On the complexity of integer programming with fixed-coefficient scaling

Jorke de Vlas
supervised by Peter Jonsson

September 6, 2024

LINKÖPINGS UNIVERSITET

Background

In Integer Programming, the goal is to solve linear inequalities over integers

Background

In Integer Programming, the goal is to solve linear inequalities over integers

- Integer variables $x_1, x_2, \ldots, x_n$

Background

In Integer Programming, the goal is to solve linear inequalities over integers

- Integer variables $x_1, x_2, \ldots, x_n$
- Inequalities of the form $a_1x_1 + a_2x_2 + \ldots + a_nx_n \leq b$, $a_i, b \in \mathbb{Z}$

Background

In Integer Programming, the goal is to solve linear inequalities over integers

- Integer variables $x_1, x_2, \ldots, x_n$
- Inequalities of the form $a_1x_1 + a_2x_2 + \ldots + a_nx_n \leq b$, $a_i, b \in \mathbb{Z}$

NP-complete in general

Background

In Integer Programming, the goal is to solve linear inequalities over integers

- Integer variables $x_1, x_2, \ldots, x_n$
- Inequalities of the form $a_1x_1 + a_2x_2 + \ldots + a_nx_n \leq b$, $a_i, b \in \mathbb{Z}$

NP-complete in general

Can we simplify to something solvable in polynomial time?

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1x_1 + a_2x_2 + \ldots + a_nx_n \leq b$, $a_i, b \in \mathbb{Z}$

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1x_1 + a_2x_2 + \ldots + a_nx_n \leq b,\ a_i, b \in \mathbb{Z}$

At most two variables per inequality (TVPI):

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1 x_1 + a_2 x_2 \leq b$, $a_1, a_2, b \in \mathbb{Z}$

At most two variables per inequality (TVPI):

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1x_1 + a_2x_2 \leq b$, $a_1, a_2, b \in \mathbb{Z}$

At most two variables per inequality (TVPI): NP-complete (Lagarias, 1985)

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1x_1 + a_2x_2 \leq b$, $a_1, a_2, b \in \mathbb{Z}$

At most two variables per inequality (TVPI): NP-complete (Lagarias, 1985)

TVPI, monotone inequalities:

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1x_1 \leq a_2x_2 + b$, $a_1, a_2 \in \mathbb{N}$, $b \in \mathbb{Z}$

At most two variables per inequality (TVPI): NP-complete (Lagarias, 1985)

TVPI, monotone inequalities:

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1 x_1 \leq a_2 x_2 + b$, $a_1, a_2 \in \mathbb{N}$, $b \in \mathbb{Z}$

At most two variables per inequality (TVPI): NP-complete (Lagarias, 1985)

TVPI, monotone inequalities: NP-complete (Lagarias, 1985)

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1x_1 \leq a_2x_2 + b$, $a_1, a_2 \in \mathbb{N}$, $b \in \mathbb{Z}$

At most two variables per inequality (TVPI): NP-complete (Lagarias, 1985)

TVPI, monotone inequalities: NP-complete (Lagarias, 1985)

TVPI, restrict scaling coefficients to $\{\pm 1\}$ (UTVPI):

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1x_1 + a_2x_2 \leq b$, $a_1, a_2 \in \{\pm 1\}$, $b \in \mathbb{Z}$

At most two variables per inequality (TVPI): NP-complete (Lagarias, 1985)

TVPI, monotone inequalities: NP-complete (Lagarias, 1985)

TVPI, restrict scaling coefficients to $\{\pm 1\}$ (UTVPI):

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1x_1 + a_2x_2 \leq b$, $a_1, a_2 \in \{\pm 1\}$, $b \in \mathbb{Z}$

At most two variables per inequality (TVPI): NP-complete (Lagarias, 1985)

TVPI, monotone inequalities: NP-complete (Lagarias, 1985)

TVPI, restrict scaling coefficients to $\{\pm 1\}$ (UTVPI): In P (Jaffar et al, 1994)

Background

Idea: further restrict allowed inequalities

Equations of the form $a_1x_1 + a_2x_2 \leq b$, $a_1, a_2 \in \{\pm 1, \pm 2\}$, $b \in \mathbb{Z}$

At most two variables per inequality (TVPI): NP-complete (Lagarias, 1985)

TVPI, monotone inequalities: NP-complete (Lagarias, 1985)

