Two Applications of Disjunctive Programming

Daniel Bienstock
Columbia University
CMU April 2018
Application 1: set covering

\[ \min c^T x \]
\[ \text{s.t. } Ax \geq e, \quad x \text{ binary} \]

Starting point: Balas and Ng (1989), all facets with coefficients 0,1,2 → There are examples with exponentially many such facets. Can we account for all valid inequalities with small coefficients?
Application 1: set covering

\[ \begin{align*} 
\text{min} & \quad c^T x \\
\text{s.t.} & \quad Ax \geq e, \quad x \text{ binary} \\
\end{align*} \]

\( A \) is a 0/1 matrix, \( e = (1, \ldots, 1)^T \)
Application 1: set covering

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad Ax \geq e, \quad x \text{ binary}
\end{align*}
\]

A is a 0/1 matrix, \( e = (1, \ldots, 1)^T \)

**Starting point:** Balas and Ng (1989), *All facets with coefficients 0,1,2*
Application 1: set covering

$$\begin{align*}
\text{min } & \quad c^T x \\
\text{s.t. } & \quad Ax \geq e, \quad x \text{ binary}
\end{align*}$$

$A$ is a 0/1 matrix, $e = (1, \ldots, 1)^T$

**Starting point:** Balas and Ng (1989), *All facets with coefficients 0,1,2*

→ There are examples with exponentially many such facets
Application 1: set covering

\[ \begin{align*}
\text{min} & \quad c^T x \\
\text{s.t.} & \quad Ax \geq e, \quad x \text{ binary}
\end{align*} \]

\(A\) is a 0/1 matrix, \(e = (1, \ldots, 1)^T\)

**Starting point:** Balas and Ng (1989), *All facets with coefficients 0,1,2*

\[ \rightarrow \text{There are examples with exponentially many such facets} \]

Can we account for all valid inequalities with small coefficients?
**Theorem** (B. and Mark Zuckerberg, 2004)

For any fixed integer $k \geq 1$ there exists a *compact, extended* formulation whose solutions satisfy all valid inequalities with coefficients in $\{0, 1, \ldots, k\}$.

*"compact:"* of polynomial size (for fixed $k$)

*"extended:"* uses additional variables, a *lifted* formulation
**Theorem** (B. and Mark Zuckerberg, 2004)

For any fixed integer $k \geq 1$ there exists a *compact, extended* formulation whose solutions satisfy all valid inequalities with coefficients in $\{0, 1, \ldots, k\}$.

*“compact:”* of polynomial size (for fixed $k$)

*“extended:”* uses additional variables, a *lifted* formulation

**Definition:** An inequality $\alpha^T x \geq b$ for *valid* has *pitch* $\leq k$ if:

the sum of the smallest $k$ positive $\alpha_j$ is at least $b$
**Theorem** (B. and Mark Zuckerberg, 2004)

For any fixed integer $k \geq 1$ there exists a *compact*, *extended* formulation whose solutions satisfy all valid inequalities with coefficients in $\{0, 1, \ldots, k\}$.

“**compact:**” of polynomial size (for fixed $k$)

“**extended:**” uses additional variables, a *lifted* formulation

**Definition:** An inequality $\alpha^T x \geq b$ for valid has **pitch** $\leq k$ if:

the sum of the smallest $k$ positive $\alpha_j$ is at least $b$

Hence, inequalities with coefficients in $\{0, 1, \ldots, k\}$ have **pitch** $\leq k$
**Theorem** (B. and Mark Zuckerberg, 2004)

For any fixed integer $k \geq 1$ there exists a *compact*, *extended* formulation whose solutions satisfy all valid inequalities with pitch $\leq k$.

“*compact:*” of polynomial size (for fixed $k$)

“*extended:*” uses additional variables, a *lifted* formulation
Theorem (B. and Mark Zuckerberg, 2004)

For any fixed integer $k \geq 1$ there exists a compact, extended formulation whose solutions satisfy all valid inequalities with pitch $\leq k$.

“compact:” of polynomial size (for fixed $k$)

“extended:” uses additional variables, a lifted formulation

Definition: An inequality $\alpha^T x \geq b$ for valid has pitch $\leq k$ if:

the sum of the smallest $k$ positive $\alpha_j$ is at least $b$
Theorem (B. and Mark Zuckerberg, 2004)

For any fixed integer \( k \geq 1 \) there exists a compact, extended formulation whose solutions satisfy all valid inequalities with pitch \( \leq k \).

“compact:” of polynomial size (for fixed \( k \))

“extended:” uses additional variables, a lifted formulation

Definition: An inequality \( \alpha^T x \geq b \) for valid has pitch \( \leq k \) if:

the sum of the smallest \( k \) positive \( \alpha_j \) is at least \( b \)

Hence, inequalities with coefficients in \( \{0, 1, \ldots, k\} \) have pitch \( \leq k \)
**Theorem** (B. and Mark Zuckerberg, 2004)

For any fixed integer \( k \geq 1 \) there exists a *compact, extended* formulation whose solutions satisfy all valid inequalities with pitch \( \leq k \).

“**compact:**” of polynomial size (for fixed \( k \))

“**extended:**” uses additional variables, a **lifted** formulation
**Theorem** (B. and Mark Zuckerberg, 2004)

For any fixed integer \( k \geq 1 \) there exists a *compact*, *extended* formulation whose solutions satisfy all valid inequalities with pitch \( \leq k \).

"compact:" of polynomial size (for fixed \( k \))

"extended:" uses additional variables, a *lifted* formulation

**Corollary:** For any fixed positive integer \( r \geq 1 \) and \( 0 < \epsilon < 1 \), there is a compact extended formulation for set-covering whose solutions satisfy the *rank-\( r \) Gomory* closure within multiplicative error \( \epsilon \).
Theorem (B. and Mark Zuckerberg, 2004)

For any fixed integer $k \geq 1$ there exists a compact, extended formulation whose solutions satisfy all valid inequalities with pitch $\leq k$.

“compact:” of polynomial size (for fixed $k$)

“extended:” uses additional variables, a lifted formulation

Corollary: For any fixed positive integer $r \geq 1$ and $0 < \epsilon < 1$, there is a compact extended formulation for set-covering whose solutions satisfy the rank-$r$ Gomory closure within multiplicative error $\epsilon$

\[ \forall c \in \mathbb{R}^n : \]

\[ \min c^T x \quad \text{s.t. } x \in \text{projected formulation} \geq \]

\[ (1 - \epsilon) \left( \min c^T x \quad \text{s.t. } x \in \text{rank}-r \text{ Gomory closure} \right) \]
Two recent, related papers:

- M. Mastrolilli (sum-of-squares mod 2)
- S. Fiorini, T. Huynh and S. Weltge (circuit complexity)
Two recent, related papers:

- M. Mastrolilli (sum-of-squares mod 2)
- S. Fiorini, T. Huynh and S. Weltge (circuit complexity)
- They point out that the B-Z formulation is 'complex'
Two recent, related papers:

- M. Mastrolilli (sum-of-squares mod 2)
- S. Fiorini, T. Huynh and S. Weltge (circuit complexity)
- They point out that the B-Z formulation is ‘complex’
- Today, a shorter proof +
Consider a (known) valid inequality

\[ \sum_{j \in S} a_j x_j \geq a_0 \quad ( > 0 ) \]

for a \textbf{binary} optimization problem.
Vector Branching (from Z’s PhD thesis)

Consider a (known) valid inequality

$$\sum_{j \in S} a_j x_j \geq a_0 \quad (> 0)$$

for a binary optimization problem.

Let $S = \{j_1, j_2, \ldots, j_t\}$. 

