# Sample Complexity of Tree Search Configuration: Cutting Planes and Beyond

Nina Balcan, Siddharth Prasad, Tuomas Sandholm, Ellen Vitercik

## Integer programs (IPs)

$$\begin{aligned}\text{max}\quad & \boldsymbol{c}\cdot\boldsymbol{x}\\ \text{s.t.}\quad & A\boldsymbol{x}\le\boldsymbol{b}\\ & \boldsymbol{x}\in\mathbb{Z}^n\end{aligned}$$


Routing

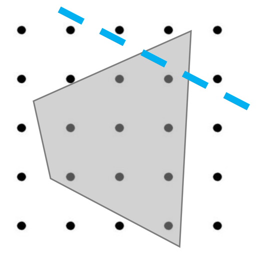

Scheduling


Planning


Finance

One of the most **useful**, **widely applicable** optimization techniques

*Cutting planes:*
Responsible for recent breakthrough speedups of IP solvers



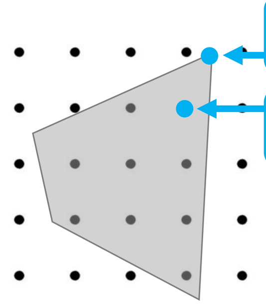

**Our contribution:**
First formal theory for using ML to select cutting planes
Sample complexity bounds

## Branch-and-bound (B&B)

B&B uses LP relaxation to do informed search through feasible set

**Linear program (LP)**

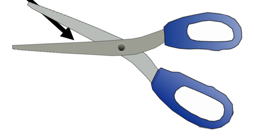
$$\begin{aligned}\text{max}\quad & \boldsymbol{c}\cdot\boldsymbol{x}\\ \text{s.t.}\quad & A\boldsymbol{x}\le\boldsymbol{b}\end{aligned}$$
~~$\boldsymbol{x}\in\mathbb{Z}^n$~~




Choose variable $i$ to branch on: add constraints $\boldsymbol{x}[i]\le\lfloor\boldsymbol{x}^*_{\text{LP}}[i]\rfloor$, $\boldsymbol{x}[i]\ge\lceil\boldsymbol{x}^*_{\text{LP}}[i]\rceil$

$$\begin{aligned}\text{max}\quad & \boldsymbol{c}\cdot\boldsymbol{x}\\ \text{s.t.}\quad & A\boldsymbol{x}\le\boldsymbol{b}\\ & \boldsymbol{x}\in\mathbb{Z}^n\end{aligned}$$

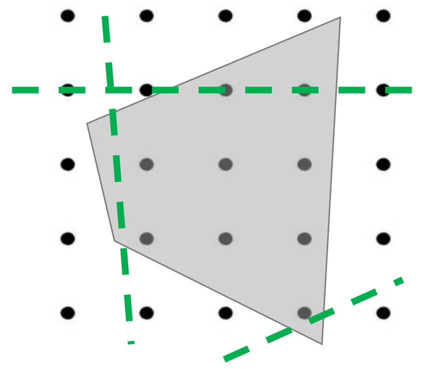
$$\begin{aligned}\text{max}\quad & \boldsymbol{c}\cdot\boldsymbol{x}\\ \text{s.t.}\quad & A\boldsymbol{x}\le\boldsymbol{b}\\ & \boldsymbol{x}[i]\le 2\\ & \boldsymbol{x}\in\mathbb{Z}^n\end{aligned}\qquad\qquad \begin{aligned}\text{max}\quad & \boldsymbol{c}\cdot\boldsymbol{x}\\ \text{s.t.}\quad & A\boldsymbol{x}\le\boldsymbol{b}\\ & \boldsymbol{x}[i]\ge 3\\ & \boldsymbol{x}\in\mathbb{Z}^n\end{aligned}$$

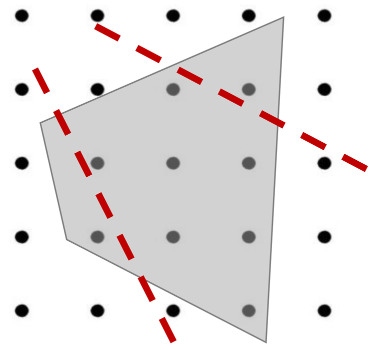


Prune subtrees if LP relaxation is:
- Integral, or
- Worse than best integral solution found so far

## Cutting planes

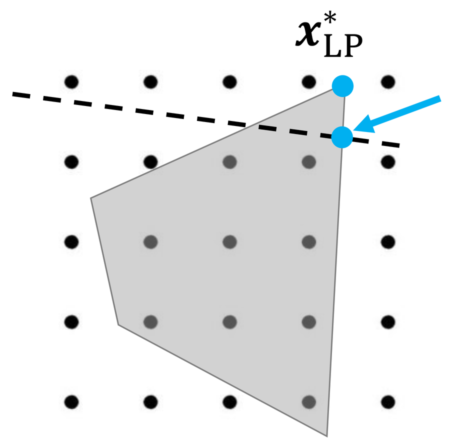
Constraint $\boldsymbol{\alpha}^T\boldsymbol{x}\le\beta$ that doesn't cut off any integer feasible points