TVPI, restrict scaling coefficients to $\{\pm 1\}$ (UTVPI): In P (Jaffar et al, 1994)

TVPI, restrict scaling coefficients to $\{\pm 1, \pm 2\}$ (BTVPI): In P (Wojciechowski & Subramani, 2023)

In this talk

TVPI, monotone inequalities, restrict scaling coefficients to powers of some $c \in \mathbb{N}$

Equations of the form $c^{a_1} x_1 \leq c^{a_2} x_2 + b$, $a_1, a_2 \in \mathbb{N}$, $b \in \mathbb{Z}$

In this talk

TVPI, monotone inequalities, restrict scaling coefficients to powers of some $c \in \mathbb{N}$

Equations of the form $c^{a_1} x_1 \leq c^{a_2} x_2 + b$, $a_1, a_2 \in \mathbb{N}$, $b \in \mathbb{Z}$

This is in P!

In this talk

TVPI, monotone inequalities, restrict scaling coefficients to powers of some $c \in \mathbb{N}$

Equations of the form $c^{a_1} x_1 \leq c^{a_2} x_2 + b$, $a_1, a_2 \in \mathbb{N}$, $b \in \mathbb{Z}$

This is in P!

WLOG, either a_1 or a_2 is zero

In this talk

TVPI, monotone inequalities, restrict scaling coefficients to powers of some $c \in \mathbb{N}$

Equations of the form $x_1 \leq c^a x_2 + b$ or $c^a x_1 + b \leq x_2$, $a \in \mathbb{N}$, $b \in \mathbb{Z}$

This is in P!

WLOG, either a_1 or a_2 is zero

Model as graph

A vertex per variable

A (directed) arc from x_1 to x_2 per inequality

Arcs have weight: $a_1 - a_2$

Model as graph

A vertex per variable

A (directed) arc from x_1 to x_2 per inequality

Arcs have weight: $a_1 - a_2$

Example:

- $x_1 \leq 2x_2 - 1$
- $4x_2 + 1 \leq x_3$
- $x_3 \leq x_1 + 4$
- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$

Model as graph

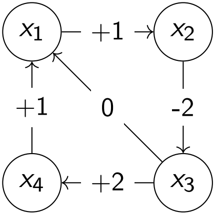
A vertex per variable

A (directed) arc from x_1 to x_2 per inequality

Arcs have weight: $a_1 - a_2$

Example:

- $x_1 \leq 2x_2 - 1$
- $4x_2 + 1 \leq x_3$
- $x_3 \leq x_1 + 4$
- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$



Transitioning into a finite domain

There exists an Ω such that:

If there exists a solution, then there exists one where each variable is bounded by Ω.

Furthermore, $\log(\Omega)$ is polynomial in the input size

Transitioning into a finite domain

There exists an Ω such that:

If there exists a solution, then there exists one where each variable is bounded by Ω.

Furthermore, $\log(\Omega)$ is polynomial in the input size

This is a (small variation of) a known result. (Papadimitriou, 1981)

Arc consistency

Associate a lower bound with each variable, initially $x_i \geq -\Omega$.

Arc consistency

Associate a lower bound with each variable, initially $x_i \geq -\Omega$.

Lower bounds can be propagated along arcs

Example: $x_1 \geq 0$, $x_1 \leq 2x_2 - 1 \implies x_2 \geq 1$

Arc consistency

Associate a lower bound with each variable, initially $x_i \geq -\Omega$.

Lower bounds can be propagated along arcs

Example: $x_1 \geq 0$, $x_1 \leq 2x_2 - 1 \implies x_2 \geq 1$

If we reach $x_i > \Omega$ for some $i \implies$ No solution exists

Arc consistency

Associate a lower bound with each variable, initially $x_i \geq -\Omega$.

Lower bounds can be propagated along arcs

Example: $x_1 \geq 0$, $x_1 \leq 2x_2 - 1 \implies x_2 \geq 1$

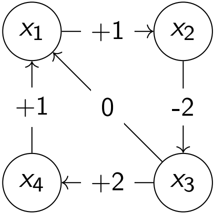
If we reach $x_i > \Omega$ for some $i \implies$ No solution exists

If no more propagations are possible $\implies$ Lower bounds form solution

Example

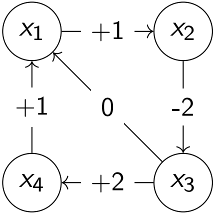
- $x_1 \leq 2x_2 - 1$
- $4x_2 + 1 \leq x_3$
- $x_3 \leq x_1 + 4$
- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$

Suppose $\Omega = 10$.



