Structural Analysis of Branch-and-Cut and the Learnability of Gomory Mixed Integer Cuts

Maria-Florina Balcan
School of Computer Science
Carnegie Mellon University
ninamf@cs.cmu.edu

Siddharth Prasad
Computer Science Department
Carnegie Mellon University
sprasad2@cs.cmu.edu

Tuomas Sandholm
Computer Science Department
Carnegie Mellon University
Optimized Markets, Inc.
Strategic Machine, Inc.
Strategy Robot, Inc.
sandholm@cs.cmu.edu

Ellen Vitercik
Management Science and Engineering Department
Computer Science Department
Stanford University
vitercik@stanford.edu

Abstract

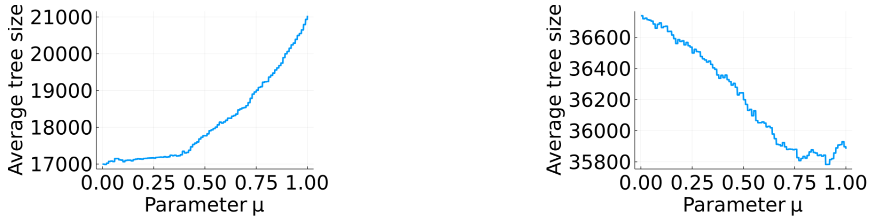
The incorporation of cutting planes within the branch-and-bound algorithm, known as branch-and-cut, forms the backbone of modern integer programming solvers. These solvers are the foremost method for solving discrete optimization problems and have a vast array of applications in machine learning, operations research, and many other fields. Choosing cutting planes effectively is a major research topic in the theory and practice of integer programming. We conduct a novel structural analysis of branch-and-cut that pins down how every step of the algorithm is affected by changes in the parameters defining the cutting planes added to an integer program. Our main application of this analysis is to derive sample complexity guarantees for using machine learning to determine which cutting planes to apply during branch-and-cut. These guarantees apply to infinite families of cutting planes, such as the family of Gomory mixed integer cuts, which are responsible for the main breakthrough speedups of integer programming solvers. We exploit geometric and combinatorial structure of branch-and-cut in our analysis, which provides a key missing piece for the recent generalization theory of branch-and-cut.

1 Introduction

Integer programming (IP) solvers are the most widely-used tools for solving discrete optimization problems. They have many applications in machine learning, operations research, and other fields, including MAP inference [34], combinatorial auctions [52], NLP [37], neural network verification [17], interpretable classification [61], training of optimal decision trees [13], and optimal clustering [48].

Under the hood, IP solvers use the tree-search algorithm branch-and-bound [41] augmented with *cutting planes*, known as *branch-and-cut* (B&C). A cutting plane is a linear constraint that is added to the linear programming (LP) relaxation at any node of the search tree. With a carefully selected cut, the LP guidance can more efficiently lead B&C to the optimal integral solution. Cutting planes, specifically the family of *Gomory mixed integer cuts* (GMI) which we study in this paper, are responsible for breakthrough speedups of modern IP solvers [15, 21].

36th Conference on Neural Information Processing Systems (NeurIPS 2022).



(a) Facility location with 40 locations and 40 clients; sampled by perturbing a base facility location IP.

(b) Facility location with 80 locations, 80 clients, and random Euclidean distance costs.

Figure 1: These figures illustrate the need for distribution-dependent policies for choosing cuts. We plot the average number of nodes B&C expands as a function of a parameter μ that controls a policy to add GMI cuts, detailed in Appendix A. In each figure, we draw a training set of facility location IPs from two different distributions. In Figure 1a, we define the distribution by starting with a uniformly random facility location instance and perturbing its costs. In Figure 1b, the costs are more structured: the facilities are located along a line and the clients have uniformly random locations. In Figure 1a, a smaller value of μ leads to small search trees, but in Figure 1b, a larger value of μ is preferable.

Successfully employing cutting planes can be challenging because there are infinitely many cuts to choose from and there are still many open questions about which cuts to employ when. A growing body of research has studied the use of machine learning for tuning various aspects of IP solvers [e.g., 2, 8, 12, 24, 29, 33, 35, 36, 38, 40, 42, 44, 45, 51, 52, 57–60], recently including cut selection [10, 11, 30, 53, 55]. We analyze a machine learning setting where there is an unknown distribution over IPs—for example, a distribution over a shipping company's routing problems. The learner receives a *training set* of IPs sampled from this distribution which it uses to learn cut parameters with strong average performance over the training set (leading, for example, to small search trees). Figure 1 illustrates that tuning cut parameters according to the instance distribution at hand can have a large impact on B&C's performance, and that for one distribution, the best parameters can be very different—in fact opposite—than the best parameters for another distribution.

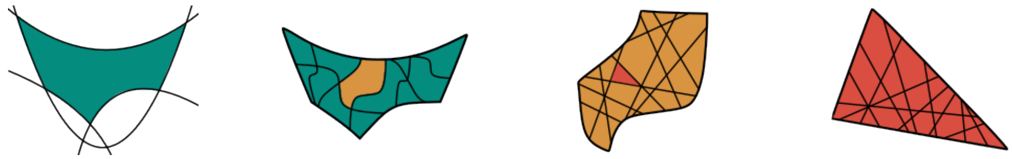
We provide *sample complexity bounds* for this procedure, which bound the number of training instances sufficient to ensure that if a set of cut parameters leads to strong average performance over the training set, it will also lead to strong expected performance on future IPs from the same distribution. These guarantees apply *no matter* what procedure is used to optimize the cut parameters over the training set—optimal or suboptimal, automated or manual.

A significant body of research has recently provided sample complexity bounds for automated algorithm configuration, further illustrating the importance of this line of research [e.g., 5–7, 9–11, 16, 25, 28]. However, these works have been unable to analyze Gomory mixed integer (GMI) cuts [26], which are perhaps the most important family of cutting planes in integer programming. They dominate most other families of cutting planes [23] and are directly responsible for the realization that a B&C framework is necessary for the speeds now achievable by modern IP solvers [4]. Prior research has been unable to handle GMI cuts because there are an uncountably infinite number of different GMI cuts that one could add, whereas prior research on cutting planes was only able to handle cutting plane families of finite effective size [10, 11]. The current work closes this gap.

The key challenge is that an infinitesimal change to any GMI cut can completely change the entire course of B&C because a cut added at the root remains in the LP relaxations stored in each node all the way to the leaves. At its core, our analysis therefore involves understanding an intricate interplay between the continuous and discrete components of our problem. The first, continuous component requires us to characterize how an LP's solution changes as a function of its constraints. The optimum will move continuously through space until it jumps from one vertex of the polytope to another. We use this characterization to analyze how the B&C tree—a discrete, combinatorial object—varies as a function of its LP guidance, which allows us to prove our sample complexity bound.

1.1 Our contributions

In order to prove our sample complexity bound for GMI cuts, we analyze how the B&C tree varies as a function of the cut parameters on any IP. We prove that the set of all possible cuts can be partitioned



(a) LP optimum closed form (Lemma 4.1)

(b) Invariant branching (Lemma 4.2)

(c) Invariant LP integrality (Lemma 4.3)

(d) Invariant B&C execution (Theorem 4.4)

Figure 2: Our B&C analysis involves successive refinements to our partition of the parameter space.

into a finite number of regions such that within any one region, B&C builds the exact same search tree. Moreover, the boundaries between regions are defined by low-degree polynomials. The simplicity of this function allows us to prove our sample complexity bound. The buildup to this result consists of three main contributions, each of which we believe may be of independent interest:

1. Our first main contribution (Section 3) addresses a fundamental question in linear programming: how does an LP's solution change when new constraints are added? As the constraints vary, the solution will jump from vertex to vertex of the LP polytope. We prove that one can partition the set of all possible constraint vectors into a finite number of regions such that within any one region, the LP's solution has a clean closed form. Moreover, we prove that the boundaries defining this partition have a specific form, defined by degree-2 polynomials.
2. We build on this result in our second main contribution (Section 4): a novel analysis of how the entire B&C search tree changes as a function of the cuts added at the root. At a high level, B&C builds this search tree by iteratively subdividing its feasible set via a process called *branching* on variables: in one subdivision, the constraint $x_i \leq k$ is enforced and in the other $x_i \geq k+1$, for some variable i and integer k (k is chosen according to the LP relaxation solution, as described in Section 2). Upon constructing a subdivision S, it checks whether the LP relaxation restricted to S is integral. If not, it further subdivides S, unless the LP relaxation's solution is worse than the best integral solution found thus far, in which case it stops searching in S—a process called *fathoming* or *pruning* S. Each subdivision is stored as a node in B&C's search tree. Our analysis of how the B&C search tree changes as a function of the cuts added has four steps, illustrated by Figure 2:
   (a) Section 4.1: First, we use our result from Section 3 to show that the cut parameter space can be partitioned into regions such that in any one region, the LP optimal solution at any node of the B&C search tree has a clean closed form, as illustrated in Figure 2a.
   (b) Section 4.2: We use this result to show that each region can be further partitioned (as illustrated in Figure 2b) such that no matter what cut we employ in any one region, all of the branching decisions that B&C makes are fixed. Intuitively, this is because the branching decisions depend on the LP relaxation, which has a closed-form solution in any one region.
   (c) Section 4.3: Next, we show that each region from Figure 2b can be further partitioned into regions (illustrated in Figure 2c) where in any one region, for every node in the B&C tree, the integrality of that node's LP relaxation is invariant no matter what cut in that region we use.
   (d) Section 4.4: Finally, we show that each of these regions can be further subdivided into regions (as in Figure 2d) where the nodes that B&C fathoms are fixed, so the tree it builds is fixed.
3. This result allows us to prove sample complexity bounds for learning high-performing cutting planes from the class of GMI cuts, our third main contribution (Section 5). Our key technical insight is that the GMI cutting plane coefficients can be viewed as a mapping that embeds our polynomial partition from the previous step (Figure 2) into the space of GMI cut parameters. We prove that the resulting embedding does not distort the polynomial hypersurfaces too much: the embedded hypersurfaces are still polynomial, with only slightly larger degree.

1.2 Related research

Learning to cut. Several papers have studied how to use machine learning for cut selection from an applied perspective [30, 53, 55], whereas our goal is to provide theoretical guarantees. Towards this end, this paper helps develop a theory of generalization for cutting plane selection. This line of inquiry began with a paper by Balcan et al. [10], who studied Chvátal-Gomory (CG) cuts [18, 27] for (pure) integer programs (IPs). Later work [11] provided a unifying sample-complexity analysis of tunable tree-search algorithms when there is a finite set of actions the algorithm can take at any given node. All prior research on generalization guarantees for integer programming [6, 10] fits this framework. In the context of single-variable branching [6], the number of possible branching decisions at any node is equal to the number of variables. In the context of CG cuts, Balcan et al. [10] showed that there are only finitely many distinct cuts at any node. These analyses followed by making pairwise comparisons between actions and understanding in what region of the parameter space one action would be chosen over another. Since there were only a finite number of actions, there were only a finite number of pairwise comparisons. This approach cannot work in our setting due to the uncountably infinite number of GMI cuts. Tackling a continuum of cutting planes requires novel techniques that we develop in this paper—in particular a structural analysis of B&C that is significantly more involved than the finite-action setting.

Sensitivity analysis of IPs and LPs. A related line of research studied the *sensitivity* of LPs, and to a lesser extent IPs, to changes in their parameters [e.g., 20, 43, 46]. This paper fits in to this line of research as we study how the solution to an LP varies as new rows are added.

2 Notation and branch-and-cut background

Integer and linear programs. An *integer program* (IP) is defined by an objective vector $\boldsymbol{c} \in \mathbb{R}^n$, a constraint matrix $A \in \mathbb{Z}^{m \times n}$, and a constraint vector $\boldsymbol{b} \in \mathbb{Z}^m$, with the form

$$\max\{\boldsymbol{c}^T\boldsymbol{x} : A\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}, \boldsymbol{x} \in \mathbb{Z}^n\}. \tag{1}$$

The *linear programming (LP) relaxation* is formed by removing the integrality constraints: $\max\{\boldsymbol{c}^T\boldsymbol{x} : A\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}\}$. We denote the optimal solution to (1) by $\boldsymbol{x}^*_{\mathsf{IP}}$ and its LP-relaxation optimal solution by $\boldsymbol{x}^*_{\mathsf{LP}}$. Let $z^*_{\mathsf{LP}} = \boldsymbol{c}^T\boldsymbol{x}^*_{\mathsf{LP}}$. If σ is a set of constraints, we let $\boldsymbol{x}^*_{\mathsf{IP}}(\sigma)$ denote the optimum of (1) subject to these additional constraints (similarly define $z^*_{\mathsf{LP}}(\sigma)$ and $\boldsymbol{x}^*_{\mathsf{LP}}(\sigma)$).

Polyhedra and polytopes. A set $\mathcal{P} \subseteq \mathbb{R}^n$ is a *polyhedron* if there exists an integer m, $A \in \mathbb{R}^{m \times n}$, and $\boldsymbol{b} \in \mathbb{R}^m$ such that $\mathcal{P} = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} \leq \boldsymbol{b}\}$. $\mathcal{P}$ is a *rational polyhedron* if there exists $A \in \mathbb{Z}^{m \times n}$ and $\boldsymbol{b} \in \mathbb{Z}^m$ such that $\mathcal{P} = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} \leq \boldsymbol{b}\}$. A bounded polyhedron is called a *polytope*. The feasible regions of all IPs considered in this paper are assumed to be rational polytopes [1] of full dimension. Let $\mathcal{P} = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^i\boldsymbol{x} \leq b_i, i \in M\}$ be a nonempty polyhedron. We assume the representation of $\mathcal{P}$ is *irredundant*, that is, $\{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^i\boldsymbol{x} \leq b_i, i \in M \setminus \{j\}\} \neq \mathcal{P}$ for all $j \in M$. For any $I \subseteq M$, the set $F_I := \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^i\boldsymbol{x} = b_i, i \in I, \boldsymbol{a}^i\boldsymbol{x} \leq b_i, i \in M \setminus I\}$ is a *face* of $\mathcal{P}$. Conversely, if F is a nonempty face of $\mathcal{P}$, then $F = F_I$ for some $I \subseteq M$. Faces of dimension 1 are called *edges* and faces of dimension 0 are called *vertices*. A detailed reference on the polyhedral theory used in our arguments can be found in Conforti et al. [19].

Given a set of constraints σ, let $\mathcal{P}(\sigma)$ denote the polyhedron that is the intersection of $\mathcal{P}$ with all inequalities in σ.

Cutting planes. A *cutting plane* is a constraint $\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta$. Let $\mathcal{P}$ be the feasible region of the LP relaxation of (1) and $\mathcal{P}_{\mathsf{I}} = \mathcal{P} \cap \mathbb{Z}^n$ be the IP's feasible set. A cut is *valid* if it is satisfied by every integer point in $\mathcal{P}_{\mathsf{I}}$: $\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta$ for all $\boldsymbol{x} \in \mathcal{P}_{\mathsf{I}}$. A valid cut *separates* a point $\boldsymbol{x} \in \mathcal{P} \setminus \mathcal{P}_{\mathsf{I}}$ if $\boldsymbol{\alpha}^T\boldsymbol{x} > \beta$. We refer to a cut both by its parameters $(\boldsymbol{\alpha}, \beta) \in \mathbb{R}^{n+1}$ and the halfspace $\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta$ in $\mathbb{R}^n$.

An important family of valid cuts that we study in this paper is the set of *Gomory mixed integer (GMI) cuts*. For decades, general-purpose cutting planes were thought to be unwieldy and useless for solving IPs quickly in practice. However, a seminal paper by Balas et al. [4] completely reversed this sentiment by showing that GMI cuts added throughout the B&C tree led to massive speedups. Today, GMI cuts are one of the most important components of state-of-the-art IP solvers.

Definition 2.1 (Gomory mixed integer cut)**.** Suppose the feasible region of the IP is in equality form $A\boldsymbol{x} = \boldsymbol{b}$, $\boldsymbol{x} \geq \boldsymbol{0}$ (which can be achieved by adding slack variables). For $\boldsymbol{u} \in \mathbb{R}^m$, let f_i

[1] This assumption is not a restrictive one. The Minkowski-Weyl theorem states that any polyhedron can be decomposed as the sum of a polytope and its recession cone. All results in this paper can be derived for rational polyhedra by considering the corresponding polytope in the Minkowski-Weyl decomposition.

denote the fractional part of $(\boldsymbol{u}^T A)_i$ and let f_0 denote the fractional part of $\boldsymbol{u}^T\boldsymbol{b}$. That is, $(\boldsymbol{u}^T A)_i = (\lfloor \boldsymbol{u}^T A \rfloor)_i + f_i$ and $\boldsymbol{u}^T\boldsymbol{b} = \lfloor \boldsymbol{u}^T\boldsymbol{b} \rfloor + f_0$. The *Gomory mixed integer (GMI) cut* parameterized by $\boldsymbol{u}$ is

$$\sum_{i:f_i\le f_0} f_i x_i + \frac{f_0}{1-f_0}\sum_{i:f_i>f_0}(1-f_i)x_i \ge f_0.$$

The form of the GMI cut is obtained via a slightly more nuanced rounding procedure than the one used to obtain the CG cut $\lfloor \boldsymbol{u}^T A\rfloor \boldsymbol{x} \le \lfloor \boldsymbol{u}^T\boldsymbol{b}\rfloor$. GMI cuts strictly dominate CG cuts. More details about GMI cuts can be found in the tutorial by Cornuéjols [22].

Branch-and-cut. We provide a high-level overview of B&C (Nemhauser and Wolsey [49], for example, provide more details). Given an IP, B&C searches the IP's feasible region by building a binary search tree. B&C solves the LP relaxation of the input IP and then adds any number of cutting planes. It stores this information at the tree's root. Let $\boldsymbol{x}^*_{\mathsf{LP}} = (\boldsymbol{x}^*_{\mathsf{LP}}[1], \dots, \boldsymbol{x}^*_{\mathsf{LP}}[n])$ be the solution to the LP relaxation with the addition of the cutting planes. B&C next uses a *variable selection policy* to choose a variable x_i to branch on. This means that it splits the IP's feasible region in two: one set where $x_i \le \lfloor \boldsymbol{x}^*_{\mathsf{LP}}[i]\rfloor$ and the other where $x_i \ge \lceil \boldsymbol{x}^*_{\mathsf{LP}}[i]\rceil$. The left child of the root now corresponds to the IP with a feasible region defined by the first subset and the right child likewise corresponds to the second subset. B&C then chooses a leaf using a *node selection policy* and recurses, adding any number of cutting planes, branching on a variable, and so on. B&C *fathoms* a node—which means that it will never branch on that node—if 1) the LP relaxation at the node is infeasible, 2) the optimal solution to the LP relaxation is integral, or 3) the optimal solution to the LP relaxation is no better than the best integral solution found thus far. Eventually, B&C will fathom every leaf, at which point it has found the globally optimal integral solution. We assume there is a bound κ on the size of the tree we allow B&C to build before we terminate, as is common in prior research [6, 10, 11, 31, 38, 39].

Every step of B&C—including node and variable selection and the choice of whether or not to fathom—depends crucially on guidance from LP relaxations. Tighter LP relaxations provide more valuable LP guidance, highlighting the importance of cuts. To give an example, this is true of the *product scoring rule* [1], a popular variable selection policy that our results apply to.