Vector Branching (from Z’s PhD thesis)

Consider a (known) valid inequality

\[ \sum_{j \in S} a_j x_j \geq a_0 \quad (> 0) \]

for a binary optimization problem. Let \( S = \{j_1, j_2, \ldots, j_t\} \). Then

- \( x_{j_1} = 1 \), or

- \( x_{j_1} = 0 \) and \( x_{j_2} = 1 \), or

- \( x_{j_1} = x_{j_2} = 0 \) and \( x_{j_3} = 1 \), or

- ...
Vector Branching (from Z’s PhD thesis)

Consider a (known) valid inequality

\[ \sum_{j \in S} a_j x_j \geq a_0 \quad ( > 0 ) \]

for a binary optimization problem. Let \( S = \{j_1, j_2, \ldots, j_t\} \). Then

- \( x_{j_1} = 1 \), or
- \( x_{j_1} = 0 \) and \( x_{j_2} = 1 \), or
Vector Branching (from Z’s PhD thesis)

Consider a (known) valid inequality

$$\sum_{j \in S} a_j x_j \geq a_0 \ ( > 0 )$$

for a binary optimization problem. Let $S = \{j_1, j_2, \ldots, j_t\}$. Then

- $x_{j_1} = 1$, or
- $x_{j_1} = 0$ and $x_{j_2} = 1$, or
- $x_{j_1} = x_{j_2} = 0$ and $x_{j_3} = 1$, or
Vector Branching (from Z’s PhD thesis)

Consider a (known) valid inequality

$$\sum_{j \in S} a_j x_j \geq a_0 \quad (> 0)$$

for a binary optimization problem.

Let $S = \{j_1, j_2, \ldots, j_t\}$. Then

- $x_{j_1} = 1$, or
- $x_{j_1} = 0$ and $x_{j_2} = 1$, or
- $x_{j_1} = x_{j_2} = 0$ and $x_{j_3} = 1$, or
- $\ldots$
- $x_{j_1} = \ldots = x_{j_{t-1}} = 0$ and $x_{j_t} = 1$,

is a valid disjunction.
Consider a (known) valid inequality

\[ \sum_{j \in S} a_j x_j \geq a_0 \quad (a_0 > 0) \]

for a binary optimization problem. Let \( S = \{j_1, j_2, \ldots, j_t\} \). Then

- \( x_{j_1} = 1 \), or
- \( x_{j_1} = 0 \) and \( x_{j_2} = 1 \), or
- \( x_{j_1} = x_{j_2} = 0 \) and \( x_{j_3} = 1 \), or
- \( \ldots \)
- \( x_{j_1} = \ldots = x_{j_{t-1}} = 0 \) and \( x_{j_t} = 1 \),

is a valid disjunction

Gives rise to an alternate scheme for branch-and-bound
Theorem

Given a set-covering problem, suppose we apply vector branching to a given constraint

\[ \sum_{j \in H} x_j \geq 1 \]
**Theorem**

Given a set-covering problem, suppose we apply vector branching to a given constraint

$$\sum_{j \in H} x_j \geq 1$$

Then, the solution to any node of the branch-and-bound (sub)tree thus created satisfies every valid inequality

$$\alpha^T x \geq 2$$

where

- $\alpha_j \in \{0, 1, 2\}$ for $j = 1, \ldots, n$
- $H$ contained in the support of $\alpha$
Example

Consider a valid inequality

$$\sum_{j \in S} x_j \geq 2$$  \hspace{1cm} (1)

and suppose we vector-branch on a set covering constraint

$$\sum_{j \in H} x_j \geq 1, \hspace{0.5cm} \text{with} \hspace{0.5cm} H \subseteq S$$

And now consider a node where \( x_{j_k} = 1 \) with \( j_k \in H \). But:
Example

Consider a valid inequality

$$\sum_{j \in S} x_j \geq 2$$  \hspace{1cm} (1)

and suppose we vector-branch on a set covering constraint

$$\sum_{j \in H} x_j \geq 1, \quad \text{with } H \subseteq S$$

And now consider a node where \( x_{jk} = 1 \) with \( j_k \in H \). But:

Since (1) is valid, so is:

$$\sum_{j \in S - j_k} x_j \geq 1$$  \hspace{1cm} (2)

But, set-covering,
Example

Consider a valid inequality

\[ \sum_{j \in S} x_j \geq 2 \] \hspace{1cm} (1)

and suppose we vector-branch on a set covering constraint

\[ \sum_{j \in H} x_j \geq 1, \quad \text{with } H \subseteq S \]

And now consider a node where \( x_{j_k} = 1 \) with \( j_k \in H \). But:

Since (1) is valid, so is:

\[ \sum_{j \in S - j_k} x_j \geq 1 \] \hspace{1cm} (2)

But, set-covering, so (2) must be implied by a set-covering constraint.
Example

Consider a valid inequality

\[ \sum_{j \in S} x_j \geq 2 \quad (1) \]

and suppose we vector-branch on a set covering constraint

\[ \sum_{j \in H} x_j \geq 1, \quad \text{with } H \subseteq S \]

And now consider a node where \( x_{jk} = 1 \) with \( j_k \in H \). But:

Since (1) is valid, so is:

\[ \sum_{j \in S - j_k} x_j \geq 1 \quad (2) \]

But, set-covering, so (2) must be implied by a set-covering constraint. So the solution to the node must satisfy (1).
Example

Consider a valid inequality

$$\sum_{j \in S} x_j \geq 2$$

(1)

and suppose we vector-branch on a set covering constraint

$$\sum_{j \in H} x_j \geq 1, \quad \text{with } H \subseteq S$$

And now consider a node where \( x_{jk} = 1 \) with \( j_k \in H \). But:

Since (1) is valid, so is:

$$\sum_{j \in S - j_k} x_j \geq 1$$

(2)

But, set-covering, so (2) must be implied by a set-covering constraint. So the solution to the node must satisfy (1). Related: Letchford 2001
Pitch \( k \)

Consider a valid inequality of pitch \( k \):

\[
\sum_{j \in S} \alpha_j x_j \geq \alpha_0 \tag{3}
\]

and suppose we vector-branch on a set covering constraint

\[
\sum_{j \in H} x_j \geq 1, \quad \text{with } H \subseteq S
\]

And now consider a node where \( x_{jk} = 1 \) with \( j_k \in H \). But:
Pitch k

Consider a valid inequality of pitch $k$:

$$\sum_{j \in S} \alpha_j x_j \geq \alpha_0$$  \hfill (3)

and suppose we vector-branch on a set covering constraint

$$\sum_{j \in H} x_j \geq 1, \quad \text{with } H \subseteq S$$

And now consider a node where $x_{jk} = 1$ with $j_k \in H$. But:

Since (3) is valid, so is:

$$\sum_{j \in S - j_k} \alpha_j x_j \geq \alpha_0 - \alpha_{j_k}$$  \hfill (4)

But,
Consider a valid inequality of pitch $k$:

$$\sum_{j \in S} \alpha_j x_j \geq \alpha_0$$  \hfill (3)

and suppose we vector-branch on a set covering constraint

$$\sum_{j \in H} x_j \geq 1, \quad \text{with } H \subseteq S$$

And now consider a node where $x_{j_k} = 1$ with $j_k \in H$. But:

Since (3) is valid, so is:

$$\sum_{j \in S - j_k} \alpha_j x_j \geq \alpha_0 - \alpha_{j_k}$$  \hfill (4)

But, (4) has pitch $\leq k - 1$
**Pitch k**

Consider a valid inequality of pitch $k$:

$$\sum_{j \in S} \alpha_j x_j \geq \alpha_0$$

and suppose we vector-branch on a set covering constraint

$$\sum_{j \in H} x_j \geq 1, \quad \text{with } H \subseteq S$$

And now consider a node where $x_{jk} = 1$ with $j_k \in H$. **But:** Since (3) is valid, so is:

$$\sum_{j \in S - j_k} \alpha_j x_j \geq \alpha_0 - \alpha_{j_k}$$

But, (4) has pitch $\leq k - 1$

So all we need is a **recursive** construction
Construction

• Set-covering system
  \[ Ax \geq e \]

• Pitch \( p \geq 2 \)
  \[ Z_{p-1} \]: recursively constructed formulation whose solutions satisfy all valid inequalities of pitch \( \leq p - 1 \).