Example

- $x_1 \leq 2x_2 - 1$
- $4x_2 + 1 \leq x_3$
- $x_3 \leq x_1 + 4$
- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$

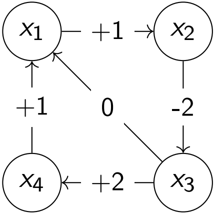


Suppose $\Omega = 10$.

$x_3 \geq -10$

Example

- $x_1 \leq 2x_2 - 1$
- $4x_2 + 1 \leq x_3$
- $x_3 \leq x_1 + 4$
- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$

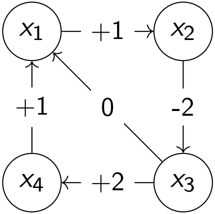


Suppose $\Omega = 10$.

$x_3 \geq -10 \implies x_4 \geq -2$

Example

- $x_1 \leq 2x_2 - 1$
- $4x_2 + 1 \leq x_3$
- $x_3 \leq x_1 + 4$
- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$

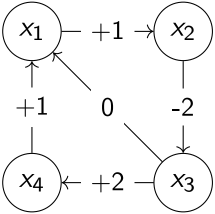


Suppose $\Omega = 10$.

$$x_3 \geq -10 \implies x_4 \geq -2 \implies x_1 \geq 0$$

Example

- $x_1 \leq 2x_2 - 1$
- $4x_2 + 1 \leq x_3$
- $x_3 \leq x_1 + 4$
- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$

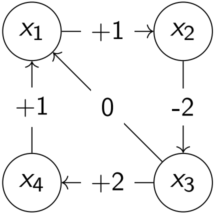


Suppose $\Omega = 10$.

$x_3 \geq -10 \implies x_4 \geq -2 \implies x_1 \geq 0 \implies x_2 \geq 1 \implies x_3 \geq 5 \implies x_4 \geq 2 \implies x_1 \geq 2 \implies x_2 \geq 2 \implies x_3 \geq 9 \implies x_1 \geq 5 \implies x_2 \geq 3 \implies x_3 \geq 12$

Example

- $x_1 \leq 2x_2 - 1$
- $4x_2 + 1 \leq x_3$
- $x_3 \leq x_1 + 4$
- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$



Suppose $\Omega = 10$.

$x_3 \geq -10 \implies x_4 \geq -2 \implies x_1 \geq 0 \implies x_2 \geq 1 \implies x_3 \geq 5 \implies x_4 \geq 2 \implies x_1 \geq 2 \implies x_2 \geq 2 \implies x_3 \geq 9 \implies x_1 \geq 5 \implies x_2 \geq 3 \implies x_3 \geq 12$

Problem: the propagation chain gets way too long ($\mathcal{O}(\Omega n)$)

Edge shortening

Consider the first two steps of the chain
$x_3 \geq -10 \implies x_4 \geq -2 \implies x_1 \geq 0$

- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$

Edge shortening

Consider the first two steps of the chain
$x_3 \geq -10 \implies x_4 \geq -2 \implies x_1 \geq 0$

- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$

This simplifies to $x_3 \leq 8x_1 - 8$

If we add this arc: $x_3 \geq -10 \implies x_1 \geq 0$

Edge shortening

Consider the first two steps of the chain
$x_3 \geq -10 \implies x_4 \geq -2 \implies x_1 \geq 0$

- $x_3 \leq 4x_4$
- $x_4 \leq 2x_1 - 2$

This simplifies to $x_3 \leq 8x_1 - 8$

If we add this arc: $x_3 \geq -10 \implies x_1 \geq 0$

Solution: preprocess graph, add new arc for every two consecutive arcs

Edge shortening

Problem: this is not always valid

- $x_3 \leq 2x_4$
- $2x_4 \leq x_1$

This does not combine into $x_3 \leq x_1$

Edge shortening

Problem: this is not always valid

- $x_3 \leq 2x_4$
- $2x_4 \leq x_1$

This does not combine into $x_3 \leq x_1$

This only happens if the first arc has positive weight and the second one has negative weight.

Edge shortening

Problem: this is not always valid

- $x_3 \leq 2x_4$
- $2x_4 \leq x_1$

This does not combine into $x_3 \leq x_1$

This only happens if the first arc has positive weight and the second one has negative weight.