Definition 2.2. Let $\boldsymbol{x}^*_{\mathsf{LP}}$ be the solution to the LP relaxation at a node and $z^*_{\mathsf{LP}} = \boldsymbol{c}^T\boldsymbol{x}^*_{\mathsf{LP}}$. The *product scoring rule* branches on the variable $i \in [n]$ that maximizes: $\max\{z^*_{\mathsf{LP}} - z^*_{\mathsf{LP}}(x_i \le \lfloor \boldsymbol{x}^*_{\mathsf{LP}}[i]\rfloor), 10^{-6}\} \cdot \max\{z^*_{\mathsf{LP}} - z^*_{\mathsf{LP}}(x_i \ge \lceil \boldsymbol{x}^*_{\mathsf{LP}}[i]\rceil), 10^{-6}\}$.

Polynomial arrangements in Euclidean space. Let $p \in \mathbb{R}[y_1, \dots, y_k]$ be a polynomial of degree at most d. The polynomial p partitions $\mathbb{R}^k$ into connected components that belong to either $\mathbb{R}^k \setminus \{(y_1,\dots,y_k) : p(y_1,\dots,y_k) = 0\}$ or $\{(y_1,\dots,y_k) : p(y_1,\dots,y_k) = 0\}$. When we discuss the connected components of $\mathbb{R}^k$ induced by p, we include connected components in both these sets. We make this distinction because previous work on sample complexity for data-driven algorithm design oftentimes only needed to consider the connected components of the former set. The number of connected components in both sets is $O(d^k)$ [47, 54, 56].

3 Linear programming sensitivity

Our main result in this section addresses a fundamental question in linear programming: how is an LP's optimal solution affected by the addition of new constraints? Later in this paper, we use this result to prove sample complexity bounds for optimizing over the canonical family of GMI cuts.

More formally, fixing an LP with m constraints and n variables, if $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta) \in \mathbb{R}^n$ denotes the new LP optimum when the constraint $\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta$ is added, we pin down a precise characterization of $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta)$ as a function of $\boldsymbol{\alpha}$ and β. We show that $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta)$ has a piece-wise closed form: there are surfaces partitioning $\mathbb{R}^{n+1}$ such that within each connected component induced by these surfaces, $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta)$ has a closed form. While the geometric intuition used to establish this piece-wise structure relies on the basic property that optimal solutions to LPs are achieved at vertices, the surfaces defining the regions are perhaps surprisingly nonlinear: they are defined by multivariate degree-2 polynomials in $\boldsymbol{\alpha}, \beta$. In Appendix B.1 we illustrate these surfaces for an example LP.

The proof requires us to: (1) track the set of edges of the LP polytope intersected by the new constraint, and once those edges are fixed, (2) track which edge yields the vertex with the highest objective.

Let $M = [m]$ denote the set of m constraints. For $E \subseteq M$, let $A_E \in \mathbb{R}^{|E| \times n}$ and $\boldsymbol{b}_E \in \mathbb{R}^{|E|}$ denote the restrictions of A and $\boldsymbol{b}$ to E. For $\boldsymbol{\alpha} \in \mathbb{R}^n$, $\beta \in \mathbb{R}$, and $E \subseteq M$ with $|E| = n - 1$, let $A_{E,\boldsymbol{\alpha}} \in \mathbb{R}^{n \times n}$ denote the matrix obtained by adding row vector $\boldsymbol{\alpha}$ to A_E and let $A^i_{E,\boldsymbol{\alpha},\beta} \in \mathbb{R}^{n \times n}$ be the matrix $A_{E,\boldsymbol{\alpha}}$ with the ith column replaced by $(\boldsymbol{b}_E, \beta)^T$.

Theorem 3.1. *Let $(\boldsymbol{c}, A, \boldsymbol{b})$ be an LP with optimal solution $\boldsymbol{x}^*_{\mathsf{LP}}$. There are at most m^n hyperplanes and m^{2n} degree-2 polynomial hypersurfaces partitioning $\mathbb{R}^{n+1}$ into connected components such that for each component C, either: (1) $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta) = \boldsymbol{x}^*_{\mathsf{LP}}$, or (2) there is a set of constraints $E \subseteq M$ with $|E| = n - 1$ such that $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta)[i] = \det(A^i_{E,\boldsymbol{\alpha},\beta}) / \det(A_{E,\boldsymbol{\alpha}})$ for all $(\boldsymbol{\alpha}, \beta) \in C$.*

Proof. First, if $\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta$ does not separate $\boldsymbol{x}^*_{\mathsf{LP}}$, then $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta) = \boldsymbol{x}^*_{\mathsf{LP}}$. The set of all such cuts is the halfspace given by $\{(\boldsymbol{\alpha}, \beta) \in \mathbb{R}^{n+1} : \boldsymbol{\alpha}^T\boldsymbol{x}^*_{\mathsf{LP}} \le \beta\}$. All other cuts separate $\boldsymbol{x}^*_{\mathsf{LP}}$ and thus pass through $\mathcal{P} = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} \le \boldsymbol{b}, \boldsymbol{x} \ge \boldsymbol{0}\}$, and the new LP optimum is achieved at a vertex created by the cut. We consider the new vertices formed by the cut, which lie on edges of $\mathcal{P}$. Each edge e of $\mathcal{P}$ can be identified with a subset $E \subset M$ of size $n - 1$ such that the edge is the set of all points $\boldsymbol{x}$ such that $\boldsymbol{a}_i^T\boldsymbol{x} = b_i$ for all $i \in E$ and $\boldsymbol{a}_i^T\boldsymbol{x} \le b_i$ for all $i \in M \setminus E$ where $\boldsymbol{a}_i$ is the ith row of A. If we drop the inequality constraints defining the edge, the equality constraints define a line in $\mathbb{R}^n$. The intersection of the cut $\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta$ and this line is the solution to the system of n linear equations in n variables: $A_E\boldsymbol{x} = \boldsymbol{b}_E, \boldsymbol{\alpha}^T\boldsymbol{x} = \beta$. By Cramer's rule, the unique solution $\boldsymbol{x} = (x_1, \ldots, x_n)$ to this system is given by $x_i = \det(A^i_{E,\boldsymbol{\alpha},\beta}) / \det(A_{E,\boldsymbol{\alpha}})$. To ensure that the intersection point lies on the edge of the polytope, we stipulate that it satisfies the inequality constraints in $M \setminus E$. That is,

$$\sum_{j=1}^{n} a_{ij} \cdot \frac{\det(A^j_{E,\boldsymbol{\alpha},\beta})}{\det(A_{E,\boldsymbol{\alpha}})} \le b_i \tag{2}$$

for every $i \in M \setminus E$ (if $\boldsymbol{\alpha}, \beta$ satisfy any of these constraints, it must be that $\det(A_{E,\boldsymbol{\alpha}}) \ne 0$, which guarantees that $A_E\boldsymbol{x} = \boldsymbol{b}_E, \boldsymbol{\alpha}^T\boldsymbol{x} = \beta$ has a unique solution). Multiplying through by $\det(A_{E,\boldsymbol{\alpha}})$ shows that this constraint is a halfspace in $\mathbb{R}^{n+1}$, since $\det(A_{E,\boldsymbol{\alpha}})$ and $\det(A^i_{E,\boldsymbol{\alpha},\beta})$ are linear in $\boldsymbol{\alpha}$ and β. The collection of all the hyperplanes defining the boundaries of these halfspaces over all edges of $\mathcal{P}$ induces a partition of $\mathbb{R}^{n+1}$ into connected components such that for all $(\boldsymbol{\alpha}, \beta)$ within a given component, the (nonempty) set of edges of $\mathcal{P}$ that the hyperplane $\boldsymbol{\alpha}^T\boldsymbol{x} = \beta$ intersects is invariant.

Now, consider a single connected component, denoted by C for brevity. Let $e_1, \ldots, e_k$ denote the edges intersected by cuts in C, and let $E_1, \ldots, E_k \subset M$ denote the sets of constraints that are binding at each of these edges, respectively. For each pair e_p, e_q, consider the surface

$$\sum_{i=1}^{n} c_i \cdot \frac{\det(A^i_{E_p,\boldsymbol{\alpha},\beta})}{\det(A_{E_p,\boldsymbol{\alpha}})} = \sum_{i=1}^{n} c_i \cdot \frac{\det(A^i_{E_q,\boldsymbol{\alpha},\beta})}{\det(A_{E_q,\boldsymbol{\alpha}})}. \tag{3}$$

Clearing the (nonzero) denominators shows this is a degree-2 polynomial hypersurface in $\boldsymbol{\alpha}, \beta$ in $\mathbb{R}^{n+1}$. This hypersurface is the set of all $(\boldsymbol{\alpha}, \beta)$ for which the LP objective values achieved at the vertices on edges e_p and e_q are equal. The collection of these surfaces for each p, q partitions C into further connected components. Within each component C', the edge containing the vertex that maximizes the objective is invariant. If this edge corresponds to binding constraints E, $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta)$ has the closed form $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta)[i] = \det(A^i_{E,\boldsymbol{\alpha},\beta}) / \det(A_{E,\boldsymbol{\alpha}})$ for all $(\boldsymbol{\alpha}, \beta) \in C'$. We now count the number of surfaces in our decomposition. $\mathcal{P}$ has at most $\binom{m}{n-1} \le m^{n-1}$ edges, and for each edge E, Equation (2) defines at most $|M \setminus E| \le m$ hyperplanes for a total of at most m^n hyperplanes. Equation (3) defines a degree-2 polynomial hypersurface for every pair of edges, of which there are at most $\binom{m^n}{2} \le m^{2n}$. $\square$

In Appendix B.2, we generalize Theorem 3.1 to understand $\boldsymbol{x}^*_{\mathsf{LP}}$ as a function of any K constraints. In this case, we show that the piecewise structure is given by degree-$2K$ multivariate polynomials.

4 Structure and sensitivity of branch-and-cut

We now use Theorem 3.1 to answer a fundamental question about B&C: what is the structure of the B&C tree as a function of cuts at the root? Answering this question brings us one step closer toward providing sample complexity guarantees for GMI cuts. Said another way, we derive conditions on $\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2 \in \mathbb{R}^n$, $\beta_1, \beta_2 \in \mathbb{R}$, such that B&C behaves identically on the two IPs

$$\max\{\boldsymbol{c}^T\boldsymbol{x} : A\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \boldsymbol{x} \in \mathbb{Z}^n_{\geq 0}\} \text{ and } \max\{\boldsymbol{c}^T\boldsymbol{x} : A\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{\alpha}_2^T\boldsymbol{x} \leq \beta_2, \boldsymbol{x} \in \mathbb{Z}^n_{\geq 0}\}.$$

We prove that the set of all cuts can be partitioned into a finite number of regions where by employing cuts from any one region, the B&C tree remains exactly the same. We also prove that the boundaries between regions are defined by low-degree polynomials. Figure 2 is a schematic diagram of our proof, which breaks the analysis of B&C into four main steps. Each step successively refines the partition obtained in the previous step, and uses the properties established in the previous step to analyze the next stage of B&C. We focus on a single cut added to the root and extend to multiple cuts in Appendix C.2. The full proofs from this section are in Appendix C.

We use the following notation in this section. Given an IP, let $\tau = \lceil \max_{\boldsymbol{x}\in\mathcal{P}} \|\boldsymbol{x}\|_\infty \rceil$ be the maximum magnitude coordinate of any LP-feasible solution, rounded up. By Cramer's rule and Hadamard's inequality, $\tau \leq a^n n^{n/2}$ where $a = \|A\|_{\infty,\infty}$. However, τ can be much smaller. For example, if A contains a row with only positive entries, then $\tau \leq \|\boldsymbol{b}\|_\infty$. Let $\mathcal{BC} := \{\boldsymbol{x}[i] \leq \ell, \boldsymbol{x}[i] \geq \ell\}_{0 \leq \ell \leq \tau, i \in [n]}$, which contains the set of all possible branching constraints. Let A_σ and $\boldsymbol{b}_\sigma$ denote A and b with the constraints in $\sigma \subseteq \mathcal{BC}$ added. For $E \subseteq M \cup \sigma$, let $A_{E,\sigma} \in \mathbb{R}^{|E|\times n}$ and $\boldsymbol{b}_E \in \mathbb{R}^{|E|}$ denote the restrictions of A_σ and $\boldsymbol{b}_\sigma$ to E. For $\boldsymbol{\alpha} \in \mathbb{R}^n$, $\beta \in \mathbb{R}$ and $E \subseteq M \cup \sigma$ with $|E| = n-1$, let $A_{E,\boldsymbol{\alpha},\sigma} \in \mathbb{R}^{n\times n}$ denote the matrix obtained by adding row vector $\boldsymbol{\alpha}$ to $A_{E,\sigma}$ and let $A^i_{E,\boldsymbol{\alpha},\beta,\sigma} \in \mathbb{R}^{n\times n}$ be the matrix $A_{E,\boldsymbol{\alpha},\sigma}$ with the ith column replaced by $(\boldsymbol{b}_{E,\sigma}, \beta)^T$.

4.1 Step 1: Understanding how the cut affects the LP optimum at any node of the B&C tree

Theorem 3.1 gives a (piecewise) closed form for the LP optimum $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta)$ at the root of the B&C tree as a function of coefficients $(\boldsymbol{\alpha}, \beta) \in \mathbb{R}^{n+1}$ determining the cut. The first step is to extend this result to get a handle on the LP optimum at any node of any B&C tree. Suppose $\sigma \subseteq \mathcal{BC}$ is a set of branching constraints (any node of any B&C tree can be identified with some $\sigma \subseteq \mathcal{BC}$). We refine the partition of space obtained in Theorem 3.1 so that within a given region of the new partition, $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)$ has a closed form for all σ. This is illustrated by Figure 2a.

Lemma 4.1. *For any IP $(\boldsymbol{c}, A, \boldsymbol{b})$, there are at most $(m+2n)^n\tau^{3n}$ hyperplanes and at most $(m+2n)^{2n}\tau^{3n}$ degree-2 polynomial hypersurfaces partitioning $\mathbb{R}^{n+1}$ into connected components such that for each component C and every $\sigma \subset \mathcal{BC}$, either: (1) $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma) = \boldsymbol{x}^*_{\mathsf{LP}}(\sigma)$ and $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma) = z^*_{\mathsf{LP}}(\sigma)$, or (2) there is a set of constraints $E \subseteq M \cup \sigma$ with $|E| = n-1$ such that $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)[i] = \frac{\det(A^i_{E,\boldsymbol{\alpha},\beta,\sigma})}{\det(A_{E,\boldsymbol{\alpha},\sigma})}$ for all $(\boldsymbol{\alpha}, \beta) \in C$.*

4.2 Step 2: Conditions for branching decisions to be identical

We next refine the decomposition obtained in Lemma 4.1 so that the branching constraints added at each step of B&C are invariant within a region, as in Figure 2b. For concreteness, we analyze the product scoring rule (Def. 2.2) used by the leading open-source solver SCIP [14]. The high-level intuition is that we zoom in on a connected component in the partition of Lemma 4.1. Within this component, we may express $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)$ explicitly in terms of $\boldsymbol{\alpha}, \beta$, for all σ. This allows us to unravel the branching rule and derive conditions for invariance.

Lemma 4.2. *For any IP $(\boldsymbol{c}, A, \boldsymbol{b})$, there are at most $3(m+2n)^n\tau^{3n}$ hyperplanes, $3(m+2n)^{3n}\tau^{4n}$ degree-2 polynomial hypersurfaces, and $(m+2n)^{6n}\tau^{4n}$ degree-5 polynomial hypersurfaces partitioning $\mathbb{R}^{n+1}$ into connected components such that within each component, the branching constraints used at every step of B&C are invariant.*

Proof sketch. If we are at a node of B&C represented by σ, the new branching constraints after expanding that node are of the form $x_i \leq \lfloor \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)[i] \rfloor$ and $x_i \geq \lceil \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)[i] \rceil$. Lemma 4.1 gives closed forms for the right-hand-sides of these two constraints, allowing us to control the rounding aspect of the constraints. The rest of the proof is a careful analysis of the product scoring rule which allows us to derive conditions ensuring that the branching variable is invariant. ☐

4.3 Step 3: When do nodes have an integral LP optimum?

We now move to the most critical phase of branch-and-cut: deciding when to fathom a node. The first reason a node might be fathomed is if the LP relaxation of the IP at that node has an integral solution. We derive conditions that ensure that nearby cuts have the same effect on the integrality of the IP at any node in the search tree. Recall $\mathcal{P}_\mathsf{I} = \mathcal{P} \cap \mathbb{Z}^n$ is the set of integer points in $\mathcal{P}$.

Lemma 4.3. *For any IP $(\boldsymbol{c}, A, \boldsymbol{b})$, there are at most $3(m+2n)^n\tau^{4n}$ hyperplanes, $3(m+2n)^{3n}\tau^{4n}$ degree-2 polynomial hypersurfaces, and $(m+2n)^{6n}\tau^{4n}$ degree-5 polynomial hypersurfaces partitioning $\mathbb{R}^{n+1}$ into connected components such that for each component C and each $\sigma \subseteq \mathcal{BC}$, $\mathbf{1}\left[\boldsymbol{x}^*_{\mathsf{LP}}\left(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma\right) \in \mathbb{Z}^n\right]$ is invariant for all $(\boldsymbol{\alpha}, \beta) \in C$.*

Proof sketch. For all σ, $\boldsymbol{x}_\mathsf{I} \in \mathcal{P}_\mathsf{I}$, and $i \in [n]$, consider the surface $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i] = \boldsymbol{x}_\mathsf{I}[i]$. By Lemma 4.1, this surface is a hyperplane. If $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) \in \mathbb{Z}^n$ for some $(\boldsymbol{\alpha}, \beta)$ in a connected component induced by these hyperplanes, $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) = \boldsymbol{x}_\mathsf{I}$ for some $\boldsymbol{x}_\mathsf{I} \in \mathcal{P}_\mathsf{I}(\sigma) \subseteq \mathcal{P}_\mathsf{I}$, which means that $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) = \boldsymbol{x}_\mathsf{I} \in \mathbb{Z}^n$ *for all* $(\boldsymbol{\alpha}, \beta)$ in that component. □

Lemma 4.3 is illustrated by Figure 2c. Next, suppose for a moment that B&C fathoms a node if and only if either the LP is infeasible or the LP optimal solution is integral—that is, the "bounding" of B&C is suppressed. In this case, the tree built by B&C is invariant within each component of the partition in Lemma 4.3. Equipped with this observation, we now analyze the full behavior of B&C.

4.4 Step 4: Pruning nodes with weak LP bounds

In this final step, we analyze the most important aspect of B&C: pruning nodes when the LP objective value is smaller than the best-known integral solution. Using the tools we have developed so far, expressing the question "is the LP value at a node smaller than the best-known integral solution?" becomes a simple matter of hyperplanes and halfspaces. This final step is illustrated by Figure 2d.

Theorem 4.4. *Given an IP $(\boldsymbol{c}, A, \boldsymbol{b})$, there is a set of at most $O(14^n(m+2n)^{3n^2}\tau^{5n^2})$ polynomial hypersurfaces of degree ≤ 5 partitioning $\mathbb{R}^{n+1}$ into connected components such that the B&C tree built after adding the cut $\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta$ at the root is invariant over all $(\boldsymbol{\alpha}, \beta)$ within a given component.*

Proof sketch. Let $Q_1, \dots, Q_{i_1}, I_1, Q_{i_1+1}, \dots, Q_{i_2}, I_2, Q_{i_2+1}, \dots$ denote the nodes of the B&C tree in order of exploration, under the assumption that a node is pruned if and only if either the LP at that node is infeasible or the LP optimal solution is integral. Here, a node is identified by the list σ of branching constraints added to the input IP. Nodes labeled by Q are either infeasible or have fractional LP optimal solutions. Nodes labeled by I have integral LP optimal solutions and are candidates for the incumbent integral solution at the point they are encountered. By Lemma 4.3, this ordered list of nodes is invariant over any connected component of our partition.