• For \( p = 2 \), \( Z_{p-1} \) is the original formulation \( Ax \geq e \).

• Now we will consider a row \( i \) of \( Ax \geq e \) and, effectively, vector-branch on it.

• Actually we will write the corresponding disjunction:

Let the row be
\[
\sum_{j \in S_i} x_j \geq 1
\]
where
\[
S_i = \{ j_1, j_2, \ldots, j_{|S_i|} \}.
\]
Construction – a few corners are cut

• Set-covering system $Ax \geq e$.

• Pitch $p \geq 2$

• $\mathcal{Z}^{p-1}$: recursively constructed formulation whose solutions satisfy all valid inequalities of pitch $\leq p - 1$.

• For $p = 2$, 
Construction – a few corners are cut

- Set-covering system $Ax \geq e$.
- Pitch $p \geq 2$
- $\mathcal{Z}^{p-1}$: recursively constructed formulation whose solutions satisfy all valid inequalities of pitch $\leq p - 1$.
- For $p = 2$, $\mathcal{Z}^{p-1}$ is the original formulation $Ax \geq e$
- Now we will consider a row $i$ of $Ax \geq e$ and, effectively, vector-branch on it.
- Actually we will write the corresponding disjunction

Let the row be

$$\sum_{j \in S^i} x_j \geq 1$$

where $S^i = \{j_1, j_2, \ldots, j_{|S^i|}\}$. 
Row $i$ of $Ax \geq e$: $\sum_{j \in S_i} x_j \geq 1$, where $S_i = \{j_1, \ldots, j_{|S_i|}\}$.

(a) For $1 \leq t \leq |S^i|$, polyhedron $D^p_i(t) \subseteq \mathbb{R}^n$ given by

\begin{align*}
x_{jt} & = 1 \\
x_{jh} & = 0 \quad \forall 1 \leq h < t, \quad \text{and} \\
x & \in \mathbb{Z}^{p-1}
\end{align*}

(b) Polyhedron $D^p_i = \text{conv}\{D^p_i(t) : 1 \leq t \leq |S^i|\}$
Row \( i \) of \( Ax \geq e \): \( \sum_{j \in S^i} x_j \geq 1 \), where \( S^i = \{j_1, \ldots, j_{|S^i|}\} \).

(a) For \( 1 \leq t \leq |S^i| \), polyhedron \( D^p_i(t) \subseteq \mathbb{R}^n \) given by

\[
\begin{align*}
x_{jt} &= 1 \\
x_{jh} &= 0 \quad \forall \ 1 \leq h < t, \quad \text{and} \\
x &\in \mathbb{Z}^{p-1}
\end{align*}
\]

(b) Polyhedron \( D^p_i \doteq \text{conv}\{D^p_i(t) : 1 \leq t \leq |S^i|\} \)

Finally: \( Z^p \doteq \bigcap_i D^{p-1}_i \)
Row $i$ of $Ax \geq e$: $\sum_{j \in S_i} x_j \geq 1$, where $S^i = \{j_1, \ldots, j_{|S^i|}\}$. 

(a) For $1 \leq t \leq |S^i|$, polyhedron $D^p_i(t) \subseteq \mathbb{R}^n$ given by

\begin{align}
x_{j_t} &= 1 \\
x_{j_h} &= 0 \quad \forall 1 \leq h < t, \quad \text{and} \\
x &\in \mathbb{Z}^{p-1} 
\end{align}

(b) Polyhedron $D^p_i = \text{conv}\{D^p_i(t) : 1 \leq t \leq |S^i|\}$

Finally: $Z^p = \bigcap_i D^p_i$.

**Lemma:** $Z^p$ can be described by a polynomial-size formulation for fixed $p$, and its feasible solutions satisfy all valid inequalities of pitch $\leq p$. 
Subapplication 1a: minimum knapsack

\[
\begin{align*}
\text{min} & \quad c^T x \\
\text{s.t.} & \quad \sum_j w_j x_j \geq b, \quad x \text{ binary} \\
& \quad w_j \geq 0, \quad b > 0
\end{align*}
\]

- "FPTAS" exists (the one I know requires a disjunction)
- Problem not well understood

Open question:
Given \( w, b \) is there a compact extended formulation that yields a constant factor approximation, \( \forall c \)?

ANY constant whatsoever?
Subapplication 1a: minimum knapsack

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad \sum_j w_j x_j \geq b, \quad x \text{ binary}
\end{align*}
\]

\(w \geq 0, \quad b > 0\)

- “FPTAS” exists
Subapplication 1a: minimum knapsack

\[
\min \ c^T x \\
\text{s.t.} \quad \sum_{j} w_j x_j \geq b, \quad x \text{ binary}
\]

\(w \geq 0, \ b > 0\)

- "FPTAS" exists (the one I know requires a disjunction)
Subapplication 1a: minimum knapsack

\[
\begin{align*}
\text{min} & \quad c^T x \\
\text{s.t.} & \quad \sum_j w_j x_j \geq b, \quad x \text{ binary}
\end{align*}
\]

\( w \geq 0, \quad b > 0 \)

- “FPTAS” exists (the one I know requires a disjunction)
- Problem not well understood
Subapplication 1a: minimum knapsack

\[ \min \quad c^T x \]
\[ \text{s.t.} \quad \sum_{j} w_j x_j \geq b, \quad x \text{ binary} \]

\( w \geq 0, \quad b > 0 \)

- “FPTAS” exists (the one I know requires a disjunction)
- Problem not well understood

Open question:
Subapplication 1a: minimum knapsack

\[ \min c^T x \]
\[ \text{s.t. } \sum_j w_j x_j \geq b, \quad x \text{ binary} \]

\( w \geq 0, \quad b > 0 \)

- “FPTAS” exists (the one I know requires a disjunction)
- Problem not well understood

**Open question:** Given \( w, b \) is there a compact extended formulation that yields a constant factor approximation, \( \forall c \)?
Subapplication 1a: minimum knapsack

\[ \begin{align*}
\text{min} & \quad c^T x \\
\text{s.t.} & \quad \sum_j w_j x_j \geq b, \quad x \text{ binary}
\end{align*} \]

\( w \geq 0, \quad b > 0 \)

- “FPTAS” exists (the one I know requires a disjunction)
- Problem not well understood

**Open question:** Given \( w, b \) is there a compact extended formulation that yields a constant factor approximation, \( \forall c \)?