Solution: only add valid arc pairs

For any three consecutive arcs, we can combine at least one pair

After $\mathcal{O}(\log(\Omega n))$ many iterations, every propagation chain reduces to one of length at most 2.

Edge shortening

Problem: this is not always valid

- $x_3 \leq 2x_4$
- $2x_4 \leq x_1$

This does not combine into $x_3 \leq x_1$

This only happens if the first arc has positive weight and the second one has negative weight.

Solution: only add valid arc pairs

For any three consecutive arcs, we can combine at least one pair

After $\mathcal{O}(\log(\Omega n))$ many iterations, every propagation chain reduces to one of length at most 2.

New problem: way too many new arcs. Coefficients get way too large.

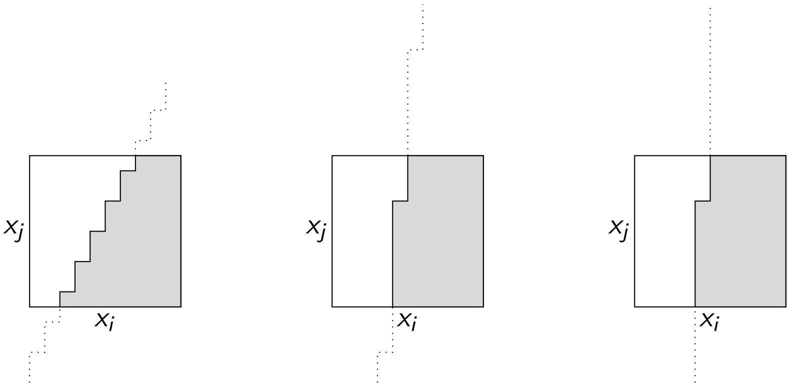
Edge compression

We can remove irrelevant arcs. We can replace arcs with large coefficients with arcs with lower coefficients.

Edge compression

We can remove irrelevant arcs. We can replace arcs with large coefficients with arcs with lower coefficients.

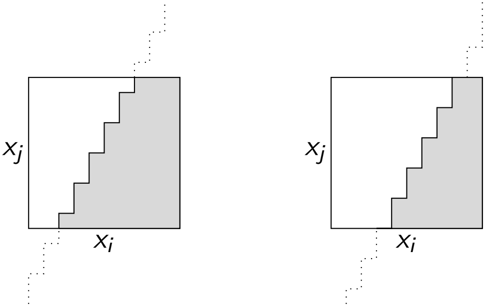
Consider arcs from x_i to x_j.



Redundant arcs

Observation: every arc is equivalent to one with weight at most $\mathcal{O}(\log(\Omega))$

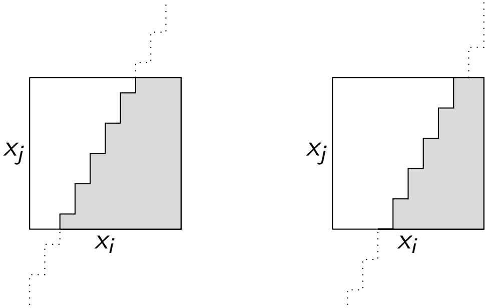
Now consider arcs from x_i to x_j with the same weight.



Redundant arcs

Observation: every arc is equivalent to one with weight at most $\mathcal{O}(\log(\Omega))$

Now consider arcs from x_i to x_j with the same weight.



We only need the arc with the most restrictive constant term

Arc count

Overall, we only need $\mathcal{O}(\log(\Omega))$ arcs between any two vertices

This means at most $\mathcal{O}(n^2 \log(\Omega))$ arcs in total.

Final algorithm

- transition into finite domain: compute Ω
- repeat $\mathcal{O}(\log(\Omega))$ times:
  - merge consecutive arcs into new arcs (if possible)
  - resize arcs with too large coefficients
  - remove irrelevant arcs
- enforce arc consistency

Runtime: $\mathcal{O}(n^8 \log^5(n))$

Final algorithm

- transition into finite domain: compute Ω
- repeat $\mathcal{O}(\log(\Omega))$ times:
  - merge consecutive arcs into new arcs (if possible)
  - resize arcs with too large coefficients
  - remove irrelevant arcs
- enforce arc consistency

Runtime: $\mathcal{O}(n^8 \log^5(n))$

Thank you for listening!