Given an node index ℓ, let $I(\ell)$ denote the incumbent node with the highest objective value encountered up until the ℓth node searched by B&C, and let $z(I(\ell))$ denote its objective value. For each node Q_ℓ, let σ_ℓ denote the branching constraints added to arrive at node Q_ℓ. The hyperplane $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_\ell) = z(I(\ell))$ (which is a hyperplane due to Lemma 4.1) induces two regions. In one region, $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_\ell) \le z(I(\ell))$ and so the subtree rooted at Q_ℓ is pruned. In the other region, $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_\ell) > z(I(\ell))$, and Q_ℓ is branched on further. Therefore, within each component induced by all such hyperplanes for all ℓ, the set of nodes that are pruned is invariant. Combined with the surfaces established in Lemma 4.3, these hyperplanes partition $\mathbb{R}^{n+1}$ into components such that as $(\boldsymbol{\alpha}, \beta)$ varies within a given component, the B&C tree is invariant. □

5 Sample complexity bounds for Gomory mixed integer cuts

In this section, we show how the results from Section 4 can be used to provide sample complexity bounds for GMI cuts (Definition 2.1), parameterized by $\boldsymbol{u} \in \mathcal{U} \subseteq \mathbb{R}^m$. We assume there is an unknown, application-specific distribution $\mathcal{D}$ over IPs. The learner receives a *training set* $\mathcal{S} \sim \mathcal{D}^N$ of N IPs sampled from this distribution. A *sample complexity guarantee* bounds the number of samples N sufficient to ensure that for any parameter setting $\boldsymbol{u} \in \mathcal{U}$, the B&C tree size on average over $\mathcal{S}$ is close to the expected tree size. More formally, let $g_{\boldsymbol{u}}(\boldsymbol{c}, A, \boldsymbol{b})$ be the size of the tree

B&C builds given the input $(\boldsymbol{c}, A, \boldsymbol{b})$ after applying the cut defined by $\boldsymbol{u}$ at the root. Given $\epsilon > 0$ and $\delta \in (0,1)$, a sample complexity guarantee bounds the number of samples N sufficient to ensure that with probability $1-\delta$ over the draw $\mathcal{S} \sim \mathcal{D}^N$, for every parameter setting $\boldsymbol{u} \in \mathcal{U}$, $|\frac{1}{N}\sum_{(\boldsymbol{c},A,\boldsymbol{b})\in\mathcal{S}} g_{\boldsymbol{u}}(\boldsymbol{c}, A, \boldsymbol{b}) - \mathbb{E}[g_{\boldsymbol{u}}(\boldsymbol{c}, A, \boldsymbol{b})]| \le \epsilon$. To derive our sample complexity guarantee, we use the notion of *pseudo-dimension* [50]. Let $\mathcal{G} = \{g_{\boldsymbol{u}} : \boldsymbol{u} \in \mathcal{U}\}$. The *pseudo-dimension of* $\mathcal{G}$, denoted $\mathrm{Pdim}(\mathcal{G})$, is the largest integer N for which there exist N IPs $(\boldsymbol{c}_1, A_1, \boldsymbol{b}_1), \ldots, (\boldsymbol{c}_N, A_N, \boldsymbol{b}_N)$ and N thresholds $r_1, \ldots, r_N \in \mathbb{R}$ such that for every binary vector $(\sigma_1, \ldots, \sigma_N) \in \{0,1\}^N$, there exists $g_{\boldsymbol{u}} \in \mathcal{G}$ such that $g_{\boldsymbol{u}}(\boldsymbol{c}_i, A_i, \boldsymbol{b}_i) \ge r_i$ if and only if $\sigma_i = 1$. The number of samples sufficient to ensure an error of ε and confidence of $1-\delta$ is $N = O(\frac{\kappa^2}{\epsilon^2}(\mathrm{Pdim}(\mathcal{G}) + \log\frac{1}{\delta}))$ [50]. Equivalently, for a given number of samples N, the error-term ε is at most $\kappa\sqrt{(\mathrm{Pdim}(\mathcal{G}) + \log(1/\delta))/N}$.

So far, $\boldsymbol{\alpha}, \beta$ have been parameters that do not depend on the input instance $\boldsymbol{c}, A, \boldsymbol{b}$. Suppose now that they do: $\boldsymbol{\alpha}, \beta$ are functions of $\boldsymbol{c}, A, \boldsymbol{b}$ and a parameter vector $\boldsymbol{u}$ (as they are for GMI cuts). Despite the structure established in the previous section, if $\boldsymbol{\alpha}, \beta$ can depend on $(\boldsymbol{c}, A, \boldsymbol{b})$ in arbitrary ways, one cannot even hope for a finite sample complexity, illustrated by the following impossibility result. The full proofs of all results from this section are in Appendix D.

Theorem 5.1. *There exist functions $\boldsymbol{\alpha}_{\boldsymbol{c},A,\boldsymbol{b}} : \mathcal{U} \to \mathbb{R}^n$ and $\beta_{\boldsymbol{c},A,\boldsymbol{b}} : \mathcal{U} \to \mathbb{R}$ such that* $\mathrm{Pdim}(\{g_{\boldsymbol{u}} : \boldsymbol{u} \in \mathcal{U}\}) = \infty$, *where $\mathcal{U}$ is any set with $|\mathcal{U}| = |\mathbb{R}|$.*

However, in the case of GMI cuts (Def. 2.1), we show that the cutting plane coefficients parameterized by $\boldsymbol{u}$ are highly structured. Combining this structure with our analysis of B&C allows us to derive polynomial sample complexity bounds. We assume that $\boldsymbol{u} \in [-U, U]^m$ for some $U > 0$.

Let $\boldsymbol{\alpha} : [-U, U]^m \to \mathbb{R}^n$ denote the function taking GMI cut parameters $\boldsymbol{u}$ to the corresponding vector of coefficients determining the resulting cutting plane, and let $\beta : [-U, U]^m \to \mathbb{R}$ denote the offset of the resulting cutting plane. So (after multiplying through by $1 - f_0$),

$$\boldsymbol{\alpha}(\boldsymbol{u})[i] = \begin{cases} f_i(1-f_0) & \text{if } f_i \le f_0 \\ f_0(1-f_i) & \text{if } f_i > f_0 \end{cases}$$

and $\beta(\boldsymbol{u}) = f_0(1-f_0)$ (f_0 and f_i are functions of $\boldsymbol{u}$, but we suppress this dependence for readability).

To understand the structure of B&C as a function of GMI cut parameters, we study the preimages of components $C \subseteq \mathbb{R}^{n+1}$ under the GMI coefficient maps $\boldsymbol{\alpha} : [-U, U]^m \to \mathbb{R}^n$, $\beta : [-U, U]^m \to \mathbb{R}$. If $C \subseteq \mathbb{R}^{n+1}$ (as in Theorem 4.4) is such that B&C (as a function of $\boldsymbol{\alpha}, \beta$) is invariant over C, then B&C (as a function of GMI parameter $\boldsymbol{u}$) is invariant over $D := \{\boldsymbol{u} : (\boldsymbol{\alpha}(\boldsymbol{u}), \beta(\boldsymbol{u})) \in C\}$. Our key structural insight for GMI cuts is that if C is the intersection of degree-d polynomial hypersurfaces in $\mathbb{R}^{n+1}$, then D is the intersection of degree-$2d$ polynomial hypersurfaces in $[-U, U]^m$. We provide the high-level intuition for this result below—the formal statements and proofs are in Appendix D.

Consider some degree-d polynomial p in variables $y_1, \ldots, y_{n+1}$ that defines C, which can be written as $\sum_{T \sqsubseteq [n+1], |T| \le d} \lambda_T \prod_{i \in T} y_i$ for some coefficients $\lambda_T \in \mathbb{R}$, where $T \sqsubseteq [n+1]$ means that T is a multiset of $[n+1]$. Evaluating at $(\boldsymbol{\alpha}(\boldsymbol{u}), \beta(\boldsymbol{u}))$, we get

$$\sum_{|T| \le d} \lambda_T \prod_{i \in T \cap S \setminus \{n+1\}} f_i(1-f_0) \prod_{i \in T \setminus S \setminus \{n+1\}} f_0(1-f_i) \prod_{i \in T \cap \{n+1\}} f_0(1-f_0).$$

Next, substitute $f_i = \boldsymbol{u}^T\boldsymbol{a}_i - \lfloor \boldsymbol{u}^T\boldsymbol{a}_i \rfloor$ and $f_0 = \boldsymbol{u}^T\boldsymbol{b} - \lfloor \boldsymbol{u}^T\boldsymbol{b} \rfloor$. Restricted to $\boldsymbol{u}$ such that the floor terms round down to some fixed integers, the above expression is a polynomial in $\boldsymbol{u}$ of degree $\le 2d$. We run this procedure for every polynomial determining C, for every connected component C in the partition of $\mathbb{R}^{n+1}$ established in Theorem 4.4 to derive our main structural result for GMI cuts.

Lemma 5.2. *Consider the family of GMI cuts parameterized by $\boldsymbol{u} \in [-U, U]^m$. For any IP $(\boldsymbol{c}, A, \boldsymbol{b})$, there are at most $O(nU^2\|A\|_1\|\boldsymbol{b}\|_1)$ hyperplanes and $2^{O(n^2)}(m+2n)^{O(n^3)}\tau^{O(n^3)}$ degree-10 polynomial hypersurfaces partitioning $[-U, U]^m$ into connected components such that the B&C tree built after adding the GMI cut defined by $\boldsymbol{u}$ is invariant over all $\boldsymbol{u}$ within a single component.*

Bounding $\mathrm{Pdim}(\{g_{\boldsymbol{u}} : \boldsymbol{u} \in [-U, U]^m\})$ is a direct application of the main theorem of Balcan et al. [9] along with standard results bounding the VC dimension of polynomial boundaries [3].

Theorem 5.3. *The pseudo-dimension of the class of tree-size functions $\{g_{\boldsymbol{u}} : \boldsymbol{u} \in [-U, U]^m\}$ on the domain of IPs with $\|A\|_1 \le a$ and $\|\boldsymbol{b}\|_1 \le b$ is $O\left(m\log(abU) + mn^3\log(m+n) + mn^3\log\tau\right)$.*

We generalize the analysis of this section to multiple GMI cuts at the root of the B&C tree in Appendix D. We show that if K GMI cuts are sequentially applied at the root, the resulting partition of the parameter space is induced by polynomials of degree $O(K^2)$.

6 Conclusions

In this paper, we investigated fundamental questions about integer programming: given an integer program, what is the structure of the branch-and-cut tree as a function of a set of additional feasible constraints? Through a detailed geometric and combinatorial analysis of how additional constraints affect the LP relaxation's optimal solution, we showed that the branch-and-cut tree is piecewise constant and precisely bounded the number of pieces. We showed that the structural insights that we developed could be used to prove sample complexity bounds for learning GMI cuts, one of the most important classes of general-purpose cutting planes in integer programming.

This paper opens up a variety of directions for future research. Our sensitivity analyses in Sections 3 and 4 are fairly general and a promising direction is to explore applications to other important topics in integer programming such as column generation and lifting. Another important direction is to further develop algorithmic approaches for choosing GMI (and other) cutting planes. Currently, solvers employ a subset of GMI cuts derived from the optimal simplex tableau due to computational efficiency—it would be interesting to see if the theory developed in this paper could expand the possibilities for efficient cutting plane generation.

Acknowledgments and Disclosure of Funding

This material is based on work supported by the National Science Foundation under grants CCF-1733556, CCF-1910321, IIS-1901403, and SES-1919453, the ARO under award W911NF2010081, the Defense Advanced Research Projects Agency under cooperative agreement HR00112020003, an AWS Machine Learning Research Award, an Amazon Research Award, a Bloomberg Research Grant, and a Microsoft Research Faculty Fellowship.

References

[1] Tobias Achterberg. *Constraint Integer Programming*. PhD thesis, Technische Universität Berlin, 2007.

[2] Alejandro Marcos Alvarez, Quentin Louveaux, and Louis Wehenkel. A machine learning-based approximation of strong branching. *INFORMS Journal on Computing*, 29(1):185–195, 2017.

[3] Martin Anthony and Peter Bartlett. *Neural Network Learning: Theoretical Foundations*. Cambridge University Press, 2009.

[4] Egon Balas, Sebastian Ceria, Gérard Cornuéjols, and N Natraj. Gomory cuts revisited. *Operations Research Letters*, 19(1):1–9, 1996.

[5] Maria-Florina Balcan. Data-driven algorithm design. In Tim Roughgarden, editor, *Beyond Worst Case Analysis of Algorithms*. Cambridge University Press, 2020.

[6] Maria-Florina Balcan, Travis Dick, Tuomas Sandholm, and Ellen Vitercik. Learning to branch. In *International Conference on Machine Learning (ICML)*, 2018.

[7] Maria-Florina Balcan, Travis Dick, and Colin White. Data-driven clustering via parameterized Lloyd's families. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[8] Maria-Florina Balcan, Tuomas Sandholm, and Ellen Vitercik. Refined bounds for algorithm configuration: The knife-edge of dual class approximability. In *International Conference on Machine Learning (ICML)*, 2020.

[9] Maria-Florina Balcan, Dan DeBlasio, Travis Dick, Carl Kingsford, Tuomas Sandholm, and Ellen Vitercik. How much data is sufficient to learn high-performing algorithms? Generalization guarantees for data-driven algorithm design. In *Annual Symposium on Theory of Computing (STOC)*, 2021.

[10] Maria-Florina Balcan, Siddharth Prasad, Tuomas Sandholm, and Ellen Vitercik. Sample complexity of tree search configuration: Cutting planes and beyond. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

[11] Maria-Florina Balcan, Siddharth Prasad, Tuomas Sandholm, and Ellen Vitercik. Improved sample complexity bounds for branch-and-cut. In *International Conference on Principles and Practice of Constraint Programming (CP)*, 2022.

[12] Yoshua Bengio, Andrea Lodi, and Antoine Prouvost. Machine learning for combinatorial optimization: a methodological tour d'horizon. *European Journal of Operational Research*, 2020.

[13] Dimitris Bertsimas and Jack Dunn. Optimal classification trees. *Machine Learning*, 106(7): 1039–1082, 2017.

[14] Ksenia Bestuzheva, Mathieu Besançon, Wei-Kun Chen, Antonia Chmiela, Tim Donkiewicz, Jasper van Doornmalen, Leon Eifler, Oliver Gaul, Gerald Gamrath, Ambros Gleixner, Leona Gottwald, Christoph Graczyk, Katrin Halbig, Alexander Hoen, Christopher Hojny, Rolf van der Hulst, Thorsten Koch, Marco Lübbecke, Stephen J. Maher, Frederic Matter, Erik Mühmer, Benjamin Müller, Marc E. Pfetsch, Daniel Rehfeldt, Steffan Schlein, Franziska Schlösser, Felipe Serrano, Yuji Shinano, Boro Sofranac, Mark Turner, Stefan Vigerske, Fabian Wegscheider, Philipp Wellner, Dieter Weninger, and Jakob Witzig. The SCIP Optimization Suite 8.0. Technical report, Optimization Online, December 2021. URL `http://www.optimization-online.org/DB_HTML/2021/12/8728.html`.

[15] Robert Bixby, Mary Fenelon, Zonghao Gu, Ed Rothberg, and Roland Wunderling. MIP: Theory and practice—closing the gap. In *IFIP Conference on System Modeling and Optimization*, pages 19–49. Springer, 1999.

[16] Avrim Blum, Chen Dan, and Saeed Seddighin. Learning complexity of simulated annealing. In *International conference on artificial intelligence and statistics*, pages 1540–1548. PMLR, 2021.

[17] Rudy Bunel, Ilker Turkaslan, Philip H.S. Torr, Pushmeet Kohli, and M. Pawan Kumar. A unified view of piecewise linear neural network verification. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[18] Vašek Chvátal. Edmonds polytopes and a hierarchy of combinatorial problems. *Discrete mathematics*, 4(4):305–337, 1973.

[19] Michele Conforti, Gérard Cornuéjols, Giacomo Zambelli, et al. *Integer programming*, volume 271. Springer, 2014.

[20] William Cook, Albertus MH Gerards, Alexander Schrijver, and Éva Tardos. Sensitivity theorems in integer linear programming. *Mathematical Programming*, 34(3):251–264, 1986.

[21] Gérard Cornuéjols. Revival of the Gomory cuts in the 1990's. *Annals of Operations Research*, 149(1):63–66, 2007.

[22] Gérard Cornuéjols. Valid inequalities for mixed integer linear programs. *Mathematical Programming*, 112(1):3–44, 2008.

[23] Gérard Cornuéjols and Yanjun Li. Elementary closures for integer programs. *Operations Research Letters*, 28(1):1–8, 2001.

[24] Giovanni Di Liberto, Serdar Kadioglu, Kevin Leo, and Yuri Malitsky. Dash: Dynamic approach for switching heuristics. *European Journal of Operational Research*, 248(3):943–953, 2016.

[25] Vikas Garg and Adam Kalai. Supervising unsupervised learning. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

[26] Ralph Gomory. An algorithm for the mixed integer problem. resreport RM-2597, The Rand Corporation, 1960.

[27] Ralph E. Gomory. Outline of an algorithm for integer solutions to linear programs. *Bulletin of the American Mathematical Society*, 64(5):275 – 278, 1958.

[28] Rishi Gupta and Tim Roughgarden. A PAC approach to application-specific algorithm selection. *SIAM Journal on Computing*, 46(3):992–1017, 2017.

[29] He He, Hal Daume III, and Jason M Eisner. Learning to search in branch and bound algorithms. In *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2014.

[30] Zeren Huang, Kerong Wang, Furui Liu, Hui-Ling Zhen, Weinan Zhang, Mingxuan Yuan, Jianye Hao, Yong Yu, and Jun Wang. Learning to select cuts for efficient mixed-integer programming. *Pattern Recognition*, 123:108353, 2022.

[31] Frank Hutter, Holger H Hoos, Kevin Leyton-Brown, and Thomas Stützle. ParamILS: An automatic algorithm configuration framework. *Journal of Artificial Intelligence Research*, 36 (1):267–306, 2009. ISSN 1076-9757.

[32] Robert G Jeroslow. Trivial integer programs unsolvable by branch-and-bound. *Mathematical Programming*, 6(1):105–109, 1974.

[33] Serdar Kadioglu, Yuri Malitsky, Meinolf Sellmann, and Kevin Tierney. ISAC—instance-specific algorithm configuration. In *European Conference on Artificial Intelligence (ECAI)*, 2010.

[34] Jörg Hendrik Kappes, Markus Speth, Gerhard Reinelt, and Christoph Schnörr. Towards efficient and exact map-inference for large scale discrete computer vision problems via combinatorial optimization. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1752–1758. IEEE, 2013.

[35] Elias Khalil, Pierre Le Bodic, Le Song, George Nemhauser, and Bistra Dilkina. Learning to branch in mixed integer programming. In *AAAI Conference on Artificial Intelligence*, 2016.