**ANY** constant whatsoever?
Application 1a: minimum knapsack

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad \sum_{j} w_j x_j \geq b \\
\end{align*}
\]

\(w \geq 0, \quad b > 0, \quad \text{integral}\)

Well-known result: equivalent to set-covering problem,
Application 1a: minimum knapsack

\[ \min c^T x \]

s.t. \[ \sum_j w_j x_j \geq b \]

\[ w \geq 0, \; b > 0, \; \text{integral} \]

Well-known result: equivalent to set-covering problem, with constraints

\[ \sum_{j \in S} x_j \geq 1, \; \forall S \; \text{with} \; \sum_{j \in S} w_j \geq w^* = \sum_j w_j - b + 1 \]
Application 1a: minimum knapsack

\[
\begin{align*}
\text{min } & \quad c^T x \\
\text{s.t. } & \quad \sum_{j} w_j x_j \geq b \\
\end{align*}
\]

\[w \geq 0, \quad b > 0, \quad \text{integral}\]

Well-known result: equivalent to set-covering problem, with constraints

\[
\sum_{j \in S} x_j \geq 1, \quad \forall S \quad \text{with} \quad \sum_{j \in S} w_j \geq w^* = \sum_{j} w_j - b + 1
\]

But exponentially many constraints
Minimum knapsack

Using equivalence with set-covering,
Minimum knapsack

Using equivalence with set-covering,

• Compact, extended formulation that yields valid inequalities of
  \textit{pitch} \leq k, for fixed \( k \)?
Minimum knapsack

Using equivalence with set-covering,

- Compact, extended formulation that yields valid inequalities of 
  \( \text{pitch} \leq k \), for fixed \( k \)? \[ X \]
Minimum knapsack

Using equivalence with set-covering,

• Compact, extended formulation that yields valid inequalities of \( \text{pitch} \leq k \), for fixed \( k \)? \( \times \)

• Compact, extended formulation that yields valid inequalities with \text{coefficients} in \( 0, 1, \ldots, k \), for fixed \( k \)?
Minimum knapsack

Using equivalence with set-covering,

- Compact, extended formulation that yields valid inequalities of \( \text{pitch} \leq k \), for fixed \( k \)? \( \times \)
- Compact, extended formulation that yields valid inequalities with coefficients in \( 0, 1, \ldots, k \), for fixed \( k \)? \( \times \)
Minimum knapsack

Using equivalence with set-covering,

- Compact, extended formulation that yields valid inequalities of pitch $\leq k$, for fixed $k$? X
- Compact, extended formulation that yields valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$? X
- Polynomial-time separation over valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$?
Minimum knapsack

Using equivalence with set-covering,

- Compact, extended formulation that yields valid inequalities of pitch $\leq k$, for fixed $k$? $\times$
- Compact, extended formulation that yields valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$? $\times$
- Polynomial-time separation over valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$? (implied)

Given $y$, either

- Find a valid inequality with coefficients in $0, 1, \ldots, k$, violated by $y$,
- Certify that $\alpha^T y \geq \alpha^0 - o(1)$ for all valid $\alpha^T x \geq \alpha^0$ with $\alpha^j \in \{0, 1, \ldots, k\}$ for all $j$.

E.g. $o(1) = O(1/n)$
Minimum knapsack

Using equivalence with set-covering,

- Compact, extended formulation that yields valid inequalities of pitch $\leq k$, for fixed $k$? X
- Compact, extended formulation that yields valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$? X
- Polynomial-time separation over valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$? (implied)
- Polynomial-time near separation over valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$.

\[\text{e.g. } o(1) = O(1/n)\]
Minimum knapsack

Using equivalence with set-covering,

- Compact, extended formulation that yields valid inequalities of $\text{pitch} \leq k$, for fixed $k$? \( \times \)
- Compact, extended formulation that yields valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$? \( \times \)
- Polynomial-time separation over valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$? (implied)
- Polynomial-time near separation over valid inequalities with coefficients in $0, 1, \ldots, k$, for fixed $k$.

Given $y$, either

- Find a valid inequality with coefficients in $0, 1, \ldots, k$, violated by $y$, or
- Certify that $\alpha^T y \geq \alpha_0 - o(1)$ for all valid $\alpha^T x \geq \alpha_0$ with $\alpha_j \in \{0, 1, \ldots, k\}$ for all $j$. 

\(e.g. o(1) = O(1/n)\)
Minimum knapsack

Using equivalence with set-covering,

- Compact, extended formulation that yields valid inequalities of \( \text{pitch} \leq k \), for fixed \( k \)? \( \times \)
- Compact, extended formulation that yields valid inequalities with coefficients in \( 0, 1, \ldots, k \), for fixed \( k \)? \( \times \)
- Polynomial-time separation over valid inequalities with coefficients in \( 0, 1, \ldots, k \), for fixed \( k \)? (implied)
- Polynomial-time near separation over valid inequalities with coefficients in \( 0, 1, \ldots, k \), for fixed \( k \).

Given \( y \), either

- Find a valid inequality with coefficients in \( 0, 1, \ldots, k \), violated by \( y \), or
- Certify that \( \alpha^T y \geq \alpha_0 - o(1) \) for all valid \( \alpha^T x \geq \alpha_0 \) with \( \alpha_j \in \{0, 1, \ldots, k\} \) for all \( j \). e.g. \( o(1) = O(1/n) \)
knapsack: $\sum_j w_j x_j \geq b, \quad w^* = \sum_j w_j - b + 1$
**knapsack:** \[ \sum_j w_j x_j \geq b, \quad w^* = \sum_j w_j - b + 1 \]

**Warmup**

Given \( y \), does it satisfy every valid inequality \( \sum_{j \in S} x_j \geq 2 \)?
knapsack: $\sum_j w_j x_j \geq b$, $w^* = \sum_j w_j - b + 1$

**Warmup**

Given $y$, does it satisfy every valid inequality $\sum_{j \in S} x_j \geq 2$?

What is $S$ here?

- Inequality is valid iff $\forall k \in S$, $\sum_{j \in S - k} w_j \geq w^*$
knapsack: $\sum_j w_j x_j \geq b, \quad w^* = \sum j w_j - b + 1$

**Warmup**

Given $y$, does it satisfy every valid inequality $\sum_{j \in S} x_j \geq 2$? What is $S$ here?

- Inequality is valid iff $\forall k \in S$, $\sum_{j \in S-k} w_j \geq w^*$
- Same as: $\sum_{j \in S-k} w_j \geq w^*$ for specific $k : \text{argmax}_{j \in S} \{w_j\}$
knapsack: \( \sum_j w_j x_j \geq b, \quad w^* \equiv \sum_j w_j - b + 1 \)

**Warmup**

Given \( y \), does it satisfy every valid inequality \( \sum_{j \in S} x_j \geq 2 \)?

What is \( S \) here?

- Inequality is valid iff \( \forall k \in S, \sum_{j \in S - k} w_j \geq w^* \)
- Same as: \( \sum_{j \in S - k} w_j \geq w^* \) for specific \( k : \arg\max_{j \in S} \{ w_j \} \)
- For \( k = 1, 2, \ldots, n \), solve minimum-knapsack problem

\[
\min \sum_j y_j z_j \quad (8)
\]

s.t. \( \sum_{j \neq k} w_j z_j \geq w^*, \quad z \text{ binary} \quad (9) \)

\[
z_k = 1, \quad z_j = 0 \quad \forall j \text{ with } w_j > w_k \quad (10)
\]
**knapsack:** \( \sum_j w_j x_j \geq b, \quad w^* = \sum_j w_j - b + 1 \)

**Warmup**

Given \( y \), does it satisfy every valid inequality \( \sum_{j \in S} x_j \geq 2 \)?

What is \( S \) here?

- Inequality is valid iff \( \forall k \in S, \quad \sum_{j \in S - k} w_j \geq w^* \)
- Same as: \( \sum_{j \in S - k} w_j \geq w^* \) for specific \( k : \arg\max_{j \in S} \{ w_j \} \)
- For \( k = 1, 2, \ldots, n \), solve minimum-knapsack problem

\[
\min \sum_j y_j z_j \quad \text{(8)}
\]

s.t.

\[
\sum_{j \neq k} w_j z_j \geq w^*, \quad z \text{ binary} \quad \text{(9)}
\]

\[
z_k = 1, \quad z_j = 0 \quad \forall j \text{ with } w_j > w_k \quad \text{(10)}
\]

**Wait, how do we solve?**
knapsack: $\sum_j w_j x_j \geq b$, $w^* = \sum_j w_j - b + 1$

Warmup

Given $y$, does it satisfy every valid inequality $\sum_{j \in S} x_j \geq 2$?