Valid cuts


Invalid cuts

Branch-and-cut (B&C): Cuts added at any node of the search tree

Tightens LP relaxation to prune nodes sooner




We study *Chvátal-Gomory (CG) cuts:* $\lfloor\boldsymbol{u}^T A\rfloor\boldsymbol{x}\le\lfloor\boldsymbol{u}^T\boldsymbol{b}\rfloor$ for $\boldsymbol{u}\in[0,1)^m$

## Learning to cut

Best cutting planes for **routing** problems
likely not suited for **scheduling**

Application domain modeled by distribution over IPs

**Key question: Sample complexity**
If CG cut yields small B&C tree size on average over a training set…

**Training set**

$$\begin{aligned}\text{max}\quad & \boldsymbol{c}_1\cdot\boldsymbol{x}\\ \text{s.t.}\quad & A_1\boldsymbol{x}\le\boldsymbol{b}_1\\ & \boldsymbol{x}\in\mathbb{Z}^n\end{aligned}\quad\cdots\quad\begin{aligned}\text{max}\quad & \boldsymbol{c}_N\cdot\boldsymbol{x}\\ \text{s.t.}\quad & A_N\boldsymbol{x}\le\boldsymbol{b}_N\\ & \boldsymbol{x}\in\mathbb{Z}^n\end{aligned}$$

**Fresh sample**

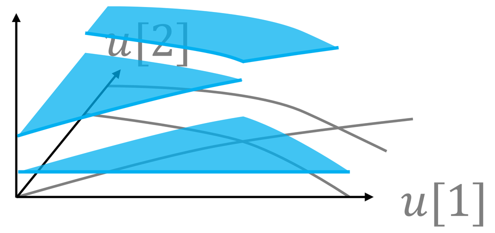
$$\begin{aligned}\text{max}\quad & \boldsymbol{c}\cdot\boldsymbol{x}\\ \text{s.t.}\quad & A\boldsymbol{x}\le\boldsymbol{b}\\ & \boldsymbol{x}\in\mathbb{Z}^n\end{aligned}$$

…will it yield a small B&C tree on a fresh IP?

Sample complexity quantified by ***pseudo-dimension***
Generalization of VC dimension

## Results

**Waves of cuts:** Solvers usually add several cuts in *waves*
- Wave 1: Add cuts $\boldsymbol{u}_1^1,\dots,\boldsymbol{u}_1^k$
- …
- Wave w: Add cuts $\boldsymbol{u}_w^1,\dots,\boldsymbol{u}_w^k$



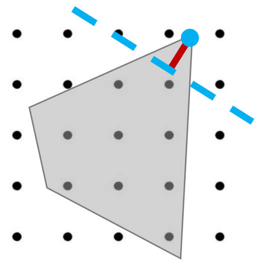

**Main challenge:** Tree-size is a complex function of $\boldsymbol{u}$
*But* it is **piecewise constant**

**Theorem:** For any IP,
$O\left(kw2^{kw}\|A\|_{1,1}+2^{kw}\|\boldsymbol{b}\|_1+kwn\right)$ polynomials partition parameters s.t.:
In each region, B&C builds the same tree

Implies **pseudo-dimension** bound $\tilde{O}(mk^2w^2\log(\alpha+\beta+n))$
for IPs with $\|A\|_{1,1}\le\alpha$, $\|\boldsymbol{b}\|_1\le\beta$

### Cut selection policies

Solvers often use *scoring rules* to choose from a pool of cuts



E.g., $\text{score}(\boldsymbol{\alpha}^T\boldsymbol{x}\le\beta)$ = distance between cut and $\boldsymbol{x}^*_{\text{LP}}$

Given $d$ scoring rules, learn mixture $\mu_1\text{score}_1+\cdots+\mu_d\text{score}_d$

**Theorem:** Class of tree-size functions parameterized by $\boldsymbol{\mu}$ has pseudo-dim
$\tilde{O}(dmw^2\log(\alpha+\beta+n))$, where $w$ = # of sequential CG cuts

### General tree search

Model captures branching, cutting planes, node selection *simultaneously*

**Theorem:**
- $t$ types of actions
- $T_j$ actions of type $j$
- Chosen according to mixtures of d scoring rules for every type:
  Pseudo-dim = $O\left(d\kappa\sum_{j=1}^{t}\log T_j+d\log d\right)$.