[36] Elias Khalil, Bistra Dilkina, George Nemhauser, Shabbir Ahmed, and Yufen Shao. Learning to run heuristics in tree search. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

[37] Daniel Khashabi, Tushar Khot, Ashish Sabharwal, Peter Clark, Oren Etzioni, and Dan Roth. Question answering via integer programming over semi-structured knowledge. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2016.

[38] Robert Kleinberg, Kevin Leyton-Brown, and Brendan Lucier. Efficiency through procrastination: Approximately optimal algorithm configuration with runtime guarantees. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2017.

[39] Robert Kleinberg, Kevin Leyton-Brown, Brendan Lucier, and Devon Graham. Procrastinating with confidence: Near-optimal, anytime, adaptive algorithm configuration. *Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[40] Michail G Lagoudakis and Michael L Littman. Learning to select branching rules in the DPLL procedure for satisfiability. *Electronic Notes in Discrete Mathematics*, 9:344–359, 2001.

[41] Ailsa H Land and Alison G Doig. An automatic method of solving discrete programming problems. *Econometrica*, pages 497–520, 1960.

[42] Kevin Leyton-Brown, Eugene Nudelman, and Yoav Shoham. Empirical hardness models: Methodology and a case study on combinatorial auctions. *Journal of the ACM*, 56(4):1–52, 2009. ISSN 0004-5411.

[43] Wu Li. The sharp lipschitz constants for feasible and optimal solutions of a perturbed linear program. *Linear algebra and its applications*, 187:15–40, 1993.

[44] Jia Hui Liang, Vijay Ganesh, Pascal Poupart, and Krzysztof Czarnecki. Learning rate based branching heuristic for sat solvers. In *International Conference on Theory and Applications of Satisfiability Testing*, pages 123–140. Springer, 2016.

[45] Andrea Lodi and Giulia Zarpellon. On learning and branching: a survey. *TOP: An Official Journal of the Spanish Society of Statistics and Operations Research*, 25(2):207–236, 2017.

[46] Olvi L Mangasarian and T-H Shiau. Lipschitz continuity of solutions of linear inequalities, programs and complementarity problems. *SIAM Journal on Control and Optimization*, 25(3): 583–595, 1987.

[47] John Milnor. On the Betti numbers of real varieties. *Proceedings of the American Mathematical Society*, 15(2):275–280, 1964.

[48] Atsushi Miyauchi, Tomohiro Sonobe, and Noriyoshi Sukegawa. Exact clustering via integer programming and maximum satisfiability. In *AAAI Conference on Artificial Intelligence*, 2018.

[49] George Nemhauser and Laurence Wolsey. *Integer and Combinatorial Optimization*. John Wiley & Sons, 1999.

[50] David Pollard. *Convergence of Stochastic Processes*. Springer, 1984.

[51] Ashish Sabharwal, Horst Samulowitz, and Chandra Reddy. Guiding combinatorial optimization with UCT. In *International Conference on AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*. Springer, 2012.

[52] Tuomas Sandholm. Very-large-scale generalized combinatorial multi-attribute auctions: Lessons from conducting $60 billion of sourcing. In Zvika Neeman, Alvin Roth, and Nir Vulkan, editors, *Handbook of Market Design*. Oxford University Press, 2013.

[53] Yunhao Tang, Shipra Agrawal, and Yuri Faenza. Reinforcement learning for integer programming: Learning to cut. *International Conference on Machine Learning (ICML)*, 2020.

[54] René Thom. Sur l'homologie des varietes algebriques reelles. In *Differential and combinatorial topology*, pages 255–265. Princeton University Press, 1965.

[55] Mark Turner, Thorsten Koch, Felipe Serrano, and Michael Winkler. Adaptive cut selection in mixed-integer linear programming. *arXiv preprint arXiv:2202.10962*, 2022.

[56] Hugh E Warren. Lower bounds for approximation by nonlinear manifolds. *Transactions of the American Mathematical Society*, 133(1):167–178, 1968.

[57] Gellért Weisz, András György, and Csaba Szepesvári. LEAPSANDBOUNDS: A method for approximately optimal algorithm configuration. In *International Conference on Machine Learning (ICML)*, 2018.

[58] Wei Xia and Roland Yap. Learning robust search strategies using a bandit-based approach. In *AAAI Conference on Artificial Intelligence*, 2018.

[59] Lin Xu, Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. Satzilla: portfolio-based algorithm selection for SAT. *Journal of Artificial Intelligence Research*, 32(1):565–606, 2008.

[60] Lin Xu, Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. Hydra-MIP: Automated algorithm configuration and selection for mixed integer programming. In *RCRA workshop on Experimental Evaluation of Algorithms for Solving Problems with Combinatorial Explosion at the International Joint Conference on Artificial Intelligence (IJCAI)*, 2011.

[61] Jiaming Zeng, Berk Ustun, and Cynthia Rudin. Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 180(3): 689–722, 2017.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 6
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Further details about plots

The version of the *facility location* problem we study involves a set of locations J and a set of clients C. Facilities are to be constructed at some subset of the locations, and the clients in C are served by these facilities. Each location $j \in J$ has a cost f_j of being the site of a facility, and a cost $s_{c,j}$ of serving client $c \in C$. Finally, each location j has a capacity κ_j which is a limit on the number of clients j can serve. The goal of the facility location problem is to arrive at a feasible set of locations for facilities and a feasible assignment of clients to these locations that minimizes the overall cost incurred.

The facility location problem can be formulated as the following $0, 1$ IP:

$$
\begin{array}{lll}
\text{minimize} & \sum_{j \in J} f_j x_j + \sum_{j \in J} \sum_{c \in C} s_{c,j} y_{c,j} & \\
\text{subject to} & \sum_{j \in J} y_{c,j} = 1 & \forall\, c \in C \\
& \sum_{c \in C} y_{c,j} \leq \kappa_j x_j & \forall\, j \in J \\
& y_{c,j} \in \{0, 1\} & \forall\, c \in C, j \in J \\
& x_j \in \{0, 1\} & \forall\, j \in J
\end{array}
$$

We consider the following two distributions over facility location IPs.

First distribution Facility location IPs are generated by perturbing the costs and capacities of a base facility location IP. We generated the base IP with 40 locations and 40 clients by choosing the location costs and client-location costs uniformly at random from $[0, 100]$ and the capacities uniformly at random from $\{0, \ldots, 39\}$. To sample from the distribution, we perturb this base IP by adding independent Gaussian noise with mean 0 and standard deviation 10 to the cost of each location, the cost of each client-location pair, and the capacity of each location.

Second distribution Facility location IPs are generated by placing 80 evenly-spaced locations along the line segment connecting the points $(0, 1/2)$ and $(1, 1/2)$ in the Cartesian plane. The location costs are all uniformly set to 1. Then, 80 clients are placed uniformly at random in the unit square $[0, 1]^2$. The cost $s_{c,j}$ of serving client c from location j is the distance between j and c. Location capacities are chosen uniformly at random from $\{0, \ldots, 43\}$.

In our experiments, we add five cuts at the root of the B&C tree. These five cuts come from the set of Chvátal-Gomory and Gomory mixed integer cuts derived from the optimal simplex tableau of the LP relaxation. The five cuts added are chosen to maximize a weighting of cutting-plane scores:

$$\mu \cdot \texttt{score}_1 + (1 - \mu) \cdot \texttt{score}_2. \tag{4}$$

$\texttt{score}_1$ is the *parallelism* of a cut, which intuitively measures the angle formed by the objective vector and the normal vector of the cutting plane—promoting cutting planes that are nearly parallel with the objective direction. $\texttt{score}_2$ is the *efficacy*, or depth, of a cut, which measures the perpendicular distance from the LP optimum to the cut—promoting cutting planes that are “deeper", as measured with respect to the LP optimum. More details about these scoring rules can be found in Balcan et al. [10] and references therein. Given an IP, for each $\mu \in [0, 1]$ (discretized at steps of 0.01) we choose the five cuts among the set of Chvátal-Gomory and Gomory mixed integer cuts that maximize (4). Figures 1a and 1b display the average tree size over 1000 samples drawn from the respective distribution for each value of μ used to choose cuts at the root. We ran our experiments using the C API of IBM ILOG CPLEX 20.1.0, with default cut generation disabled, and a 64-core machine with 512 GB of RAM.

B Omitted results and proofs from Section 3

B.1 Example in two dimensions

Consider the LP

$$\max\{x + y : x \leq 1, y \geq 0, y \leq x\}.$$

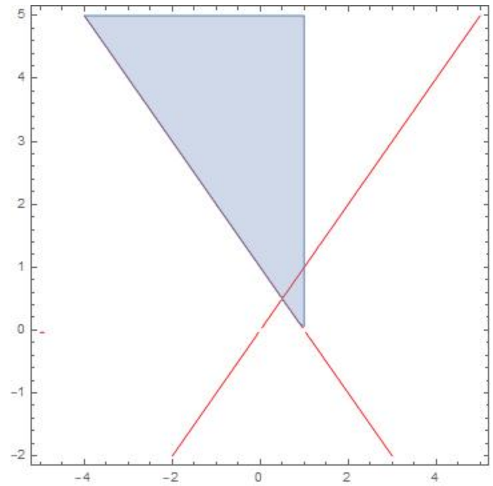


Figure 3: Decomposition of the parameter space: the blue region contains the set of (α_1, α_2) such that the constraint intersects the feasible region at $x = 1$ and $x = y$. The red lines consist of all (α_1, α_2) such that the objective value is equal at these intersection points. The red lines partition the blue region into two components: one where the new optimum is achieved at the intersection of h and $x = y$, and one where the new optimum is achieved at the intersection of h and $x = 1$.

The optimum is at $(x^*, y^*) = (1, 1)$. Consider adding an additional constraint $\alpha_1 x + \alpha_2 y \le 1$. Let h denote the hyperplane $\alpha_1 x + \alpha_2 y = 1$. We derive a description of the set of parameters (α_1, α_2) such that h intersects the hyperplanes $x = 1$ and $y = x$. The intersection of h and $x = 1$ is given by

$$(x, y) = \left(1, \frac{1-\alpha_1}{\alpha_2}\right),$$

which exists if and only if $\alpha_2 \neq 0$. This intersection point is in the LP feasible region if and only if $0 \le \frac{1-\alpha_1}{\alpha_2} \le 1$ (which additionally ensures that $\alpha_2 \neq 0$). Similarly, h intersects $y = x$ at

$$(x, y) = \left(\frac{1}{\alpha_1 + \alpha_2}, \frac{1}{\alpha_1 + \alpha_2}\right),$$

which exists if and only if $\alpha_1 + \alpha_2 \neq 0$. This intersection point is in the LP feasible region if and only if $0 \le \frac{1}{\alpha_1+\alpha_2} \le 1$. Now, we put down an "indifference" curve in (α_1, α_2)-space that represents the set of (α_1, α_2) such that the value of the objective achieved at the two aforementioned intersection points is equal. This surface is given by

$$\frac{2}{\alpha_1 + \alpha_2} = 1 + \frac{1-\alpha_1}{\alpha_2}.$$

Since $\alpha_1 + \alpha_2 \neq 0$ and $\alpha_2 \neq 0$ (for the relevant α_1, α_2 in consideration), this is equivalent to $\alpha_1^2 - \alpha_2^2 - \alpha_1 + \alpha_2 = 0$, which is a degree-2 curve in α_1, α_2. The left-hand-side can be factored to write this as $(\alpha_1 - \alpha_2)(\alpha_1 + \alpha_2 - 1) = 0$. Therefore, this curve is given by the two lines $\alpha_1 = \alpha_2$ and $\alpha_1 + \alpha_2 = 1$. Figure 3 illustrates the resulting partition of (α_1, α_2)-space.

It turns out that when $n = 2$ the indifference curve can always be factored into a product of linear terms. Let the objective of the LP be (c_1, c_2), and let $s_1 x + s_2 y = u_1$ and $t_1 x + t_2 y = v_1$ be two intersecting edges of the LP feasible region. Let $\alpha_1 x + \alpha_2 y = \beta$ be an additional constraint. The intersection points of this constraint with the two lines, if they exist, are given by

$$\left(\frac{s_2\beta - u\alpha_2}{s_2\alpha_1 - s_1\alpha_2}, \frac{s_1\beta - u\alpha_1}{s_1\alpha_2 - s_2\alpha_1}\right) \text{ and } \left(\frac{t_2\beta - v\alpha_2}{t_2\alpha_1 - t_1\alpha_2}, \frac{t_2\beta - v\alpha_1}{t_1\alpha_2 - t_2\alpha_1}\right).$$

The indifference surface is thus given by

$$c_1\frac{s_2\beta - u\alpha_2}{s_2\alpha_1 - s_1\alpha_2} + c_2\frac{s_1\beta - u\alpha_1}{s_1\alpha_2 - s_2\alpha_1} = c_1\frac{t_2\beta - v\alpha_2}{t_2\alpha_1 - t_1\alpha_2} + c_2\frac{t_2\beta - v\alpha_1}{t_1\alpha_2 - t_2\alpha_1}.$$

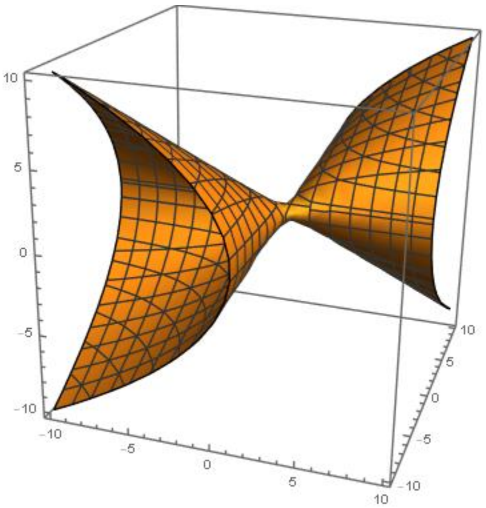


Figure 4: Indifference surface for two edges of the feasible region of an LP in three variables.

For α_1, α_2 such that $s_2\alpha_1 - s_1\alpha_2 \neq 0$ and $t_2\alpha_1 - t_1\alpha_2 \neq 0$, clearing denominators and some manipulation yields

$$(c_1\alpha_2 - c_2\alpha_1)((ut_1 - vs_1)\alpha_2 - (ut_2 - vs_2)\alpha_1 + (s_2t_2 - t_1s_2)\beta) = 0.$$

This curve consists of the two planes $c_1\alpha_2 - c_2\alpha_1 = 0$ and $(ut_1 - vs_1)\alpha_2 - (ut_2 - vs_2)\alpha_1 + (s_2t_2 - t_1s_2)\beta = 0$.

This is however not true if $n > 2$. For example, consider an LP in three variables x, y, z with the constraints $x + y \leq 1, x + z \leq 1, x \leq 1, z \leq 1$. Writing out the indifference surface (assuming the objective is $\boldsymbol{c} = (1, 1, 1)^T$) for the vertex on the intersection of $\{x + y = 1, x = 1\}$ and the vertex on $\{x + z = 1, z = 1\}$ yields

$$\alpha_1\alpha_2 - \alpha_2\beta - \alpha_3^2 + \alpha_3\beta = 0.$$

Setting $\beta = 1$, we can plot the resulting surface in $\alpha_1, \alpha_2, \alpha_3$ (Figure 4).

B.2 Linear programming sensitivity for multiple constraints

Lemma B.1. *Let $(\boldsymbol{c}, A, \boldsymbol{b})$ be an LP and let M denote the set of its m constraints. Let $\boldsymbol{x}^*_{\mathsf{LP}}$ and z^*_{LP} denote the optimal solution and its objective value, respectively. For $F \subseteq M$, let $A_F \in \mathbb{R}^{|F|\times n}$ and $\boldsymbol{b}_F \in \mathbb{R}^{|F|}$ denote the restrictions of A and $\boldsymbol{b}$ to F. For $k \leq n$, $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_k \in \mathbb{R}^n$, $\beta_1, \ldots, \beta_k \in \mathbb{R}$, and $F \subseteq M$ with $|F| = n - k$, let $A_{F,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k} \in \mathbb{R}^{n\times n}$ denote the matrix obtained by adding row vectors $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_k$ to A_F and let $A^i_{F,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k} \in \mathbb{R}^{n\times n}$ be the matrix $A_{F,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k} \in \mathbb{R}^{n\times n}$ with the ith column replaced by $\begin{bmatrix}\boldsymbol{b}_F & \beta_1 & \cdots & \beta_k\end{bmatrix}^T$. There is a set of at most K hyperplanes, nK^nm^n degree-K polynomial hypersurfaces, and nK^nm^{2n} degree-$2K$ polynomial hypersurfaces partitioning $\mathbb{R}^{K(n+1)}$ into connected components such that for each component C, one of the following holds: either (1) $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K) = \boldsymbol{x}^*_{\mathsf{LP}}$, or (2) there is a subset of cuts indexed by $\ell_1, \ldots, \ell_k \in [K]$ and a set of constraints $F \subseteq M$ with $|F| = n - k$ such that*

$$\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K) = \left(\frac{\det(A^1_{F,\boldsymbol{\alpha}_{\ell_1},\beta_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k},\beta_{\ell_k}})}{\det(A_{F,\boldsymbol{\alpha}_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k}})}, \ldots, \frac{\det(A^n_{F,\boldsymbol{\alpha}_{\ell_1},\beta_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k},\beta_{\ell_k}})}{\det(A_{F,\boldsymbol{\alpha}_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k}})}\right),$$

for all $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K) \in C$.

Proof. First, if none of $\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K$ separate $\boldsymbol{x}^*_{\mathsf{LP}}$, then $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K) = \boldsymbol{x}^*_{\mathsf{LP}}$ and $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K) = z^*_{\mathsf{LP}}$. The set of all such cuts is given by the intersection of halfspaces in $\mathbb{R}^{K(n+1)}$ given by

$$\bigcap_{j=1}^{K}\left\{(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_k, \beta_k) \in \mathbb{R}^{K(n+1)} : \boldsymbol{\alpha}_j^T\boldsymbol{x}^*_{\mathsf{LP}} \leq \beta_j\right\}. \tag{5}$$

All other vectors of K cuts contain at least one cut that separates $\boldsymbol{x}^*_{\mathsf{LP}}$, and those cuts therefore pass through $\mathcal{P} = \{\boldsymbol{x} \in \mathbb{R}^n : A\boldsymbol{x} \le \boldsymbol{b}, \boldsymbol{x} \ge \boldsymbol{0}\}$. The new LP optimum is thus achieved at a vertex created by the cuts that separate $\boldsymbol{x}^*_{\mathsf{LP}}$. As in the proof of Theorem 3.1, we consider all possible new vertices formed by our set of K cuts. In the case of a single cut, these new vertices necessarily were on edges of $\mathcal{P}$, but now they may lie on higher dimensional faces.