What is $S$ here?

- Inequality is valid iff $\forall k \in S$, $\sum_{j \in S - k} w_j \geq w^*$
- Same as: $\sum_{j \in S - k} w_j \geq w^*$ for specific $k : \arg\max_{j \in S} \{w_j\}$
- For $k = 1, 2, \ldots, n$, solve minimum-knapsack problem

$$\min \sum_j y_j z_j$$  \hfill (8)

subject to

$$\sum_{j \neq k} w_j z_j \geq w^*, \quad z \text{ binary}$$  \hfill (9)

$$z_k = 1, \quad z_j = 0 \ \forall j \text{ with } w_j > w_k$$  \hfill (10)

Wait, how do we solve?

In objective round up $y_j$, to next multiple of $1/n^2$
knapsack: \[ \sum_j w_j x_j \geq b, \quad w^* = \sum_j w_j - b + 1 \]

**Warmup**

Given \( y \), does it satisfy every valid inequality \( \sum_{j \in S} x_j \geq 2 \)?

What is \( S \) here?

- Inequality is valid iff \( \forall k \in S, \sum_{j \in S - k} w_j \geq w^* \)
- Same as: \( \sum_{j \in S - k} w_j \geq w^* \) for specific \( k : \arg\max_{j \in S} \{w_j\} \)
- For \( k = 1, 2, \ldots, n \), solve minimum-knapsack problem

\[
\begin{align*}
\min & \quad \sum_j y_j z_j \\
\text{s.t.} & \quad \sum_{j \neq k} w_j z_j \geq w^*, \quad z \text{ binary} \\
& \quad z_k = 1, \quad z_j = 0 \quad \forall j \text{ with } w_j > w_k
\end{align*}
\]

Wait, how do we solve?

In objective round up \( y_j \), to next multiple of \( 1/n^2 \)

So, get approximate separation, with violation if objective < 2
knapsack: $\sum_j w_j x_j \geq b$, $w^* = \sum_j w_j - b + 1$

Second warmup

Given $y$, does it satisfy every valid inequality $2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2$?
knapsack: \( \sum_j w_j x_j \geq b \), \( w^* = \sum_j w_j - b + 1 \)

**Second warmup**

Given \( y \), does it satisfy every valid inequality \( 2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2 \)?

What are \( T, S \) here?

- Inequality is valid iff \( \forall k \in S, \sum_{j \in T \cup S - k} w_j \geq w^* \)
knapsack: $\sum_j w_j x_j \geq b$, $w^* = \sum_j w_j - b + 1$

Second warmup

Given $y$, does it satisfy every valid inequality $2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2$?

What are $T, S$ here?

- Inequality is valid iff $\forall k \in S, \sum_{j \in T \cup S - k} w_j \geq w^*$
- Same as: $\sum_{j \in T \cup S - k} w_j \geq w^*$ for specific $k : \arg\max_{j \in S}\{w_j\}$
knapsack: $\sum_j w_j x_j \geq b$, $w^* = \sum_j w_j - b + 1$

**Second warmup**

Given $y$, does it satisfy every valid inequality $2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2$?

What are $T$, $S$ here?

- Inequality is valid iff $\forall k \in S$, $\sum_{j \in T \cup S - k} w_j \geq w^*$
- Same as: $\sum_{j \in T \cup S - k} w_j \geq w^*$ for specific $k : \text{argmax}_{j \in S} \{w_j\}$

**Example:** $10x_1 + 10x_2 + 5x_3 + 7x_4 + 6x_5 \geq 10$

**Valid:** $2(x_1 + x_2 + x_3) + x_4 + x_5 \geq 2$

**Stronger:** $2(x_1 + x_2) + x_3 + x_4 + x_5 \geq 2$

The stronger inequality is **monotone** in the knapsack weights:
(bigger weight in knapsack $\rightarrow$ bigger coefficient in inequality)
knapsack: $\sum_j w_j x_j \geq b$, $w^* = \sum_j w_j - b + 1$

Second warmup

Given $y$, does it satisfy every valid inequality $2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2$? What are $T, S$ here?

- Inequality is valid iff $\forall k \in S$, $\sum_{j \in T \cup S - k} w_j \geq w^*$
- Same as: $\sum_{j \in T \cup S - k} w_j \geq w^*$ for specific $k : \arg\max_{j \in S} \{w_j\}$

Example: $10x_1 + 10x_2 + 5x_3 + 7x_4 + 6x_5 \geq 10$

Stronger: $2(x_1 + x_2) + x_3 + x_4 + x_5 \geq 2$

The stronger inequality is monotone in the knapsack weights:
(bigger weight in knapsack $\rightarrow$ bigger coefficient in inequality)
knapsack: $\sum_j w_j x_j \geq b$, $w^* = \sum_j w_j - b + 1$

Second warmup

Given $y$, does it satisfy every valid inequality $2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2$?

What are $T, S$ here?

- Inequality is valid iff $\forall k \in S$, $\sum_{j \in T \cup S - k} w_j \geq w^*$
- Same as: $\sum_{j \in T \cup S - k} w_j \geq w^*$ for specific $k : \arg\max_{j \in S} \{w_j\}$

Example: $10x_1 + 10x_2 + 5x_3 + 7x_4 + 6x_5 \geq 10$

Valid: $2(x_1 + x_2 + x_3) + x_4 + x_5 \geq 2$

Stronger: $2(x_1 + x_2) + x_3 + x_4 + x_5 \geq 2$
knapsack: $\sum_j w_j x_j \geq b, \quad w^* \triangleq \sum_j w_j - b + 1$

**Second warmup**

Given $y$, does it satisfy every valid inequality $2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2$? What are $T, S$ here?

- Inequality is valid iff $\forall k \in S, \sum_{j \in T \cup S - k} w_j \geq w^*$
- Same as: $\sum_{j \in T \cup S - k} w_j \geq w^*$ for specific $k : \arg\max_{j \in S} \{w_j\}$

**Example:** $10x_1 + 10x_2 + 5x_3 + 7x_4 + 6x_5 \geq 10$

**Valid:** $2(x_1 + x_2 + x_3) + x_4 + x_5 \geq 2$

**Stronger:** $2(x_1 + x_2) + x_3 + x_4 + x_5 \geq 2$
knapsack: \( \sum_j w_j x_j \geq b, \quad w^* = \sum_j w_j - b + 1 \)

**Second warmup**

Given \( y \), does it satisfy every valid inequality \( 2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2 \)?

What are \( T, S \) here?

- Inequality is valid iff \( \forall k \in S, \sum_{j \in T \cup S - k} w_j \geq w^* \)
- Same as: \( \sum_{j \in T \cup S - k} w_j \geq w^* \) for specific \( k : \arg\max_{j \in S} \{ w_j \} \)

**Example:** \( 10x_1 + 10x_2 + 5x_3 + 7x_4 + 6x_5 \geq 10 \)

**Valid:** \( 2(x_1 + x_2 + x_3) + x_4 + x_5 \geq 2 \)

**Stronger:** \( 2(x_1 + x_2) + x_3 + x_4 + x_5 \geq 2 \)

The stronger inequality is **monotone** in the knapsack weights:
(bigger weight in knapsack \( \rightarrow \) bigger coefficient in inequality)
knapsack: \[ \sum_j w_j x_j \geq b, \quad w^* = \sum_j w_j - b + 1 \]

**Second warmup**

Given \( y \), does it satisfy every valid inequality \[ 2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2? \]
knapsack: \[ \sum_j w_j x_j \geq b, \quad w^* = \sum_j w_j - b + 1 \]

Second warmup

Given \( y \), does it satisfy every valid inequality \[ 2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2? \]

What are \( T, S \) here?

