513, 1988.

[2] Francisco Barahona, Michael Jünger, and Gerhard Reinelt. Experiments in quadratic 0–1 programming. *Mathematical Programming*, 44(1):127–137, May 1989.

[3] Francisco Barahona and Ali Ridha Mahjoub. On the cut polytope. *Mathematical Programming*, 36(2):157–173, 1986.

[4] Alain Billionnet and Sourour Elloumi. Using a mixed integer quadratic programming solver for the unconstrained quadratic 0-1 problem. *Math. Program.*, 109(1):55–68, 2007.

[5] Biq Mac Library – Binary quadratic and Max cut Library. `https://biqmac.aau.at/biqmaclib.html`, 2009.

[6] Thorsten Bonato, Michael Jünger, Gerhard Reinelt, and Giovanni Rinaldi. Lifting and separation procedures for the cut polytope. *Mathematical Programming: A*, 2013.

[7] Jonas Charfreitag, Michael Jünger, Sven Mallach, and Petra Mutzel. McSparse: Exact solutions of sparse maximum cut and sparse unconstrained binary quadratic optimization problems. In Cynthia A. Phillips and Bettina Speckmann, editors, *2022 Proc. of the Symp. on Algorithm Engineering and Experiments (ALENEX)*, pages 54–66. SIAM, 2022.

[8] Jonas Charfreitag, Sven Mallach, and Petra Mutzel. Integer programming for the maximum cut problem: A refined model and implications for branching. In Jonathan Berry and David B. Shmoys, editors, *Proc. of the 2023 SIAM Conf. on Applied and Computational Discrete Algorithms (ACDA23)*, pages 63–74, 2023.

[9] Caterina De Simone. The cut polytope and the boolean quadric polytope. *Discrete Mathematics*, 79(1):71–75, 1990.

[10] Samuel de Sousa, Yll Haxhimusa, and Walter G. Kropatsch. Estimation of distribution algorithm for the max-cut problem. In Walter G. Kropatsch, Nicole M. Artner, Yll Haxhimusa, and Xiaoyi Jiang, editors, *Graph-Based Representations in Pattern Recognition*, pages 244–253, Berlin, Heidelberg, 2013. Springer.

[11] Peter L. Hammer. Some network flow problems solved with pseudo-boolean programming. *Operations Research*, 13(3):388–399, 1965.

[12] Michael Jünger and Sven Mallach. Odd-cycle separation for maximum cut and binary quadratic optimization. In Michael A. Bender, Ola Svensson, and Grzegorz Herman, editors, *27th Annual European Symp. on Algorithms (ESA 2019)*, volume 144 of *Leibniz Intern. Proc. in Informatics (LIPIcs)*, pages 63:1–63:13, Dagstuhl, Germany, 2019.

[13] Michael Jünger and Sven Mallach. Exact facetial odd-cycle separation for maximum cut and binary quadratic optimization. *INFORMS J. on Computing*, 33(4):1419–1430, 2021.

[14] Richard M. Karp. Reducibility among combinatorial problems. In Raymond E. Miller and James W. Thatcher, editors, *Proceedings of a symposium on the Complexity of Computer Computations, New York*, The IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972.

[15] Frauke Liers. *Contributions to determining exact ground-states of Ising spin-glasses and to their physics*. PhD thesis, Universität zu Köln, 2004.

[16] Viet Hung Nguyen and Michel Minoux. Linear size MIP formulation of max-cut: new properties, links with cycle inequalities and computational results. *Optimization Letters*, 15(4):1041–1060, 2021.

[17] Manfred Padberg. The boolean quadric polytope: Some characteristics, facets and relatives. *Mathematical Programming*, 45(1):139–172, Aug 1989.

[18] Daniel Rehfeldt, Thorsten Koch, and Yuji Shinano. Faster exact solution of sparse maxcut and qubo problems. *Mathematical Programming Computation*, Apr 2023.

[19] Xiaojie Zhang and Paul H. Siegel. Adaptive cut generation algorithm for improved linear programming decoding of binary linear codes. *IEEE Transactions on Information Theory*, 58(10):6581–6594, 2012.