Consider a subset of $k \le n$ cuts that separate $\boldsymbol{x}^*_{\mathsf{LP}}$. Without loss of generality, denote these cuts by $\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \dots, \boldsymbol{\alpha}_k^T\boldsymbol{x} \le \beta_k$. We now establish conditions for these k cuts to "jointly" form a new vertex of $\mathcal{P}$. Any vertex created by these cuts must lie on a face f of $\mathcal{P}$ with $\dim(f) = k$ (in the case that $k = n$, the relevant face f with $\dim(f) = n$ is $\mathcal{P}$ itself). Letting M denote the set of m constraints that define $\mathcal{P}$, each dimension-k face f of $\mathcal{P}$ can be identified with a (potentially empty) subset $F \subset M$ of size $n - k$ such that f is precisely the set of all points $\boldsymbol{x}$ such that

$$\begin{aligned} \boldsymbol{a}_i^T\boldsymbol{x} &= b_i \qquad \forall\, i \in F \\ \boldsymbol{a}_i^T\boldsymbol{x} &\le b_i \qquad \forall\, i \in M \setminus F, \end{aligned}$$

where $\boldsymbol{a}_i$ is the ith row of A. Let $A_F \in \mathbb{R}^{n-k\times n}$ denote the restriction of A to only the rows in F, and let $\boldsymbol{b}_F \in \mathbb{R}^{n-k}$ denote the entries of $\boldsymbol{b}$ corresponding to the constraints in F. Consider removing the inequality constraints defining the face. The intersection of the cuts $\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \dots, \boldsymbol{\alpha}_k^T\boldsymbol{x} \le \beta_k$ and this unbounded surface (if it exists) is precisely the solution to the system of n linear equations

$$\begin{aligned} A_F\boldsymbol{x} &= \boldsymbol{b}_F \\ \boldsymbol{\alpha}_1^T\boldsymbol{x} &= \beta_1 \\ &\ \vdots \\ \boldsymbol{\alpha}_k^T\boldsymbol{x} &= \beta_k. \end{aligned}$$

Let $A_{F,\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_k} \in \mathbb{R}^{n\times n}$ denote the matrix obtained by adding row vectors $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_k$ to A_F, and let $A^i_{F,\boldsymbol{\alpha}_1,\beta_1,\dots,\boldsymbol{\alpha}_k,\beta_k} \in \mathbb{R}^{n\times n}$ denote the matrix $A_{F,\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_k}$ where the ith column is replaced by

$$\begin{bmatrix} \boldsymbol{b}_F \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} \in \mathbb{R}^n.$$

By Cramer's rule, the solution to this system is given by

$$\boldsymbol{x} = \left(\frac{\det(A^1_{F,\boldsymbol{\alpha}_1,\beta_1,\dots,\boldsymbol{\alpha}_k,\beta_k})}{\det(A_{F,\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_k})}, \dots, \frac{\det(A^n_{F,\boldsymbol{\alpha}_1,\beta_1,\dots,\boldsymbol{\alpha}_k,\beta_k})}{\det(A_{F,\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_k})} \right),$$

and the value of the objective at this point is

$$\boldsymbol{c}^T\boldsymbol{x} = \sum_{i=1}^n c_i \cdot \frac{\det(A^i_{F,\boldsymbol{\alpha}_1,\beta_1,\dots,\boldsymbol{\alpha}_k,\beta_k})}{\det(A_{F,\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_k})}.$$

Now, to ensure that the unique intersection point $\boldsymbol{x}$ (1) exists and (2) actually lies on f (or simply lies in $\mathcal{P}$, in the case that $F = \emptyset$) , we stipulate that it satisfies the inequality constraints in $M \setminus F$. That is,

$$\sum_{j=1}^n a_{ij} \frac{\det(A^1_{F,\boldsymbol{\alpha}_1,\beta_1,\dots,\boldsymbol{\alpha}_k,\beta_k})}{\det(A_{F,\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_k})} \le b_i \tag{6}$$

for every $i \in M \setminus F$. If $\boldsymbol{\alpha}_1, \beta_1 \dots, \boldsymbol{\alpha}_k, \beta_k$ satisfies any of these constraints, it must be that $\det(A_{F,\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_k}) \ne 0$, which guarantees that $A_F\boldsymbol{x} = \boldsymbol{b}_F, \boldsymbol{\alpha}_1^T\boldsymbol{x} = \beta_1, \dots, \boldsymbol{\alpha}_k^T\boldsymbol{x} = \beta_k$ indeed has a unique solution. Now, $\det(A_{F,\boldsymbol{\alpha}_1,\dots,\boldsymbol{\alpha}_k})$ is a polynomial in $\boldsymbol{\alpha}_1, \dots, \boldsymbol{\alpha}_k$ of degree $\le k$, since it is multilinear in each coefficient of each $\boldsymbol{\alpha}_\ell$, $\ell = 1, \dots, k$. Similarly, $\det(A^1_{F,\boldsymbol{\alpha}_1,\beta_1,\dots,\boldsymbol{\alpha}_k,\beta_k})$ is a polynomial in $\boldsymbol{\alpha}_1, \beta_1, \dots, \boldsymbol{\alpha}_k, \beta_k$ of degree $\le k$, again because it is multilinear in each cut parameter. Hence, the boundary each constraint of the form given by Equation 6 is a polynomial of degree at most k.

The collection of these polynomials for every k, every subset of $\{\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \le \beta_K\}$ of size k, and every face of $\mathcal{P}$ of dimension k, along with the hyperplanes determining separation constraints (Equation 5), partition $\mathbb{R}^{K(n+1)}$ into connected components such that for all $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K)$ within a given connected component, there is a fixed subset of K and a fixed set of faces of $\mathcal{P}$ such that the cuts with indices in that subset intersect every face in the set at a common vertex.

Now, consider a single connected component, denoted by C. Let $f_1, \ldots, f_\ell$ denote the faces intersected by vectors of cuts in C, and let (without loss of generality) $1, \ldots, k$ denote the subset of cuts that intersect these faces. Let $F_1, \ldots, F_\ell \subset M$ denote the sets of constraints that are binding at each of these faces, respectively. For each pair f_p, f_q, consider the surface

$$\sum_{i=1}^n c_i \cdot \frac{\det(A^i_{F_p,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k})}{\det(A_{F_p,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k})} = \sum_{i=1}^n c_i \cdot \frac{\det(A^i_{F_q,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k})}{\det(A_{F_q,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k})},$$

which can be equivalently written as

$$\sum_{i=1}^n c_i \cdot \det(A^i_{F_p,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k}) \det(A_{F_q,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k}) = \sum_{i=1}^n c_i \cdot \det(A^i_{F_q,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k}) \det(A_{F_p,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k}). \tag{7}$$

This is a degree-$2k$ polynomial hypersurface in $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K) \in \mathbb{R}^{K(n+1)}$. This hypersurface is precisely the set of all cut vectors for which the LP objective achieved at the vertex on face f_p is equal to the LP objective value achieved at the vertex on face f_q. The collection of these surfaces for each p, q partitions C into further connected components. Within each of these connected components, the face containing the vertex that maximizes the objective is invariant, and the subset of cuts passing through that vertex is invariant. If $F \subseteq M$ is the set of binding constraints representing this face, and $\ell_1, \ldots, \ell_k \in [K]$ represent the subset of cuts intersecting this face, $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \le \beta_K)$ and $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \le \beta_K)$ have the closed forms:

$$\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \le \beta_K) = \left(\frac{\det(A^1_{F,\boldsymbol{\alpha}_{\ell_1},\beta_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k},\beta_{\ell_k}})}{\det(A_{F,\boldsymbol{\alpha}_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k}})}, \ldots, \frac{\det(A^n_{F,\boldsymbol{\alpha}_{\ell_1},\beta_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k},\beta_{\ell_k}})}{\det(A_{F,\boldsymbol{\alpha}_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k}})} \right),$$

and

$$z^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \le \beta_K) = \sum_{i=1}^n c_i \cdot \frac{\det(A^i_{F,\boldsymbol{\alpha}_{\ell_1},\beta_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k},\beta_{\ell_k}})}{\det(A_{F,\boldsymbol{\alpha}_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k}})}.$$

for all $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K)$ within this component. We now count the number of surfaces used to obtain our decomposition. First, we added K hyperplanes encoding separation constraints for each of the K cuts (Equation 5). Then, for every subset $S \subseteq K$ of size $\le n$, and for every face F of $\mathcal{P}$ with $\dim(F) = |S|$, we first considered at most $|M \setminus F| \le m$ degree-$\le K$ polynomial hypersurfaces representing decision boundaries for when cuts in S intersected that face (Equation 6). The number of k-dimensional faces of $\mathcal{P}$ is at most $\binom{m}{n-k} \le m^{n-k} \le m^{n-1}$, so the total number of these hypersurfaces is at most $(\binom{K}{0} + \cdots + \binom{K}{n})m^n \le nK^n m^n$. Finally, we considered a degree-$2K$ polynomial hypersurface for every subset of cuts and every pair of faces with degree equal to the size of the subset, of which there are at most $nK^n\binom{m^n}{2} \le nK^n m^{2n}$. ☐

C Omitted results and proofs from Section 4

We first require the following lemma which bounds the number of relevant subsets of $\mathcal{BC} := \{\boldsymbol{x}[i] \le \ell, \boldsymbol{x}[i] \ge \ell\}_{0 \le \ell \le \tau, i \in [n]}$ that define a possible node expanded by B&C. $\mathcal{BC}$ is a set of size $2n(\tau + 1)$ so naïvely there are at most $2^{2n(\tau+1)}$ subsets of branching constraints. The following observation allows us to greatly reduce the number of sets we consider.

Lemma C.1. *Fix an IP $(\boldsymbol{c}, A, \boldsymbol{b})$. Define an equivalence relation on pairs of branching-constraint sets $\sigma_1, \sigma_2 \subseteq \mathcal{BC}$, by $\sigma_1 \sim \sigma_2 \iff \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_1) = \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_2)$ for all possible cutting planes $\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta$. The number of equivalence classes of $\sim$ is at most τ^{3n}.*

Proof of Lemma C.1. Consider as an example $\sigma_1 = \{\boldsymbol{x}[1] \le 1, \boldsymbol{x}[1] \le 5\}$ and $\sigma_2 = \{\boldsymbol{x}[1] \le 1\}$. We have $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_1) = \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_2)$ for any cut $\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta$, because the constraint

$\boldsymbol{x}[1] \leq 5$ is redundant in σ_1. More generally, any $\sigma \subseteq \mathcal{BC}$ can be reduced by preserving only the tightest $\leq$ constraint and tightest $\geq$ constraint without affecting the resulting LP optimal solutions. The number of such unique reduced sets is at most $((\tau+2)^2)^n < \tau^{3n}$ (for each variable, there are $\tau + 2$ possibilities for the tightest $\leq$ constraint: no constraint or one of $\boldsymbol{x}[i] \leq 0, \ldots, \boldsymbol{x}[i] \leq \tau$, and similarly $\tau + 2$ possibilities for the $\geq$ constraint). $\square$

Proof of Lemma 4.1. We carry out the same reasoning in the proof of Theorem 3.1 for each reduced σ. The number of edges of $\mathcal{P}(\sigma)$ is at most $\binom{m+|\sigma|}{n-1} \leq (m+|\sigma|)^{n-1}$. For each edge E, we considered at most $|(M \cup \sigma) \setminus E| \leq m + |\sigma|$ hyperplanes, for a total of at most $(m+|\sigma|)^n$ halfspaces. Then, we had a degree-2 polynomial hypersurface for every pair of edges, of which there are at most $\binom{(m+|\sigma|)^n}{2} \leq (m+|\sigma|)^{2n}$. Summing over all reduced σ (of which there are at most τ^{3n}), combined with the fact that if σ is reduced then $|\sigma| \leq 2n$, we get a total of at most $(m+2n)^n\tau^{3n}$ hyperplanes and at most $(m+2n)^{2n}\tau^{3n}$ degree-2 hypersurfaces, as desired. $\square$

Let $\mathcal{V} \subseteq \mathbb{R}^{n+1}$ denote the set of all valid cuts for the input IP $(\boldsymbol{c}, A, \boldsymbol{b})$. The set $\mathcal{V}$ is a polyhedron since it can be expressed as

$$\mathcal{V} = \bigcap_{\overline{\boldsymbol{x}} \in \mathcal{P}_{\mathsf{I}}} \{(\boldsymbol{\alpha}, \beta) \in \mathbb{R}^{n+1} : \boldsymbol{\alpha}^T \overline{\boldsymbol{x}} \leq \beta\},$$

and $\mathcal{P}_{\mathsf{I}}$ is finite as $\mathcal{P}$ is bounded. For cuts outside $\mathcal{V}$, we assume the B&C tree takes some special form denoting an invalid cut. Our goal now is to decompose $\mathcal{V}$ into connected components such that $\mathbf{1}\left[\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma) \in \mathbb{Z}^n\right]$ is invariant for all $(\boldsymbol{\alpha}, \beta)$ in each component.

Proof of Lemma 4.3. Fix a connected component C in the decomposition that includes the facets defining $\mathcal{V}$ and the surfaces obtained in Lemma 4.2. For all $\sigma \in \mathcal{BC}$, $\boldsymbol{x}_{\mathsf{I}} \in \mathcal{P}_{\mathsf{I}}$, and $i = 1, \ldots, n$, consider the surface

$$\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)[i] = \boldsymbol{x}_{\mathsf{I}}[i]. \tag{8}$$

This surface is a hyperplane, since by Lemma 4.1, either $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)[i] = \boldsymbol{x}^*_{\mathsf{LP}}(\sigma)[i]$ or $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)[i] = \frac{\det(A^i_{E,\boldsymbol{\alpha},\beta,\sigma})}{\det(A_{E,\boldsymbol{\alpha},\sigma})}$, where $E \subseteq M \cup \sigma$ is the subset of constraints corresponding to σ and C. Clearly, within any connected component of C induced by these hyperplanes, for every σ and $\boldsymbol{x}_{\mathsf{I}} \in \mathcal{P}_{\mathsf{I}}$, $\mathbf{1}[\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma) = \boldsymbol{x}_{\mathsf{I}}]$ is invariant. Finally, if $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma) \in \mathbb{Z}^n$ for some cut $\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta$ within a given connected component, $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma) = \boldsymbol{x}_{\mathsf{I}}$ for some $\boldsymbol{x}_{\mathsf{I}} \in \mathcal{P}_{\mathsf{IH}}(\sigma) \subseteq \mathcal{P}_{\mathsf{I}}$, which means that $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma) = \boldsymbol{x}_{\mathsf{I}} \in \mathbb{Z}^n$ *for all* cuts $\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta$ in that connected component.

We now count the number of hyperplanes given by Equation 8. For each σ, there are $\binom{m+|\sigma|}{n-1} \leq (m+2n)^{n-1}$ binding edge constraints $E \subseteq M \cup \sigma$ defining the formula of Lemma 4.1, and we have $n|\mathcal{P}_{\mathsf{I}}|$ hyperplanes for each E. Since $\tau = \max_{\boldsymbol{x} \in \mathcal{P}_{\mathsf{I}}} \|\boldsymbol{x}\|_\infty$, $|\mathcal{P}_{\mathsf{I}}| \leq \tau^n$. So the total number of hyperplanes given by Equation 8 is at most $\tau^{3n}(m+2n)^{n-1}n\tau^n \leq (m+2n)^n\tau^{4n}$. The number of facets defining $\mathcal{V}$ is at most $|\mathcal{P}_{\mathsf{IH}}| \leq |\mathcal{P}_{\mathsf{I}}| \leq \tau^n$. Adding these to the counts obtained in Lemma 4.2 yields the final tallies in the lemma statement. $\square$

Proof of Theorem 4.4. Fix a connected component C in the decomposition induced by the set of hyperplanes and degree-2 hypersurfaces established in Lemma 4.3. Let

$$Q_1, \ldots, Q_{i_1}, I_1, Q_{i_1+1}, \ldots, Q_{i_2}, I_2, Q_{i_2+1}, \ldots \tag{9}$$

denote the nodes of the tree branch-and-cut creates, in order of exploration, under the assumption that a node is pruned if and only if either the LP at that node is infeasible or the LP optimal solution is integral (so the “bounding” of branch-and-bound is suppressed). Here, a node is identified by the list σ of branching constraints added to the input IP. Nodes labeled by Q are either infeasible or have fractional LP optimal solutions. Nodes labeled by I have integral LP optimal solutions and are candidates for the incumbent integral solution at the point they are encountered. (The nodes are functions of $\boldsymbol{\alpha}$ and β, as are the indices $i_1, i_2, \ldots$.) By Lemma 4.3 and the observation following it, this ordered list of nodes is invariant over all $(\boldsymbol{\alpha}, \beta) \in C$.

Now, given an node index ℓ, let $I(\ell)$ denote the incumbent node with the highest objective value encountered up until the ℓth node searched by B&C, and let $z(I(\ell))$ denote its objective value. For each node Q_ℓ, let σ_ℓ denote the branching constraints added to arrive at node Q_ℓ. The hyperplane

$$z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_\ell) = z(I(\ell)) \tag{10}$$

(which is a hyperplane due to Lemma 4.1) partitions C into two subregions. In one subregion, $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_\ell) \le z(I(\ell))$, that is, the objective value of the LP optimal solution is no greater than the objective value of the current incumbent integer solution, and so the subtree rooted at Q_ℓ is pruned. In the other subregion, $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_\ell) > z(I(\ell))$, and Q_ℓ is branched on further. Therefore, within each connected component of C induced by all hyperplanes given by Equation 10 for all ℓ, the set of node within the list (9) that are pruned is invariant. Combined with the surfaces established in Lemma 4.3, these hyperplanes partition $\mathbb{R}^{n+1}$ into connected components such that as $(\boldsymbol{\alpha}, \beta)$ varies within a given component, the tree built by branch-and-cut is invariant.

Finally, we count the total number of surfaces inducing this partition. Unlike the counting stages of the previous lemmas, we will first have to count the number of connected components induced by the surfaces established in Lemma 4.3. This is because the ordered list of nodes explored by branch-and-cut (9) can be different across each component, and the hyperplanes given by Equation 10 depend on this list. From Lemma 4.3 we have $3(m+2n)^n\tau^{4n}$ hyperplanes, $3(m+2n)^{3n}\tau^{4n}$ degree-2 polynomial hypersurfaces, and $(m+2n)^{6n}\tau^{4n}$ degree-5 polynomial hypersurfaces. To determine the connected components of $\mathbb{R}^{n+1}$ induced by the zero sets of these polynomials, it suffices to consider the zero set of the product of all polynomials defining these surfaces. Denote this product polynomial by p. The degree of the product polynomial is the sum of the degrees of $3(m+2n)^n\tau^{4n}$ degree-1 polynomials, $3(m+2n)^{3n}\tau^{4n}$ degree-2 polynomials, and $(m+2n)^{6n}\tau^{4n}$ degree-5 polynomials, which is at most $3(m+2n)^n\tau^{4n} + 2\cdot 3(m+2n)^{3n}\tau^{4n} + 5\cdot(m+2n)^{6n}\tau^{4n} < 14(m+2n)^{3n}\tau^{4n}$. By Warren's theorem, the number of connected components of $\mathbb{R}^{n+1} \setminus \{(\boldsymbol{\alpha}, \beta) : p(\boldsymbol{\alpha}, \beta) = 0\}$ is $O((14(m+2n)^{3n}\tau^{4n})^{n-1})$, and by the Milnor-Thom theorem, the number of connected components of $\{(\boldsymbol{\alpha}, \beta) : p(\boldsymbol{\alpha}, \beta) = 0\}$ is $O((14(m+2n)^{3n}\tau^{4n})^{n-1})$ as well. So, the number of connected components induced by the surfaces in Lemma 4.3 is $O(14^n(m+2n)^{3n^2}\tau^{4n^2})$. For every connected component C in Lemma 4.3, the closed form of $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_\ell)$ is already determined due to Lemma 4.1, and so the number of hyperplanes given by Equation 10 is at most the number of possible $\sigma \subseteq \mathcal{BC}$, which is at most τ^{3n}. So across all connected components C, the total number of hyperplanes given by Equation 10 is $O(14^n(m+2n)^{3n^2}\tau^{5n^2})$. Finally, adding this to the surface-counts established in Lemma 4.3 yields the lemma statement. $\square$

C.1 Product scoring rule for variable selection

Let σ be the set of branching constraints added thus far. The product scoring rule branches on the variable $i \in [n]$ that maximizes:

$$\max\{z^*_{\mathsf{LP}}(\sigma) - z^*_{\mathsf{LP}}(x_i \le \lfloor x^*_{\mathsf{LP}}(\sigma)[i]\rfloor, \sigma), \gamma\} \cdot \max\{z^*_{\mathsf{LP}}(\sigma) - z^*_{\mathsf{LP}}(x_i \ge \lceil x^*_{\mathsf{LP}}(\sigma)[i]\rceil, \sigma), \gamma\},$$

where $\gamma = 10^{-6}$.