- Inequality is valid iff \( \forall k \in S, \sum_{j \in T \cup S - k} w_j \geq w^* \)
knapsack: \( \sum_j w_j x_j \geq b, \quad w^* := \sum_j w_j - b + 1 \)

**Second warmup**

Given \( y \), does it satisfy every valid inequality \( 2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2? \)

What are \( T, S \) here?

- Inequality is valid iff \( \forall k \in S, \sum_{j \in T \cup S - k} w_j \geq w^* \)
- Same as: \( \sum_{j \in T \cup S - k} w_j \geq w^* \) for specific \( k : \text{argmax}_{j \in S} \{w_j\} \)
knapsack: $\sum_j w_j x_j \geq b, \quad w^* = \sum_j w_j - b + 1$

Second warmup

Given $y$, does it satisfy every valid inequality $2 \sum_{j \in T} x_j + \sum_{j \in S} x_j \geq 2$? What are $T, S$ here?

- Inequality is valid iff $\forall k \in S, \sum_{j \in T \cup S - k} w_j \geq w^*$
- Same as: $\sum_{j \in T \cup S - k} w_j \geq w^*$ for specific $k : \arg\max_{j \in S} \{w_j\}$
- To separate $y$, for $k = 1, 2, \ldots, n$, solve minimum-knapsack problem

$$\min 2 \sum_{j \in B} \tilde{y}_j z_j + \sum_{j \in L} \tilde{y}_j z_j \quad (\tilde{y} = y \text{ "rounded up" })$$

s.t. $\sum_{j \neq k} w_j z_j \geq w^*, \quad z \text{ binary}$

$z_k = 1, \quad L = \{ j : w_j \leq w_k \} \quad B = \{ j : w_j > w_k \}$
General case? First, coefficients in 0, 1, 2, 3

Example: \[8x_1 + 5x_2 + 4x_3 + 4x_4 + 4x_5 \geq 13\] (the knapsack)

Valid: \[x_1 + 2x_2 + x_3 + x_4 + x_5 \geq 3\] (non-monotone)

Not valid: \[x_1 + x_2 + x_3 + x_4 + x_5 \geq 3\]

A non-monotone “transposition” or “error”
General case? First, coefficients in 0, 1, 2, 3

**Example:** \[ 8x_1 + 5x_2 + 4x_3 + 4x_4 + 4x_5 \geq 13 \] (the knapsack)

**Valid:** \[ x_1 + 2x_2 + x_3 + x_4 + x_5 \geq 3 \] (non-monotone)

**Not valid:** \[ x_1 + x_2 + x_3 + x_4 + x_5 \geq 3 \]

A non-monotone “transposition” or “error”

**Example:** \[ 6x_1 + 6x_2 + 5x_3 + 4x_4 + 4x_5 \geq 13 \] (the knapsack)

**Valid:** \[ x_1 + x_2 + 2x_3 + x_4 + x_5 \geq 3 \] (non-monotone, 2 errors)

**Yes valid:** \[ x_1 + x_2 + x_3 + x_4 + x_5 \geq 3 \]
General case? First, coefficients in 0, 1, 2, 3

**Example:** \(8x_1 + 5x_2 + 4x_3 + 4x_4 + 4x_5 \geq 13\) (the knapsack)

**Valid:** \(x_1 + 2x_2 + x_3 + x_4 + x_5 \geq 3\) (non-monotone)

**Not valid:** \(x_1 + x_2 + x_3 + x_4 + x_5 \geq 3\)

A non-monotone “transposition” or “error”

**Example:** \(6x_1 + 6x_2 + 5x_3 + 4x_4 + 4x_5 \geq 13\) (the knapsack)

**Valid:** \(x_1 + x_2 + 2x_3 + x_4 + x_5 \geq 3\) (non-monotone, 2 errors)

**Yes valid:** \(x_1 + x_2 + x_3 + x_4 + x_5 \geq 3\)

\(\rightarrow\) When right-hand side \(= 3\), at most **one** error
General case? First, coefficients in 0, 1, 2, 3

Example:  $8x_1 + 5x_2 + 4x_3 + 4x_4 + 4x_5 \geq 13$ (the knapsack)

Valid:  $x_1 + 2x_2 + x_3 + x_4 + x_5 \geq 3$ (non-monotone)

Not valid:  $x_1 + x_2 + x_3 + x_4 + x_5 \geq 3$

A non-monotone “transposition” or “error”

Example:  $6x_1 + 6x_2 + 5x_3 + 4x_4 + 4x_5 \geq 13$ (the knapsack)

Valid:  $x_1 + x_2 + 2x_3 + x_4 + x_5 \geq 3$ (non-monotone, 2 errors)

Yes valid:  $x_1 + x_2 + x_3 + x_4 + x_5 \geq 3$

→ When right-hand side = 3, at most one error
Separation by enumeration of errors; each case is a knapsack;
General case? First, coefficients in 0, 1, 2, 3

Example: $8x_1 + 5x_2 + 4x_3 + 4x_4 + 4x_5 \geq 13$ (the knapsack)

Valid: $x_1 + 2x_2 + x_3 + x_4 + x_5 \geq 3$ (non-monotone)

Not valid: $x_1 + x_2 + x_3 + x_4 + x_5 \geq 3$

A non-monotone “transposition” or “error”

Example: $6x_1 + 6x_2 + 5x_3 + 4x_4 + 4x_5 \geq 13$ (the knapsack)

Valid: $x_1 + x_2 + 2x_3 + x_4 + x_5 \geq 3$ (non-monotone, 2 errors)

Yes valid: $x_1 + x_2 + x_3 + x_4 + x_5 \geq 3$

→ When right-hand side $= 3$, at most one error
Separation by enumeration of errors; each case is a knapsack; $O(n^2)$ cases
General case? (coefficients in 0, 1, 2, ..., k)

Basic principle: an inequality

\[ k x(S_k) + (k - 1) x(S_{k-1}) + \ldots + x(S_1) \geq k \]  

is equivalent to its set of covers –
General case? (coefficients in 0, 1, 2, ..., k)

Basic principle: an inequality

\[ k x(S_k) + (k - 1) x(S_{k-1}) + \ldots + x(S_1) \geq k \]  \hspace{1cm} (11)

is equivalent to its set of covers –

so (11) is valid iff
General case? (coefficients in 0, 1, 2, ..., k)

Basic principle: an inequality

\[ k \times (S_k) + (k - 1) \times (S_{k-1}) + \ldots + x(S_1) \geq k \]  

(11)

is equivalent to its set of covers –

so (11) is valid iff its covers are also covers for the original knapsack
General case? (coefficients in 0, 1, 2, ..., k)

Basic principle: an inequality

\[ k x(S_k) + (k - 1) x(S_{k-1}) + \ldots + x(S_1) \geq k \quad (11) \]

is equivalent to its set of covers –

so (11) is valid iff its covers are also covers for the original knapsack

Corollary: can show that (11) can have at most \(< k\) errors
General case? (coefficients in 0, 1, 2, ..., k)

Basic principle: an inequality

\[ k \times (S_k) + (k - 1) \times (S_{k-1}) + \ldots + x(S_1) \geq k \quad (11) \]

is equivalent to its set of covers –

so (11) is valid iff its covers are also covers for the original knapsack

**Corollary:** can show that (11) can have at most \(< k\) errors

or else it is dominated, or invalid

Separation by **enumeration** of errors; each case is a knapsack;
**General case? (coefficients in 0, 1, 2, ..., k)**

Basic principle: an inequality

\[ k x(S_k) + (k - 1) x(S_{k-1}) + \ldots + x(S_1) \geq k \]  \hspace{1cm} (11)

is equivalent to its set of covers –

so (11) is valid iff its covers are also covers for the original knapsack

**Corollary:** can show that (11) can have at most \(< k\) errors

or else it is dominated, or invalid

Separation by **enumeration** of errors; each case is a knapsack;

\[ O(n^{F(k)}) \] cases
Application 2: polynomial optimization problems and NN training

Polynomial optimization:

$$\min c^T x$$

subject to

$$f_i(x) \leq 0, \quad i = 1, \ldots, m$$

(polynomial ineq.)

$$0 \leq x_j \leq 1, \quad \text{all } j \quad (12)$$
Application 2: polynomial optimization problems and NN training

Polynomial optimization:

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad f_i(x) \leq 0, \quad i = 1, \ldots, m \quad \text{(polynomial ineq.)} \\
& \quad 0 \leq x_j \leq 1, \quad \text{all } j
\end{align*}
\] (12)

- **Intersection graph**
Application 2: polynomial optimization problems and NN training

Polynomial optimization:

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad f_i(x) \leq 0, \quad i = 1, \ldots, m \quad \text{(polynomial ineq.)} \\
& \quad 0 \leq x_j \leq 1, \quad \text{all } j
\end{align*}
\] (12)

- **Intersection graph**
  A vertex for each variable and an edge anytime two variables appear in the same \( f_i \)

- **Tree-width**
Polynomial optimization:

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad f_i(x) \leq 0, \quad i = 1, \ldots, m \quad \text{(polynomial ineq.)} \\
& \quad 0 \leq x_j \leq 1, \quad \text{all } j
\end{align*}
\]

- **Intersection graph**
Polynomial optimization:

$$\min \ c^T x$$

s.t.  \( f_i(x) \leq 0, \quad i = 1, \ldots, m \)  \quad \text{(polynomial ineq.)}$$

$$0 \leq x_j \leq 1, \quad \text{all } j$$

- **Intersection graph**
  A vertex for each variable and an edge anytime two variables appear in the same \( f_i \)

- **Tree-width** of a graph \( G \)
  Minimum clique number (minus one) over all chordal supergraphs of \( G \)
Polynomial optimization:

\[
\begin{align*}
\text{min} & \quad c^T x \\
\text{s.t.} & \quad f_i(x) \leq 0, \quad i = 1, \ldots, m \quad \text{(polynomial ineq.)} \\
& \quad 0 \leq x_j \leq 1, \quad \text{all } j
\end{align*}
\]

**Theorem** (B. and Muñoz 2015, SIOPT 2018).

Suppose:

the intersection graph has tree-width \( \omega \) and the \( f_i \) of degree \( \leq \rho \).
Polynomial optimization:

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad f_i(x) \leq 0, \quad i = 1, \ldots, m \quad \text{(polynomial ineq.)} \\
& \quad 0 \leq x_j \leq 1, \quad \text{all } j
\end{align*}
\]

**Theorem** (B. and Muñoz 2015, SIOPT 2018).

Suppose:

the intersection graph has tree-width \( \omega \) and the \( f_i \) of degree \( \leq \rho \).

Then, for every \( 0 < \epsilon < 1 \) there is a **disjunctive LP** relaxation with
Polynomial optimization:

\[
\begin{align*}
\min & \quad c^T x \\
\text{s.t.} & \quad f_i(x) \leq 0, \quad i = 1, \ldots, m \quad (\text{polynomial ineq.}) \\
& \quad 0 \leq x_j \leq 1, \quad \text{all } j
\end{align*}
\]

**Theorem** (B. and Muñoz 2015, SIOPT 2018).

Suppose:

the intersection graph has tree-width \( \omega \) and the \( f_i \) of degree \( \leq \rho \).

Then, for every \( 0 < \epsilon < 1 \) there is a **disjunctive LP** relaxation with

\[
O \left( (2\rho/\epsilon)^{\omega+1} n \log(\rho/\epsilon) \right)
\]
variables and constraints.
Polynomial optimization:

$$\min \ c^T x$$

s.t. $$f_i(x) \leq 0, \quad i = 1, \ldots, m$$ (polynomial ineq.)

$$0 \leq x_j \leq 1, \quad \text{all } j$$

**Theorem** (B. and Muñoz 2015, SI OPT 2018).

Suppose:

the intersection graph has tree-width $$\omega$$ and the $$f_i$$ of degree $$\leq \rho$$.

Then, for every $$0 < \epsilon < 1$$ there is a disjunctive LP relaxation with

$$O \left( (2\rho/\epsilon)^{\omega+1} n \log(\rho/\epsilon) \right)$$ variables and constraints

Optimality and feasibility errors $$O(\epsilon)$$ (additive)
Subapplication 2a: training of deep neural networks with RLUs

As per Arora Basu Mianjy Mukherjee ICLR '18

The setup:

• Data points \((x_i, y_i)\), \(1 \leq i \leq D\), \(x_i \in \mathbb{R}^n\), \(y_i \in \mathbb{R}\)

• Task: compute a function \(f: \mathbb{R}^n \rightarrow \mathbb{R}\) to minimize
  \[
  \frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2
  \]

• \(f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \cdots \circ \sigma \circ T_1 (\text{"\circ" = composition})\)

• \(\sigma(t) = \max\{0, t\}\)

• Each \(T_h\) affine: \(T_h(y) = A_h y + b_h\), for some \(w\), \(A_1\) is \(n \times w\), \(A_{k+1}\) is \(w \times 1\), \(A_h\) is \(w \times w\) otherwise. Similarly with the \(b_h\).
Subapplication 2a: training of deep neural networks with RLUs

As per Arora Basu Mianjy Mukherjee ICLR ’18

The setup:

- $D$ data points $(x_i, y_i), 1 \leq i \leq D, \ x_i \in \mathbb{R}^n, \ y_i \in \mathbb{R}$

- $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ (composition)

- $\sigma(t) = \max\{0, t\}$

- Each $T_h$ affine: $T_h(y) = A_h y + b_h$

- For some $w$, $A_1$ is $n \times w$, $A_{k+1}$ is $w \times 1$, $A_h$ is $w \times w$ otherwise.

- Similarly with the $b_h$. 
Subapplication 2a: training of deep neural networks with RLUs

As per Arora Basu Mianjy Mukherjee ICLR ’18

The setup:

- **D** data points \((x_i, y_i), 1 \leq i \leq D, x_i \in \mathbb{R}^n, y_i \in \mathbb{R}\)
- **Task:** compute a function \(f : \mathbb{R}^n \rightarrow \mathbb{R}\) to minimize

\[
\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2
\]
Subapplication 2a: training of deep neural networks with RLUs

As per Arora Basu Mianjy Mukherjee ICLR '18

The setup:

- **$D$ data points** $(x_i, y_i), \ 1 \leq i \leq D, \ x_i \in \mathbb{R}^n, \ y_i \in \mathbb{R}$
- **Task:** compute a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ to minimize

$$\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2$$

- $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ ("$\circ$" = composition)
Subapplication 2a: training of deep neural networks with RLUs

As per Arora Basu Mianjy Mukherjee ICLR ’18

The setup:

• $D$ data points $(x_i, y_i)$, $1 \leq i \leq D$, $x_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$

• **Task:** compute a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ to minimize

$$\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2$$

• $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ ("$\circ$" = composition)

• $\sigma(t) = \max\{0, t\}$
Subapplication 2a: training of deep neural networks with RLUs

As per Arora Basu Mianjy Mukherjee ICLR ’18

The setup:

- $D$ data points $(x_i, y_i)$, $1 \leq i \leq D$, $x_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$
- **Task:** compute a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ to minimize

$$
\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2
$$

- $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ ("$\circ$" = composition)
- $\sigma(t) = \max\{0, t\}$
- Each $T_h$ affine: $T_h(y) = A_h y + b_h$
Subapplication 2a: training of deep neural networks with RLUs