Lemma C.2. *There is a set of of at most $3(m+2n)^n\tau^{3n}$ hyperplanes and $(m+2n)^{2n}\tau^{3n}$ degree-2 polynomial hypersurfaces partitioning $\mathbb{R}^{n+1}$ into connected components such that for any connected component C and any σ, the set of branching constraints $\{x_i \le \lfloor \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i]\rfloor, x_i \ge \lceil \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i]\rceil \mid i \in [n]\}$ is invariant across all $(\boldsymbol{\alpha}, \beta) \in C$.*

Proof. Fix a connected component C in the decomposition established in Lemma 4.1. By Lemma 4.1, for each σ, either $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) = \boldsymbol{x}^*_{\mathsf{LP}}(\sigma)$ or there exists $E \subseteq M \cup \sigma$ such that $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i] = \frac{\det(A^i_{E,\boldsymbol{\alpha},\beta,\sigma})}{\det(A_{E,\boldsymbol{\alpha},\sigma})}$ for all $(\boldsymbol{\alpha}, \beta) \in C$. Fix a variable $i \in [n]$, which corresponds to two branching constraints

$$x_i \le \lfloor \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i]\rfloor \text{ and } x_i \ge \lceil \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i]\rceil. \tag{11}$$

If C is a component where $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) = \boldsymbol{x}^*_{\mathsf{LP}}(\sigma)$, then these two branching constraints are trivially invariant over $(\boldsymbol{\alpha}, \beta) \in C$. Otherwise, in order to further decompose C such that the right-hand-sides of these constraints are invariant for every σ, we add the two decision boundaries given by

$$k \le \frac{\det(A^i_{E,\boldsymbol{\alpha},\beta,\sigma})}{\det(A_{E,\boldsymbol{\alpha},\sigma})} \le k+1$$

for every i, σ, and every integer $k = 0, \dots, \tau - 1$, where $\tau = \max_{\boldsymbol{x} \in \mathcal{P} \cap \mathbb{Z}^n} \|\boldsymbol{x}\|_\infty$. This ensures that within every connected component of C induced by these boundaries (hyperplanes),

$$\left\lfloor \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i] \right\rfloor = \left\lfloor \frac{\det(A^i_{E,\boldsymbol{\alpha},\beta,\sigma})}{\det(A_{E,\boldsymbol{\alpha},\sigma})} \right\rfloor \text{ and } \left\lceil \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i] \right\rceil = \left\lceil \frac{\det(A^i_{E,\boldsymbol{\alpha},\beta,\sigma})}{\det(A_{E,\boldsymbol{\alpha},\sigma})} \right\rceil$$

are invariant, so the branching constraints from Equation (11) are invariant. For a fixed σ, there are two hyperplanes for every $E \subseteq M \cup \sigma$ corresponding to an edge of $\mathcal{P}(\sigma)$ and $i = 1, \dots, n$, for a total of at most $2n\binom{m+|\sigma|}{n-1} \le 2n(m+|\sigma|)^{n-1}$ hyperplanes. Summing over all reduced σ, we get a total of $2n(m+2n)^{n-1}\tau^{3n} < 2(m+2n)^n\tau^{3n}$ hyperplanes. Adding these hyperplanes to the set of hyperplanes established in Lemma 4.1 yields the lemma statement. $\square$

Proof of Lemma 4.2. Fix a connected component C in the decomposition established in Lemma C.2. We know that for each set of branching constraints σ:

- By Lemma 4.1, either $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) = \boldsymbol{x}^*_{\mathsf{LP}}(\sigma)$ or there exists $E \subseteq M \cup \sigma$ such that $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i] = \frac{\det(A^i_{E,\boldsymbol{\alpha},\beta,\sigma})}{\det(A_{E,\boldsymbol{\alpha},\sigma})}$ for all $(\boldsymbol{\alpha}, \beta) \in C$ and all $i \in [n]$, and
- The set of branching constraints $\{x_i \le \left\lfloor \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i] \right\rfloor, x_i \ge \left\lceil \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[i] \right\rceil \mid i \in [n]\}$ is invariant across all $(\boldsymbol{\alpha}, \beta) \in C$.

Suppose that σ is the list of branching constraints added so far. For any variable $k \in [n]$, let

$$\sigma_k^- = (x_k \le \left\lfloor \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[k] \right\rfloor, \sigma) \text{ and } \sigma_k^+ = (x_k \ge \left\lceil \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma)[k] \right\rceil, \sigma).$$

So long as $(\boldsymbol{\alpha}, \beta) \in C$, σ_k^- and σ_k^+ are fixed. With this notation, we can write the product scoring rule as

$$\max\{z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) - z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_k^-), \gamma\} \cdot \max\{z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) - z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_k^+), \gamma\},$$

where $\gamma = 10^{-6}$.

By Lemma 4.1, we know that across all $(\boldsymbol{\alpha}, \beta) \in C$, either $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_k^+) = z^*_{\mathsf{LP}}(\sigma_k^+)$ or there exists $E_k^+ \subseteq M \cup \sigma_k^+$ such that

$$z^*_{\mathsf{LP}}\left(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_k^+\right) = \sum_{i=1}^n c_i \cdot \frac{\det\left(A^i_{E_k^+,\boldsymbol{\alpha},\beta,\sigma_k^+}\right)}{\det\left(A_{E_k^+,\boldsymbol{\alpha},\sigma_k^+}\right)},$$

and similarly for σ_k^-, defined according to some edge set $E_k^- \subseteq M \cup \sigma_k^-$. Therefore, for each $k \in [n]$, there is a single degree-2 polynomial hypersurface partitioning C into connected components such that within each connected component, either

$$z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) - z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_k^-) \ge \gamma \tag{12}$$

or vice versa, and similarly for σ_k^+. In particular, the former hypersurface will have one of four forms:

1. $z^*_{\mathsf{LP}}(\sigma) - z^*_{\mathsf{LP}}(\sigma_k^-) \ge \gamma$, which is uniformly satisfied or not satisfied across all $(\boldsymbol{\alpha}, \beta) \in C$,
2. $z^*_{\mathsf{LP}}(\sigma) - \sum_{i=1}^n c_i \cdot \frac{\det\left(A^i_{E_k^-,\boldsymbol{\alpha},\beta,\sigma_k^-}\right)}{\det\left(A_{E_k^-,\boldsymbol{\alpha},\sigma_k^-}\right)} \ge \gamma$, which is a hyperplane,
3. $\sum_{i=1}^n c_i \cdot \frac{\det\left(A^i_{E,\boldsymbol{\alpha},\beta,\sigma}\right)}{\det(A_{E,\boldsymbol{\alpha},\sigma})} - z^*_{\mathsf{LP}}(\sigma_k^-) \ge \gamma$, which is a hyperplane, or
4. $\sum_{i=1}^n c_i \left(\frac{\det\left(A^i_{E,\boldsymbol{\alpha},\beta,\sigma}\right)}{\det(A_{E,\boldsymbol{\alpha},\sigma})} - \frac{\det\left(A^i_{E_k^-,\boldsymbol{\alpha},\beta,\sigma_k^-}\right)}{\det\left(A_{E_k^+,\boldsymbol{\alpha},\sigma_k^-}\right)}\right) \ge \gamma$, which is a degree-2 polynomial hypersurface.

Simply said, these are all degree-2 polynomial hypersurfaces.

Within any region induced by these hypersurfaces, the comparison between any two variables x_k and x_j will have the form

$$\max\{z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) - z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_k^-), \gamma\} \cdot \max\{z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) - z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_k^+), \gamma\}$$
$$\ge \max\{z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) - z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_j^-), \gamma\} \cdot \max\{z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma) - z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_j^+), \gamma\}$$

which at its most complex will equal

$$\sum_{i=1}^n c_i \left(\frac{\det\left(A^i_{E,\boldsymbol{\alpha},\beta,\sigma}\right)}{\det\left(A_{E,\boldsymbol{\alpha},\sigma}\right)} - \frac{\det\left(A^i_{E_k^-,\boldsymbol{\alpha},\beta,\sigma_k^-}\right)}{\det\left(A_{E_k^-,\boldsymbol{\alpha},\sigma_k^-}\right)} \right) \cdot \sum_{i=1}^n c_i \left(\frac{\det\left(A^i_{E,\boldsymbol{\alpha},\beta,\sigma}\right)}{\det\left(A_{E,\boldsymbol{\alpha},\sigma}\right)} - \frac{\det\left(A^i_{E_k^+,\boldsymbol{\alpha},\beta,\sigma_k^+}\right)}{\det\left(A_{E_k^+,\boldsymbol{\alpha},\sigma_k^+}\right)} \right) \tag{13}$$
$$\ge \sum_{i=1}^n c_i \left(\frac{\det\left(A^i_{E,\boldsymbol{\alpha},\beta,\sigma}\right)}{\det\left(A_{E,\boldsymbol{\alpha},\sigma}\right)} - \frac{\det\left(A^i_{E_j^-,\boldsymbol{\alpha},\beta,\sigma_j^-}\right)}{\det\left(A_{E_j^-,\boldsymbol{\alpha},\sigma_j^-}\right)} \right) \cdot \sum_{i=1}^n c_i \left(\frac{\det\left(A^i_{E,\boldsymbol{\alpha},\beta,\sigma}\right)}{\det\left(A_{E,\boldsymbol{\alpha},\sigma}\right)} - \frac{\det\left(A^i_{E_j^+,\boldsymbol{\alpha},\beta,\sigma_j^+}\right)}{\det\left(A_{E_j^+,\boldsymbol{\alpha},\sigma_j^+}\right)} \right).$$

This inequality can be written as a degree-5 polynomial hypersurface. In any region induced by these hypersurfaces, the variable that branch-and-cut branches on will be fixed.

We now count the total number of hypersurfaces. First, we count the number of degree-2 polynomial hypersurfaces from Equation (12): there is a hypersurface defined by each variable x_k, set of branching constraints σ, cutoff $t \in [\tau]$ such that $\sigma_k^- = (x_k \le t, \sigma)$, set $E \subseteq M \cup \sigma$ corresponding to an edge of $\mathcal{P}(\sigma)$, and set $E_k^- \subseteq M \cup \sigma_k^-$ (and similarly for σ_k^+ and E_k^+). For a fixed σ, this amounts to $2n\tau\binom{m+|\sigma|}{n-1}\binom{m+|\sigma|+1}{n-1} \le 2n\tau(m+|\sigma|+1)^{2(n-1)}$ hypersurfaces. Summing over all τ^{3n} reduced σ, we have $2n\tau^{3n+1}(m+2n+1)^{2(n-1)}$ degree-2 polynomial hypersurfaces.

Next, we count the number of degree-5 polynomial hypersurfaces from Equation (13): there is a hypersurface defined by each pair of variables x_k, x_j, set of branching constraints σ, cutoffs $t_k, t_j \in [\tau]$ such that $\sigma_k^- = (x_k \le t_k, \sigma)$ and $\sigma_j^- = (x_j \le t_j, \sigma)$, and sets $E, E_k^-, E_k^+, E_j^-, E_j^+$ corresponding to edges of $\mathcal{P}(\sigma), \mathcal{P}(\sigma_k^-), \mathcal{P}(\sigma_k^+), \mathcal{P}(\sigma_j^-), \mathcal{P}(\sigma_j^+)$. For a fixed σ, this amounts to $n^2\tau^2\binom{m+|\sigma|}{n-1}\binom{m+|\sigma|+1}{n-1}^4 \le n^2\tau^2(m+|\sigma|+1)^{5(n-1)}$ hypersurfaces. Summing over all τ^{3n} reduced σ, we have $n^2\tau^{3n+2}(m+2n+1)^{5(n-1)}$ degree-5 polynomial hypersurfaces.

Adding these hypersurfaces to those from Lemma C.2, we get the lemma statement. □

C.2 Extension to multiple cutting planes

We can similarly derive a multi-cut version of Lemma 4.1 that controls $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \le \beta_K, \sigma)$ for any set of branching constraints. We use the following notation. Let $(\boldsymbol{c}, A, \boldsymbol{b})$ be an LP and let M denote the set of its m constraints. For $F \subseteq M \cup \sigma$, let $A_{F,\sigma} \in \mathbb{R}^{|F|\times n}$ and $\boldsymbol{b}_{F,\sigma} \in \mathbb{R}^{|F|}$ denote the restrictions of A_σ and $\boldsymbol{b}_\sigma$ to F. For $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_k \in \mathbb{R}^n$, $\beta_1, \ldots, \beta_k \in \mathbb{R}$, and $F \subseteq M \cup \sigma$ with $|F| = n - k$, let $A_{F,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k,\sigma} \in \mathbb{R}^{n\times n}$ denote the matrix obtained by adding row vectors $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_k$ to $A_{F,\sigma}$ and let $A^i_{F,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k,\sigma} \in \mathbb{R}^{n\times n}$ be the matrix $A_{F,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k,\sigma} \in \mathbb{R}^{n\times n}$ with the ith column replaced by $\begin{bmatrix}\boldsymbol{b}_{F,\sigma} & \beta_1 & \cdots & \beta_k\end{bmatrix}^T$.

Corollary C.3. *Fix an IP $(\boldsymbol{c}, A, \boldsymbol{b})$. There is a set of at most K hyperplanes, $nK^n(m+2n)^n\tau^{3n}$ degree-K polynomial hypersurfaces, and $nK^n(m+2n)^{2n}\tau^{3n}$ degree-$2K$ polynomial hypersurfaces partitioning $\mathbb{R}^{K(n+1)}$ into connected components such that for each component C and every $\sigma \subseteq \mathcal{BC}$, one of the following holds: either (1) $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \le \beta_K, \sigma) = \boldsymbol{x}^*_{\mathsf{LP}}(\sigma)$, or (2) there is a subset of cuts indexed by $\ell_1, \ldots, \ell_k \in [K]$ and a set of constraints $F \subseteq M \cup \sigma$ with $|F| = n - k$ such that*

$$\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \le \beta_K, \sigma) = \left(\frac{\det(A^1_{F,\boldsymbol{\alpha}_{\ell_1},\beta_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k},\beta_{\ell_k},\sigma})}{\det(A_{F,\boldsymbol{\alpha}_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k},\sigma})}, \ldots, \frac{\det(A^n_{F,\boldsymbol{\alpha}_{\ell_1},\beta_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k},\beta_{\ell_k},\sigma})}{\det(A_{F,\boldsymbol{\alpha}_{\ell_1},\ldots,\boldsymbol{\alpha}_{\ell_k},\sigma})} \right),$$

for all $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K) \in C$.

Proof. The exact same reasoning in the proof of Lemma B.1 applies. We still have K hyperplanes. Now, for each σ, for each subset $S \subseteq K$ with $|S| \leq n$, and for every face F of $\mathcal{P}(\sigma)$ with $\dim(F) = |S|$, we have at most m degree-K polynomial hypersurfaces. The number of k-dimensional faces of $\mathcal{P}(\sigma)$ is at most $\binom{m+|\sigma|}{n-k} \leq (m+2n)^{n-1}$, so the total number of these hypersurfaces is at most $nK^n(m+2n)^n\tau^{3n}$. Finally, for every σ, we considered a degree-$2K$ polynomal hypersurfaces for every subset of cuts and every pair of faces with degree equal to the size of the subset, of which there are at most $nK^n(m+2n)^{2n}\tau^{3n}$, as desired. $\square$

We now refine the decomposition obtained in Lemma 4.1 so that the branching constraints added at each step of branch-and-cut are invariant within a region. For ease of exposition, we assume that branch-and-cut uses a lexicographic variable selection policy. This means that the variable branched on at each node of the search tree is fixed and given by the lexicographic ordering $x_1, \ldots, x_n$. Generalizing the argument to work for other policies, such as the product scoring rule, can be done as in the single-cut case.

Lemma C.4. *Suppose branch-and-cut uses a lexicographic variable selection policy. Then, there is a set of of at most K hyperplanes, $3n^2K^n(m+2n)^n\tau^{3n}$ degree-K polynomial hypersurfaces, and $nK^n(m+2n)^{2n}\tau^{3n}$ degree-$2K$ polynomial hypersurfaces partitioning $\mathbb{R}^{n+1}$ into connected components such that within each connected component, the branching constraints used at every step of branch-and-cut are invariant.*

Proof. Fix a connected component C in the decomposition established in Corollary C.3. Then, by Corollary C.3, for each σ, either $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K, \sigma) = \boldsymbol{x}^*_{\mathsf{LP}}(\sigma)$ or there exists cuts (without less of generality) labeled by indices $1, \ldots, k \in [K]$ and there exists $F \subseteq M \cup \sigma$ such that

$$\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K, \sigma)[i] = \frac{\det(A^i_{F,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k,\sigma})}{\det(A_{F,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k,\sigma})}$$

for all $(\boldsymbol{\alpha}, \beta) \in C$ and all $i \in [n]$. Now, if we are at a stage in the branch-and-cut tree where σ is the list of branching constraints added so far, and the ith variable is being branched on next, the two constraints generated are

$$x_i \leq \left\lfloor \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K, \sigma)[i] \right\rfloor \text{ and } x_i \geq \left\lceil \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K, \sigma)[i] \right\rceil,$$

respectively. If C is a component where $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \leq \beta_K, \sigma) = \boldsymbol{x}^*_{\mathsf{LP}}(\sigma)$, then there is nothing more to do, since the branching constraints at that point are trivially invariant over $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K) \in C$. Otherwise, in order to further decompose C such that the right-hand-side of these constraints are invariant for every σ and every $i = 1, \ldots, n$, we add the two decision boundaries given by

$$k \leq \frac{\det(A^i_{F,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k,\sigma})}{\det(A_{F,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k,\sigma})} \leq k+1$$

for every i, σ, and every integer $k = 0, \ldots, \tau - 1$, where $\tau = \lceil \max_{\boldsymbol{x}\in\mathcal{P}} \|\boldsymbol{x}\|_\infty \rceil$. This ensures that within every connected component of C induced by these boundaries (degree-K polynomial hypersurfaces),

$$\left\lfloor \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)[i] \right\rfloor = \left\lfloor \frac{\det(A^i_{F,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k,\sigma})}{\det(A_{F,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k,\sigma})} \right\rfloor$$

and

$$\left\lceil \boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \leq \beta, \sigma)[i] \right\rceil = \left\lceil \frac{\det(A^i_{F,\boldsymbol{\alpha}_1,\beta_1,\ldots,\boldsymbol{\alpha}_k,\beta_k,\sigma})}{\det(A_{F,\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_k,\sigma})} \right\rceil$$

are invariant, so the branching constraints added by, for example, a lexicographic branching rule, are invariant. For a fixed σ, there are two hypersurfaces for every subset $S \subseteq [K]$, every $F \subseteq M \cup \sigma$ corresponding to a $|S|$-dimensional face of $\mathcal{P}(\sigma)$, and every $i = 1, \ldots, n$, for a total of at most $2n^2K^n\binom{m+|\sigma|}{|S|} \leq 2n^2K^n(m+2n)^n$. Summing over all reduced σ, we get a total of $2n^2K^n(m+2n)^n\tau^{3n}$ hypersurfaces. Adding these hypersurfaces to the set of hypersurfaces established in Corollary C.3 yields the lemma statement. $\square$

Now, as in the single-cut case, we consider the constraints that ensure that all cuts are valid. Let $\mathcal{V} \subseteq \mathbb{R}^{K(n+1)}$ denote the set of all vectors of valid K cuts. As before, $\mathcal{V}$ is a polyhedron, since we may write

$$\mathcal{V} = \bigcap_{k=1}^{K} \bigcap_{\boldsymbol{x}_{\mathsf{IH}} \in \mathcal{P}_{\mathsf{IH}}} \left\{ (\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_k) \in \mathbb{R}^{K(n+1)} : \boldsymbol{\alpha}_k^T \boldsymbol{x}_{\mathsf{IH}} \leq \beta_k \right\}.$$

We now refine our decomposition further to control the integrality of the various LP solutions at each node of branch-and-cut.