As per Arora Basu Mianjy Mukherjee ICLR '18

The setup:

• \( D \) data points \((x_i, y_i), 1 \leq i \leq D, \ x_i \in \mathbb{R}^n, \ y_i \in \mathbb{R}\)

• Task: compute a function \( f : \mathbb{R}^n \rightarrow \mathbb{R} \) to minimize

\[
\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2
\]

• \( f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1 \) \quad (“\circ” = composition)

• \( \sigma(t) = \max\{0, t\} \)

• Each \( T_h \) affine: \( T_h(y) = A_h y + b_h \),

• For some \( w, \ A_1 \) is \( n \times w, \ A_{k+1} \) is \( w \times 1, \ A_h \) is \( w \times w \) otherwise.
Subapplication 2a: training of deep neural networks with RLUs

As per Arora Basu Mianjy Mukherjee ICLR '18

The setup:

- $D$ data points $(x_i, y_i), 1 \leq i \leq D, x_i \in \mathbb{R}^n, y_i \in \mathbb{R}$
- Task: compute a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ to minimize

$$\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2$$

- $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ ("$\circ$" = composition)
- $\sigma(t) = \max\{0, t\}$
- Each $T_h$ affine: $T_h(y) = A_h y + b_h$
- For some $w$, $A_1$ is $n \times w$, $A_{k+1}$ is $w \times 1$, $A_h$ is $w \times w$ otherwise. Similarly with the $b_h$. 

• $D$ data points $(x_i, y_i), 1 \leq i \leq D$, $x_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$

• **Task:** compute a function $f : \mathbb{R}^n \to \mathbb{R}$ to minimize

$$\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2$$

• $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ ("$\circ$" = composition)

• $\sigma(t) = \max\{0, t\}$

• Each $T_h$ affine: $T_h(y) = A_h y + b_h$.

• For some $w$, $A_1$ is $n \times w$, $A_{k+1}$ is $w \times 1$, $A_h$ is $w \times w$ otherwise. Similarly with the $b_h$.
• $D$ data points $(x_i, y_i), 1 \leq i \leq D, \ x_i \in \mathbb{R}^n, \ y_i \in \mathbb{R}$

• Task: compute a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ to minimize

$$\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2$$

• $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ ("$\circ$" = composition)

• $\sigma(t) = \max\{0, t\}$

• Each $T_h$ affine: $T_h(y) = A_h y + b_h$.

• For some $w$, $A_1$ is $n \times w$, $A_{k+1}$ is $w \times 1$, $A_h$ is $w \times w$ otherwise. Similarly with the $b_h$.

**Theorem** (Arora et al 2018).

If $k = 1$ (one "hidden layer") there is an exact algorithm of complexity
• $D$ data points $(x_i, y_i)$, $1 \leq i \leq D$, $x_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$

• **Task**: compute a function $f : \mathbb{R}^n \to \mathbb{R}$ to minimize

$$\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2$$

• $f = T_{k+1} \circ \sigma \circ T_{k} \circ \sigma \ldots \circ \sigma \circ T_{1}$ ("$\circ$" = composition)

• $\sigma(t) = \max\{0, t\}$

• Each $T_h$ affine: $T_h(y) = A_h y + b_h$, 

• For some $w$, $A_1$ is $n \times w$, $A_{k+1}$ is $w \times 1$, $A_h$ is $w \times w$ otherwise. Similarly with the $b_h$.

**Theorem** (Arora et al 2018).

If $k = 1$ (one "hidden layer") there is an exact algorithm of complexity

$$O ( 2^w D^{nw} \text{poly}(D, n, w) )$$

**Polynomial** in the size of the data set, for fixed $n, w$
• $D$ data points $(x_i, y_i)$, $1 \leq i \leq D$, $x_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$

• **Task:** compute $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ to minimize
$$\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2$$

• $A_1$ is $n \times w$, $A_{k+1}$ is $w \times 1$, $A_h$ is $w \times w$ otherwise.

**Theorem** (Arora et al 2018).

If $k = 1$ (one “hidden layer”) there is an exact algorithm of complexity
$$O(2^w D^{nw} \text{poly}(D, n, w))$$
• \( D \) data points \((x_i, y_i), 1 \leq i \leq D, \ x_i \in \mathbb{R}^n, \ y_i \in \mathbb{R}\)

• **Task:** compute \( f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1 \) to minimize \( \frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2 \)

• \( A_1 \) is \( n \times w \), \( A_{k+1} \) is \( w \times 1 \), \( A_h \) is \( w \times w \) otherwise.

**Theorem** (Arora et al 2018).

If \( k = 1 \) (one “hidden layer”) there is an exact algorithm of complexity

\[
O\left(2^w D^{nw} \text{poly}(D, n, w)\right)
\]

**Application of B. and Muñoz poly-opt result:**
• $D$ data points $(x_i, y_i)$, $1 \leq i \leq D$, $x_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$

• **Task:** compute $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ to minimize

$$\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2$$

• $A_1$ is $n \times w$, $A_{k+1}$ is $w \times 1$, $A_h$ is $w \times w$ otherwise.

**Theorem** (Arora et al 2018).

**If** $k = 1$ (one “hidden layer”) **there is an exact algorithm of complexity**

$$O \left( 2^w D^{nw} \text{poly}(D, n, w) \right)$$

**Application of B. and Muñoz poly-opt result:**

• **Weakening:** Assume that a bound on the absolute value of the entries in the $A_h$, $b_h$ is known
• $D$ data points $(x_i, y_i)$, $1 \leq i \leq D$, $x_i \in \mathbb{R}^n$, $y_i \in \mathbb{R}$

• **Task:** compute $f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1$ to minimize
  $$\frac{1}{D} \sum_{i=1}^D (y_i - f(x_i))^2$$

• $A_1$ is $n \times w$, $A_{k+1}$ is $w \times 1$, $A_h$ is $w \times w$ otherwise.

**Theorem** (Arora et al 2018).

If $k = 1$ (one “hidden layer”) there is an exact algorithm of complexity

$$O \left( 2^w D^{nw} \text{poly}(D, n, w) \right)$$

**Application of B. and Muñoz poly-opt result:**

• **Weakening:** Assume that a bound on the absolute value of the entries in the $A_h$, $b_h$ is known

• **Weakening:** For any $0 < \epsilon < 1$, additive errors $O(\epsilon)$
• \( D \) data points \((x_i, y_i), 1 \leq i \leq D\), \( x_i \in \mathbb{R}^n, y_i \in \mathbb{R}\)

• **Task:** compute \( f = T_{k+1} \circ \sigma \circ T_k \circ \sigma \ldots \circ \sigma \circ T_1 \) to minimize

\[
\frac{1}{D} \sum_{i=1}^{D} (y_i - f(x_i))^2
\]

• \( A_1 \) is \( n \times w \), \( A_{k+1} \) is \( w \times 1 \), \( A_h \) is \( w \times w \) otherwise.

**Theorem** (Arora et al 2018).

If \( k = 1 \) (one “hidden layer”) there is an exact algorithm of complexity

\[
O \left( 2^w D^{nw} \text{poly}(D, n, w) \right)
\]

**Application of B. and Muñoz poly-opt result:**

• **Weakening:** Assume that a bound on the absolute value of the entries in the \( A_h, b_h \) is known

• **Weakening:** For any \( 0 < \epsilon < 1 \), additive errors \( O(\epsilon) \)

**Theorem.** For any \( k, n, w, \epsilon \) approximate LP of size

\[
O \left( \left( \frac{4}{\epsilon} \right)^{O((k-1)w^2 + nw)} \right) \text{poly}(D, n, w, k)
\]