Lemma C.5. *Given an IP $(\boldsymbol{c}, A, \boldsymbol{b})$, there is a set of at most $2K\tau^n$ hyperplanes, $4n^2K^n(m+2n)^n\tau^{4n}$ degree-K polynomial hypersurfaces, and $nK^n(m+2n)^{2n}\tau^{3n}$ degree-$2K$ polynomial hypersurfaces partitioning $\mathbb{R}^{K(n+1)}$ into connected components such that for each component C, and each $\sigma \subseteq \mathcal{BC}$,*

$$\mathbf{1}\left[\boldsymbol{x}^*_{\mathsf{LP}}\left(\boldsymbol{\alpha}_1^T \boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T \boldsymbol{x} \leq \beta_K, \sigma\right) \in \mathbb{Z}^n\right]$$

is invariant for all $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K) \in C$.

Proof. Fix a connected component C in the decomposition that includes the facets defining $\mathcal{V}$ and the surfaces obtained in Lemma C.4. For all $\sigma \in \mathcal{BC}$, $\boldsymbol{x}_{\mathsf{I}} \in \mathcal{P}_{\mathsf{I}}$, and $i = 1, \ldots, n$, consider the surface

$$\boldsymbol{x}^*_{\mathsf{LP}}\left(\boldsymbol{\alpha}_1^T \boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T \boldsymbol{x} \leq \beta_K, \sigma\right)[i] = \boldsymbol{x}_{\mathsf{I}}[i]. \tag{14}$$

This surface is a polynomial hypersurface of degree at most K, due to Corollary C.3. Clearly, within any connected component of C induced by these hyperplanes, for every σ and $\boldsymbol{x}_{\mathsf{I}} \in \mathcal{P}_{\mathsf{I}}$, $\mathbf{1}[\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T \boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T \boldsymbol{x} \leq \beta_K, \sigma) = \boldsymbol{x}_{\mathsf{I}}]$ is invariant. Finally, if $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T \boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T \boldsymbol{x} \leq \beta_K, \sigma) \in \mathbb{Z}^n$ for some K cuts $\boldsymbol{\alpha}_1^T \boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T \boldsymbol{x} \leq \beta_K$ within a given connected component, $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T \boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T \boldsymbol{x} \leq \beta_K, \sigma) = \boldsymbol{x}_{\mathsf{I}}$ for some $\boldsymbol{x}_{\mathsf{I}} \in \mathcal{P}_{\mathsf{IH}}(\sigma) \subseteq \mathcal{P}_{\mathsf{I}}$, which means that $\boldsymbol{x}^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T \boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T \boldsymbol{x} \leq \beta_K, \sigma) = \boldsymbol{x}_{\mathsf{I}} \in \mathbb{Z}^n$ *for all* vectors of K cuts $\boldsymbol{\alpha}_1^T \boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_K^T \boldsymbol{x} \leq \beta_K$ in that connected component.

We now count the number of hyperplanes given by Equation 14. For each σ, there are nK^n possible subsets of cut indices and at most $(m+2n)^{n-1}$ binding face constraints $F \subseteq M \cup \sigma$ defining the formula of Corollary C.3. For each subset-face pair, there are $n|\mathcal{P}_{\mathsf{I}}| \leq n\tau^n$ degree-K polynomial hypersurfaces given by Equation 14. So the total number of such hypersurfaces over all σ is at most $\tau^{3n}n^2K^n(m+2n)^{n-1}\tau^n$. The number of facets defining $\mathcal{V}$ is at most $K|\mathcal{P}_{\mathsf{I}}| \leq K\tau^n$. Adding these to the counts obtained in Lemma C.4 yields the final tallies in the lemma statement. □

At this point, as in the single-cut case, if the bounding aspect of branch-and-cut is suppressed, our decomposition yields connected components over which the branch-and-cut tree built is invariant. We now prove our main structural theorem for B&C as a function of multiple cutting planes at the root.

Theorem C.6. *Given an IP $(\boldsymbol{c}, A, \boldsymbol{b})$, there is a set of at most $O(12^n n^{2n} K^{2n^2}(m+2n)^{2n^2}\tau^{5n^2})$ polynomial hypersurfaces of degree at most $2K$ partitioning $\mathbb{R}^{K(n+1)}$ into connected components such that the branch-and-cut tree built after adding the K cuts $\boldsymbol{\alpha}_1^T \boldsymbol{x} \leq \beta_1, \ldots, \boldsymbol{\alpha}_k^T \boldsymbol{x} \leq \beta_k$ at the root is invariant over all $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K)$ within a given component. In particular, $f_{\boldsymbol{c},A,\boldsymbol{b}}(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K)$ is invariant over each connected component.*

Proof. Fix a connected component C in the decomposition induced by the set of hyperplanes, degree-K hypersurfaces, and degree-$2K$ hypersurfaces established in Lemma C.5. Let

$$Q_1, \ldots, Q_{i_1}, I_1, Q_{i_1+1}, \ldots, Q_{i_2}, I_2, Q_{i_2+1}, \ldots \tag{15}$$

denote the nodes of the tree branch-and-cut creates, in order of exploration, under the assumption that a node is pruned if and only if either the LP at that node is infeasible or the LP optimal solution is integral (so the "bounding" of branch-and-bound is suppressed). Here, a node is identified by the list σ of branching constraints added to the input IP. Nodes labeled by Q are either infeasible or have fractional LP optimal solutions. Nodes labeled by I have integral LP optimal solutions and are candidates for the incumbent integral solution at the point they are encountered. (The nodes are functions of $\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K$, as are the indices $i_1, i_2, \ldots$.) By Lemma C.5, this ordered list of nodes is invariant for all $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_k) \in C$.

Now, given an node index ℓ, let $I(\ell)$ denote the incumbent node with the highest objective value encountered up until the ℓth node searched by B&C, and let $z(I(\ell))$ denote its objective value. For each node Q_ℓ, let σ_ℓ denote the branching constraints added to arrive at node Q_ℓ. The hyperplane

$$z^*_{\mathsf{LP}}\left(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_K^T\boldsymbol{x} \le \beta_K, \sigma_\ell\right) = z(I(\ell)) \tag{16}$$

(which is a hyperplane due to Corollary C.3) partitions C into two subregions. In one subregion, $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_k^T\boldsymbol{x} \le \beta_k, \sigma_\ell) \le z(I(\ell))$, that is, the objective value of the LP optimal solution is no greater than the objective value of the current incumbent integer solution, and so the subtree rooted at Q_ℓ is pruned. In the other subregion, $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}_1^T\boldsymbol{x} \le \beta_1, \ldots, \boldsymbol{\alpha}_k^T\boldsymbol{x} \le \beta_k, \sigma_\ell) > z(I(\ell))$, and Q_ℓ is branched on further. Therefore, within each connected component of C induced by all hyperplanes given by Equation 16 for all ℓ, the set of node within the list (15) that are pruned is invariant. Combined with the surfaces established in Lemma C.5, these hyperplanes partition $\mathbb{R}^{K(n+1)}$ into connected components such that as $(\boldsymbol{\alpha}_1, \beta_1 \ldots, \boldsymbol{\alpha}_K, \beta_K)$ varies within a given component, the tree built by branch-and-cut is invariant.

Finally, we count the total number of surfaces inducing this partition. Unlike the counting stages of the previous lemmas, we will first have to count the number of connected components induced by the surfaces established in Lemma C.5. This is because the ordered list of nodes explored by branch-and-cut (15) can be different across each component, and the hyperplanes given by Equation 16 depend on this list. From Lemma C.5 we have $6n^2K^n(m+2n)^{2n}\tau^{4n}$ polynomial hypersurfaces of degree $\le 2K$. The set of all $(\boldsymbol{\alpha}_1, \beta_1, \ldots \boldsymbol{\alpha}_K, \beta_k) \in \mathbb{R}^{K(n+1)}$ such that $(\boldsymbol{\alpha}_1, \beta_1, \ldots, \boldsymbol{\alpha}_K, \beta_K)$ lies on the boundary of any of these surfaces is precisely the zero set of the product of all polynomials defining these surfaces. Denote this product polynomial by p. The degree of the product polynomial is the sum of the degrees of $6n^2K^n(m+2n)^{2n}\tau^{4n}$ polynomials of degree $\le 2K$, which is at most $2K \cdot 6Kn^2K^n(m+2n)^{2n}\tau^{4n} = 12n^2K^{n+2}(m+2n)^{2n}\tau^{4n}$. By Warren's theorem, the number of connected components of $\mathbb{R}^{n+1} \setminus \{(\boldsymbol{\alpha}, \beta) : p(\boldsymbol{\alpha}, \beta) = 0\}$ is $O((12n^2K^{n+2}(m+2n)^{2n}\tau^{4n})^{n-1})$, and by the Milnor-Thom theorem, the number of connected components of $\{(\boldsymbol{\alpha}, \beta) : p(\boldsymbol{\alpha}, \beta) = 0\}$ is $O((12n^2K^{n+2}(m+2n)^{2n}\tau^{4n})^{n-1})$ as well. So, the number of connected components induced by the surfaces in Lemma C.5 is $O(12^n n^{2n} K^{2n^2}(m+2n)^{2n^2}\tau^{4n^2})$. For every connected component C in Lemma C.5, the closed form of $z^*_{\mathsf{LP}}(\boldsymbol{\alpha}^T\boldsymbol{x} \le \beta, \sigma_\ell)$ is already determined due to Corollary C.3, and so the number of hyperplanes given by Equation 16 is at most the number of possible $\sigma \subseteq \mathcal{BC}$, which is at most τ^{3n}. So across all connected components C, the total number of hyperplanes given by Equation 16 is $O(12^n n^{2n} K^{2n^2}(m+2n)^{2n^2}\tau^{5n^2})$. Finally, adding this to the surface-counts established in Lemma C.5 yields the theorem statement. □

D Omitted results from Section 5

Proof of Theorem 5.1. For a set $\mathcal{X}$, $\mathcal{X}^{<\mathbb{N}}$ denotes the set of finite sequences of elements from $\mathcal{X}$. There is a bijection between the set of IPs $(\boldsymbol{c}, A, \boldsymbol{b}) \in \mathcal{I} := \mathbb{R}^n \times \mathbb{Z}^{m\times n} \times \mathbb{Z}^m$ and $\mathbb{R}$, so IPs can be uniquely represented as real numbers (and vice versa). Now, consider the set of all finite sequences of pairs of IPs and ± 1 labels of the form $((\boldsymbol{c_1}, A_1, \boldsymbol{b}_1), \varepsilon_1), \ldots, ((\boldsymbol{c_N}, A_N, \boldsymbol{b}_N), \varepsilon_N)$, $\varepsilon_1, \ldots, \varepsilon_N \in \{-1, 1\}$, that is, the set $(\mathcal{I} \times \{-1,1\})^{<\mathbb{N}}$. There is a bijection between this set and $(\mathbb{R} \times \{-1,1\})^{<\mathbb{N}}$, and in turn there is a bijection between $(\mathbb{R} \times \{-1,1\})^{<\mathbb{N}}$ and $\mathbb{R}$. Hence, there exists a bijection between $\mathcal{U}$ and $(\mathcal{I} \times \{-1,1\})^{<\mathbb{N}}$. Fix such a bijection $\varphi : \mathcal{U} \to (\mathcal{I} \times \{-1,1\})^{<\mathbb{N}}$, and let $\varphi^{-1} : (\mathcal{I} \times \{-1,1\})^{<\mathbb{N}} \to \mathcal{U}$ denote the inverse of φ, which is well defined and also a bijection.

Let n be odd. For $c \in \mathbb{R}$, let $\mathsf{IP}_c \in \mathcal{I}$ denote the IP

$$\begin{array}{ll} \text{maximize} & c \\ \text{subject to} & 2x_1 + \cdots + 2x_n = n \\ & \boldsymbol{x} \in \{0,1\}^n. \end{array} \tag{17}$$

Since n is odd, IP_c is infeasible, independent of c. Jeroslow [32] showed that without the use of cutting planes or heuristics, branch-and-bound builds a tree of size $2^{(n-1)/2}$ before determining infeasibility and terminating. The objective c is irrelevant, but is important in generating distinct IPs with this property. Consider the cut $x_1 + \cdots + x_n \le \lfloor n/2 \rfloor$, which is a valid cut for IP_c (this is in fact a Chvátal-Gomory cut [10]). In particular, since n is odd, $x_1 + \cdots + x_n \le \lfloor n/2 \rfloor \implies x_1 + \cdots + x_n \le (n-1)/2 < n/2$, so the equality constraint of IP_c is violated by this cut. Thus, the feasible region of the LP relaxation after adding this cut is empty, and branch-and-bound will terminate immediately at the root (building a tree of size 1). Denote this cut by $(\boldsymbol{\alpha}^{(-1)}, \beta^{(-1)}) = (\mathbf{1}, \lfloor n/2 \rfloor)$. On the other hand, let $(\boldsymbol{\alpha}^{(1)}, \beta^{(1)}) = (\mathbf{0}, 0)$ be the trivial cut $0 \leq 0$. Adding this cut to the IP constraints does not change the feasible region, so branch-and-bound will build a tree of size $2^{(n-1)/2}$.

We now define $\boldsymbol{\alpha}_{\boldsymbol{c},A,\boldsymbol{b}}$ and $\beta_{\boldsymbol{c},A,\boldsymbol{b}}$. Let

$$(\boldsymbol{\alpha}_{\boldsymbol{c},A,\boldsymbol{b}}(\boldsymbol{u}), \beta_{\boldsymbol{c},A,\boldsymbol{b}}(\boldsymbol{u})) = \begin{cases} (\boldsymbol{\alpha}^{(1)}, \beta^{(1)}) & \text{if } ((\boldsymbol{c}, A, \boldsymbol{b}), 1) \in \varphi(\boldsymbol{u}) \text{ and } ((\boldsymbol{c}, A, \boldsymbol{b}), -1) \notin \varphi(\boldsymbol{u}) \\ (\boldsymbol{\alpha}^{(-1)}, \beta^{(-1)}) & \text{if } ((\boldsymbol{c}, A, \boldsymbol{b}), -1) \in \varphi(\boldsymbol{u}) \text{ and } ((\boldsymbol{c}, A, \boldsymbol{b}), 1) \notin \varphi(\boldsymbol{u}) \\ (\mathbf{0}, 0) & \text{otherwise} \end{cases}.$$

The choice to use $(\mathbf{0}, 0)$ in the case that either $((\boldsymbol{c}, A, \boldsymbol{b}), \varepsilon) \notin \varphi(\boldsymbol{u})$ for each $\varepsilon \in \{-1, 1\}$, or $((\boldsymbol{c}, A, \boldsymbol{b}), -1) \in \varphi(\boldsymbol{u})$ and $((\boldsymbol{c}, A, \boldsymbol{b}), 1) \in \varphi(\boldsymbol{u})$ is arbitrary and unimportant. Now, for any integer $N > 0$, constructing a set of N IPs and N thresholds that is shattered is almost immediate. Let $c_1, \ldots, c_N \in \mathbb{R}$ be distinct reals, and let $1 < r_1, \ldots, r_N < 2^{(n-1)/2}$. Then, the set $\{(\mathsf{IP}_{c_1}, r_1), \ldots, (\mathsf{IP}_{c_N}, r_N)\}$ can be shattered. Indeed, given a sign pattern $(\varepsilon_1, \ldots, \varepsilon_N) \in \{-1, 1\}^N$, let

$$\boldsymbol{u} = \varphi^{-1}\left((\mathsf{IP}_{c_1}, \varepsilon_1), \ldots, (\mathsf{IP}_{c_N}, \varepsilon_N)\right).$$

Then, if $\varepsilon_i = 1$, $(\boldsymbol{\alpha}_{\mathsf{IP}_{c_i}}(\boldsymbol{u}), \beta_{\mathsf{IP}_{c_i}}(\boldsymbol{u})) = (\boldsymbol{\alpha}^{(1)}, \beta^{(1)})$, so $g_{\boldsymbol{u}}(\mathsf{IP}_{c_i}) = 2^{(n-1)/2}$ and $\operatorname{sign}(g_{\boldsymbol{u}}(\mathsf{IP}_{c_i}) - r_i) = 1$. If $\varepsilon_i = -1$, $(\boldsymbol{\alpha}_{\mathsf{IP}_{c_i}}(\boldsymbol{u}), \beta_{\mathsf{IP}_{c_i}}(\boldsymbol{u})) = (\boldsymbol{\alpha}^{(-1)}, \beta^{(-1)})$, so $g_{\boldsymbol{u}}(\mathsf{IP}_{c_i}) = 1$ and $\operatorname{sign}(g_{\boldsymbol{u}}(\mathsf{IP}_{c_i}) - r_i) = -1$. So for any N there is a set of IPs and thresholds that can be shattered, which yields the theorem statement. $\square$

Lemma D.1. *Consider the family of GMI cuts parameterized by $\boldsymbol{u} \in [-U, U]^m$. There is a set of at most $O(nU^2 \|A\|_1 \|\boldsymbol{b}\|_1)$ hyperplanes partitioning $[-U, U]^m$ into connected components such that $\lfloor \boldsymbol{u}^T \boldsymbol{a}_i \rfloor$, $\lfloor \boldsymbol{u}^T \boldsymbol{b} \rfloor$, and $\mathbf{1}[f_i \leq f_0]$ are invariant, for every i, within each component.*

Proof of Lemma D.1. We have $f_i = \boldsymbol{u}^T \boldsymbol{a}_i - \lfloor \boldsymbol{u}^T \boldsymbol{a}_i \rfloor$, $f_0 = \boldsymbol{u}^T \boldsymbol{b} - \lfloor \boldsymbol{u}^T \boldsymbol{b} \rfloor$, and since $\boldsymbol{u} \in [-U, U]^m$, $\lfloor \boldsymbol{u}^T \boldsymbol{a}_i \rfloor \in [-U \|\boldsymbol{a}_i\|_1, U \|\boldsymbol{a}_i\|_1]$ and $\lfloor \boldsymbol{u}^T \boldsymbol{b} \rfloor \in [-U \|\boldsymbol{b}\|_1, U \|\boldsymbol{b}\|_1]$. Now, for all i, $k_i \in [-U \|\boldsymbol{a}_i\|_1, U \|\boldsymbol{a}_i\|_1] \cap \mathbb{Z}$ and $k_0 \in [-U \|\boldsymbol{b}\|_1, U \|\boldsymbol{b}\|_1] \cap \mathbb{Z}$, put down the hyperplanes defining the two halfspaces

$$\lfloor \boldsymbol{u}^T \boldsymbol{a}_i \rfloor = k_i \iff k_i \leq \boldsymbol{u}^T \boldsymbol{a}_i < k_i + 1 \tag{18}$$

and the hyperplanes defining the two halfspaces

$$\lfloor \boldsymbol{u}^T \boldsymbol{b} \rfloor = k_0 \iff k_0 \leq \boldsymbol{u}^T \boldsymbol{b} < k_0 + 1. \tag{19}$$

In addition, consider the hyperplane

$$\boldsymbol{u}^T \boldsymbol{a}_i - k_i = \boldsymbol{u}^T \boldsymbol{b} - k_0 \tag{20}$$

for each i. Within any connected component of $\mathbb{R}^m$ determined by these hyperplanes, $\lfloor \boldsymbol{u}^T \boldsymbol{a}_i \rfloor$ and $\lfloor \boldsymbol{u}^T \boldsymbol{b} \rfloor$ are constant. Furthermore, $\mathbf{1}[f_i \leq f_0]$ is invariant within each connected component, since if $\lfloor \boldsymbol{u}^T \boldsymbol{a}_i \rfloor = k_i$ and $\lfloor \boldsymbol{u}^T \boldsymbol{b} \rfloor = k_0$, $f_i \leq f_0 \iff \boldsymbol{u}^T \boldsymbol{a}_i - k_i \leq \boldsymbol{u}^T \boldsymbol{b} - k_0$, which is the hyperplane given by Equation 20. The total number of hyperplanes of type 18 is $O(nU \|A\|_1)$, the total number of hyperplanes of type 19 is $O(U \|\boldsymbol{b}\|_1)$, and the total number of hyperplanes of type 20 is $nU^2 \|A\|_1 \|\boldsymbol{b}\|_1$. Summing yields the lemma statement. $\square$

The next lemma allows us to transfer the polynomial partition of $\mathbb{R}^{n+1}$ from Theorem 4.4 to a polynomial partition of $[-U, U]^m$, incurring only a factor 2 increase in degree.

Lemma D.2. *Let $p \in \mathbb{R}[y_1, \ldots, y_{n+1}]$ be a polynomial of degree d. Let $D \subseteq [-U, U]^m$ be a connected component from Lemma D.1. Define $q : D \to \mathbb{R}$ by $q(\boldsymbol{u}) = p(\boldsymbol{\alpha}(\boldsymbol{u}), \beta(\boldsymbol{u}))$. Then q is a polynomial in $\boldsymbol{u}$ of degree $2d$.*

Proof. By Lemma D.1, there are integers k_0, k_i for $i \in [n]$ such that $\lfloor \boldsymbol{u}^T \boldsymbol{a}_i \rfloor = k_i$ and $\lfloor \boldsymbol{u}^T \boldsymbol{b} \rfloor = k_0$ for all $\boldsymbol{u} \in D$. Also, the set $S = \{i : f_i \leq f_0\}$ is fixed over all $\boldsymbol{u} \in D$.

A degree-d polynomial p in variables $y_1, \dots, y_{n+1}$ can be written as $\sum_{T \sqsubseteq [n+1], |T| \le d} \lambda_T \prod_{i \in T} y_i$ for some coefficients $\lambda_T \in \mathbb{R}$, where $T \sqsubseteq [n+1]$ means that T is a multiset of $[n+1]$. Evaluating at $(\boldsymbol{\alpha}(\boldsymbol{u}), \beta(\boldsymbol{u}))$, we get

$$\sum_{|T| \le d} \lambda_T \prod_{\substack{i \in T \cap S \\ i \neq n+1}} f_i(1-f_0) \prod_{\substack{i \in T \setminus S \\ i \neq n+1}} f_0(1-f_i) \prod_{\substack{i \in T \\ i = n+1}} f_0(1-f_0).$$

Now, $f_i = \boldsymbol{u}^T \boldsymbol{a}_i - k_i$ and $f_0 = \boldsymbol{u}^T \boldsymbol{b} - k_0$ are linear in $\boldsymbol{u}$. The sum is over all multisets of size at most d, so each monomial consists of the product of at most d degree-2 terms of the form $f_i(1-f_0)$, $f_0(1-f_i)$, or $f_0(1-f_0)$. Thus, $\deg(q) \le 2d$, as desired. □

Proof of Lemma 5.2. Let $C \subseteq \mathbb{R}^{n+1}$ be a connected component in the partition established in Theorem 4.4, so C can be written as the intersection of at most $14^n(m+2n)^{3n^2}\tau^{5n^2}$ polynomial constraints of degree at most 5. Let $D \subseteq [-U, U]^m$ be a connected component in the partition established in Lemma D.1. By Lemma D.2, there are at most $14^n(m+2n)^{3n^2}\tau^{5n^2}$ polynomials of degree at most 10 partitioning D into connected components such that within each component, $\mathbf{1}[(\boldsymbol{\alpha}(\boldsymbol{u}), \beta(\boldsymbol{u})) \in C]$ is invariant. If we consider the overlay of these polynomial surfaces over all components C, we will get a partition of $[-U, U]^m$ such that *for every* C, $\mathbf{1}[(\boldsymbol{\alpha}(\boldsymbol{u}), \beta(\boldsymbol{u})) \in C]$ is invariant over each connected component of $[-U, U]^m$. Once we have this we are done, since all $\boldsymbol{u}$ in the same connected component of $[-U, U]^m$ will be sent to the same connected component of $\mathbb{R}^{n+1}$ by $(\boldsymbol{\alpha}(\boldsymbol{u}), \beta(\boldsymbol{u}))$, and thus by Theorem 4.4 the behavior of branch-and-cut will be invariant.

We now tally up the total number of surfaces. The number of connected components C was given by Warren's theorem and the Milnor-Thom theorem to be $O(14^{n(n+1)}(m+2n)^{3n^2(n+1)}\tau^{5n^2(n+1)})$, so the total number of degree-10 hypersurfaces is $14^n(m+2n)^{3n^2}\tau^{5n^2}$ times this quantity, which yields the lemma statement. □

D.1 Multiple GMI cuts at the root

In this section we extend our results to allow for multiple GMI cuts at the root of the B&C tree. These cuts can be added simultaneously, sequentially, or in rounds. If GMI cuts $\boldsymbol{u}_1$, $\boldsymbol{u}_2$ are added simultaneously, both of them have the same dimension and are defined in the usual way. If GMI cuts $\boldsymbol{u}_1$, $\boldsymbol{u}_2$ are added sequentially, $\boldsymbol{u}_2$ has one more entry than $\boldsymbol{u}_1$. This is because when cuts are added sequentially, the LP relaxation is re-solved after the addition of the first cut, and the second cut has a multiplier for all original constraints as well as for the first cut (this ensures that the second cut can be chosen in a more informed manner). If K cuts are made at the root, they can be added in sequential rounds of simultaneous cuts. In the following discussion, we focus on the case where all K cuts are added sequentially—the other cases can be viewed as instantiations of this. We refer the reader to the discussion in Balcan et al. [10] for more details.

To prove an analogous result for multiple GMI cuts (in sequence, that is, each successive GMI cut has one more parameter than the previous), we combine the reasoning used in the single-GMI-cut case with some technical observations in Balcan et al. [10].

Lemma D.3. *Consider the family of K sequential GMI cuts parameterized by $\boldsymbol{u}_1 \in [-U, U]^m, \boldsymbol{u}_2 \in [-U, U]^{m+1}, \dots, \boldsymbol{u}_K \in [-U, U]^{m+K-1}$. For any IP $(\boldsymbol{c}, A, \boldsymbol{b})$, there are at most*

$$O\left(nK(1+U)^{2K} \|A\|_1 \|\boldsymbol{b}\|_1\right)$$

degree-K polynomial hypersurfaces and

$$2^{O(n^2)} K^{O(n^3)} (m+2n)^{O(n^3)} \tau^{O(n^3)}$$

degree-$4K^2$ polynomial hypersurfaces partitioning $[-U, U]^m \times \cdots \times [-U, U]^{m+K-1}$ connected components such that the B&C tree built after sequentially adding the GMI cuts defined by $\boldsymbol{u}_1, \dots, \boldsymbol{u}_K$ is invariant over all $(\boldsymbol{u}_1, \dots, \boldsymbol{u}_K)$ within a single component.

Proof. We start with the setup used by Balcan et al. [10] to prove similar results for sequential Chvátal-Gomory cuts. Let $\boldsymbol{a}_1, \dots, \boldsymbol{a}_n \in \mathbb{R}^m$ be the columns of A. We define the following augmented columns $\widetilde{\boldsymbol{a}}_i^1 \in \mathbb{R}^m, \ldots, \widetilde{\boldsymbol{a}}_i^K \in \mathbb{R}^{m+K-1}$ for each $i \in [n]$, and the augmented constraint vectors $\widetilde{\boldsymbol{b}}^1 \in \mathbb{R}^m, \ldots, \widetilde{\boldsymbol{b}}^K \in \mathbb{R}^{m+K-1}$ via the following recurrences:

$$\widetilde{\boldsymbol{a}}_i^1 = \boldsymbol{a}_i$$
$$\widetilde{\boldsymbol{a}}_i^k = \begin{bmatrix} \widetilde{\boldsymbol{a}}_i^{k-1} \\ \boldsymbol{u}_{k-1}^T \widetilde{\boldsymbol{a}}_i^{k-1} \end{bmatrix}$$

and

$$\widetilde{\boldsymbol{b}}^1 = \boldsymbol{b}$$
$$\widetilde{\boldsymbol{b}}^k = \begin{bmatrix} \widetilde{\boldsymbol{b}}^{k-1} \\ \boldsymbol{u}_{k-1}^T \widetilde{\boldsymbol{b}}^{k-1} \end{bmatrix}$$

for $k = 2, \ldots, K$. In other words, $\widetilde{\boldsymbol{a}}_i^k$ is the ith column of the constraint matrix of the IP and $\widetilde{\boldsymbol{b}}^k$ is the constraint vector after applying cuts $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{k-1}$. An identical induction argument to that of Balcan et al. [10] shows that for each $k \in [K]$,

$$\left\lfloor \boldsymbol{u}_k^T \widetilde{\boldsymbol{a}}_i^k \right\rfloor \in \left[-(1+U)^k \|\boldsymbol{a}_i\|_1, (1+U)^k \|\boldsymbol{a}_i\|_1 \right]$$

and

$$\left\lfloor \boldsymbol{u}_k^T \widetilde{\boldsymbol{b}}^k \right\rfloor \in \left[-(1+U)^k \|\boldsymbol{b}\|_1, (1+U)^k \|\boldsymbol{b}\|_1 \right].$$

Now, as in the single-GMI-cut setting, consider the surfaces

$$\left\lfloor \boldsymbol{u}_k^T \widetilde{\boldsymbol{a}}_i^k \right\rfloor = \ell_i \iff \ell_i \le \boldsymbol{u}_k^T \widetilde{\boldsymbol{a}}_i^k < \ell_i + 1 \tag{21}$$

and

$$\left\lfloor \boldsymbol{u}_k^T \widetilde{\boldsymbol{b}}^k \right\rfloor = \ell_0 \iff \ell_i \le \boldsymbol{u}_k^T \widetilde{\boldsymbol{b}}^k < \ell_0 + 1 \tag{22}$$

for every i, k, and every integer $\ell_i \in [-(1+U)^k \|\boldsymbol{a}_i\|_1, (1+U)^k \|\boldsymbol{a}_i\|_1] \cap \mathbb{Z}$ and every integer $\ell_0 \in [-(1+U)^k \|\boldsymbol{b}\|_1, (1+U)^k \|\boldsymbol{b}\|_1] \cap \mathbb{Z}$. In addition, consider the surfaces

$$\boldsymbol{u}_k^T \widetilde{\boldsymbol{a}}_i^k - \ell_i = \boldsymbol{u}_k^T \widetilde{\boldsymbol{b}}^k - \ell_0 \tag{23}$$

for each i, k, ℓ_i, ℓ_0. As observed by Balcan et al. [10], $\boldsymbol{u}_k^T \widetilde{\boldsymbol{a}}_i^k$ is a polynomial in $\boldsymbol{u}_1[1], \ldots, \boldsymbol{u}_1[m], \boldsymbol{u}_2[1], \ldots, \boldsymbol{u}_2[m+1], \ldots, \boldsymbol{u}_k[1], \ldots, \boldsymbol{u}_k[m+k-1]$ of degree at most k (as is $\boldsymbol{u}_k^T \widetilde{\boldsymbol{b}}^k$), so surfaces 21, 22, and 23 are all degree-K polynomial hypersurfaces for all i, k. Within any connected component of $[-U, U]^m \times \cdots \times [-U, U]^{m+K-1}$ induced by these hypersurfaces, $\lfloor \boldsymbol{u}_k^T \widetilde{\boldsymbol{a}}_i^k \rfloor$ and $\lfloor \boldsymbol{u}_k^T \widetilde{\boldsymbol{b}}^k \rfloor$ are constant. Furthermore $\mathbf{1}[f_i^k \le f_0^k]$ is invariant for every i, k, where $f_i^k = \boldsymbol{u}_k^T \widetilde{\boldsymbol{a}}_i^k - \lfloor \boldsymbol{u}_k^T \widetilde{\boldsymbol{a}}_i^k \rfloor$ and $f_0^k = \boldsymbol{u}_k^T \widetilde{\boldsymbol{b}}^k - \lfloor \boldsymbol{u}_k^T \widetilde{\boldsymbol{b}}^k \rfloor$.

Now, fix a connected component $D \subseteq [-U, U]^m \times \cdots \times [-U, U]^{m+K-1}$ induced by the above hypersurfaces, and let $C \subseteq \mathbb{R}^{K(n+1)}$ be the intersection of q polynomial inequalities of degree at most d. Consider a single degree-d polynomial inequality in $K(n+1)$ variables $y_1, \ldots, y_{K(n+1)}$, which can be written as

$$\sum_{\substack{T \sqsubseteq [K(n+1)] \\ |T| \le d}} \lambda_T \prod_{j \in T} y_j = \sum_{\substack{T_1, \ldots, T_K \sqsubseteq [n+1] \\ |T_1| + \cdots + |T_K| \le d}} \lambda_{T_1, \ldots, T_K} \prod_{j_1 \in T_1} y_{j_1} \cdots \prod_{j_K \in T_K} y_{j_K} \le \gamma.$$

Now, the sets $S_1, \ldots, S_K$ defined by $S_k = \{i : f_i^k \le f_0^k\}$ are fixed within D, so we can write this as

$$\sum_{\substack{T_1, \ldots, T_K \sqsubseteq [n+1] \\ |T_1| + \cdots + |T_K| \le d}} \lambda_{T_1, \ldots, T_K} \prod_{k=1}^{K} \left[\prod_{\substack{j \in T_k \cap S_k \\ j \ne n+1}} f_j^k (1 - f_0^k) \prod_{\substack{j \in T_k \setminus S_k \\ j \ne n+1}} f_0^k (1 - f_j^k) \prod_{\substack{j \in T_k \\ j = n+1}} f_0^k (1 - f_0^k) \right] \le \gamma.$$

We have that f_j^k and f_0^k are degree-k polynomials in $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k$. Since the sum is over all multisets $T_1, \ldots, T_K$ such that $|T_1| + \cdots + |T_K| \le d$, there are at most d terms across the products, each of the form $f_j^k (1 - f_0)^k$, $f_0^k (1 - f_j^k)$, or $f_0^k (1 - f_0)^k$. Therefore, the left-hand-side is a polynomial of degree at most $2dK$, and if $C \subseteq \mathbb{R}^{K(n+1)}$ is the intersection of q polynomial inequalities each of degree at most d, the set

$$\{(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K) \in D : (\boldsymbol{\alpha}(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K), \beta(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K)) \in C\} \subseteq [-U, U]^m \times \cdots \times [-U, U]^{m+K-1}$$

can be expressed as the intersection of q degree-$2dK$ polynomial inequalities.

To finish, we run this process for every connected component $C \subseteq \mathbb{R}^{K(n+1)}$ in the partition established by Theorem C.6. This partition consists of $O(12^n n^{2n} K^{2n^2}(m+2n)^{2n^2}\tau^{5n^2})$ degree-$2K$ polynomials over $\mathbb{R}^{K(n+1)}$. By Warren's theorem and the Milnor-Thom theorem, these polynomials partition $\mathbb{R}^{K(n+1)}$ into $O(12^{n(n+1)} n^{2n(n+1)} K^{2n^2(n+1)}(m+2n)^{2n^2(n+1)}\tau^{5n^2(n+1)})$ connected components. Running the above argument for each of these connected components of $\mathbb{R}^{K(n+1)}$ yields a total of $O\left(12^{n(n+1)} n^{2n(n+1)} K^{2n^2(n+1)}(m+2n)^{2n^2(n+1)}\tau^{5n^2(n+1)}\right) \cdot O\left(12^n n^{2n} K^{2n^2}(m+2n)^{2n^2}\tau^{5n^2}\right) = 2^{O(n^2)} K^{O(n^3)}(m+2n)^{O(n^3)}\tau^{O(n^3)}$ polynomials of degree $4K^2$. Finally, we count the surfaces of the form (21), (22), and (23). The total number of degree-K polynomials of type 21 is at most $O(nK(1+U)^K \|A\|_1)$, the total number of degree-k polynomials of type 22 is $O(K(1+U)^K \|\boldsymbol{b}\|_1)$, and the total number of degree-K polynomials of type 23 is $O(nK(1+U)^{2K} \|A\|_1 \|\boldsymbol{b}\|)$. Summing these counts yields the desired number of surfaces in the lemma statement.

In any connected component of $[-U, U]^m$ determined by these surfaces, $\mathbf{1}[(\boldsymbol{\alpha}(\boldsymbol{u}), \beta(\boldsymbol{u})) \in C]$ is invariant for every connected component $C \subseteq \mathbb{R}^{K(n+1)}$ in the partition of $\mathbb{R}^{K(n+1)}$ established in Theorem C.6. This means that the tree built by branch-and-cut is invariant, which concludes the proof. $\square$

Finally, applying the main result of Balcan et al. [9] to Lemma D.3, we get the following pseudo-dimension bound for the class of K sequential GMI cuts at the root of the B&C tree.

Theorem D.4. *For $\boldsymbol{u}_1 \in [-U, U]^m, \boldsymbol{u}_2 \in [-U, U]^{m+1}, \ldots, \boldsymbol{u}_K \in [-U, U]^{m+K-1}$, let $g_{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K}(\boldsymbol{c}, A, \boldsymbol{b})$ denote the number of nodes in the tree B&C builds given the input $(\boldsymbol{c}, A, \boldsymbol{b})$ after sequentially applying the GMI cuts defined by $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K$ at the root. The pseudo-dimension of the set of functions $\{g_{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K} : (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_K) \in [-U, U]^m \times \cdots \times [-U, U]^{m+K-1}\}$ on the domain of IPs with $\|A\|_1 \leq a$ and $\|\boldsymbol{b}\|_1 \leq b$ is*

$$O\left(mK^3 \log U + mn^3 K^2 \log(mnK\tau) + mK^2 \log(ab)\right).$$