Congruency-Constrained Optimization

— at the interface of Integer Programming & Combinatorial Optimization —

Martin Nägele

Research Institute for Discrete Mathematics & HCM

University of Bonn

Based on joint past and ongoing work with
Ch. Nöbel, R. Santiago, B. Sudakov, and R. Zenklusen.

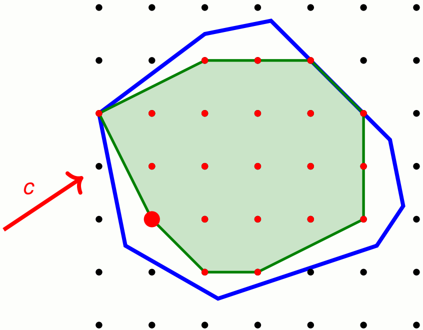
Swiss National Science Foundation

DFG Deutsche Forschungsgemeinschaft
German Research Foundation

Motivation & Background

bounded subdeterminant IPs — successes
in the bimodular case — new results

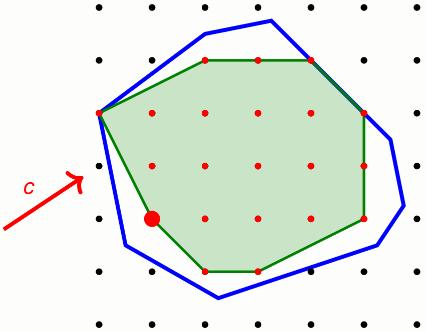
Towards general classes of efficiently solvable IPs



Integer Linear Programming (IP)

Given $A \in \mathbb{Z}^{m\times n}$, $b \in \mathbb{Z}^m$, and $c \in \mathbb{Z}^n$, solve
$$\min\{c^\top x \colon Ax \leqslant b,\ x \in \mathbb{Z}^n\}\ .$$

Towards general classes of efficiently solvable IPs



Integer Linear Programming (IP)

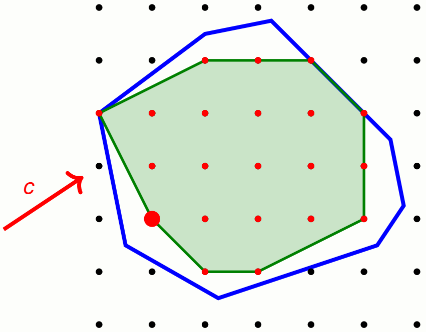
Given $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m$, and $c \in \mathbb{Z}^n$, solve

$$\min\{c^\top x : Ax \leqslant b,\ x \in \mathbb{Z}^n\} \ .$$

An interesting class of efficiently solvable IPs

A totally unimodular (TU) $\implies$ Integral relaxation.

Towards general classes of efficiently solvable IPs



Integer Linear Programming (IP)

Given $A \in \mathbb{Z}^{m\times n}$, $b \in \mathbb{Z}^m$, and $c \in \mathbb{Z}^n$, solve

$$\min\{c^\top x \colon Ax \leqslant b,\ x \in \mathbb{Z}^n\} \ .$$

An interesting class of efficiently solvable IPs

A totally unimodular (TU) $\implies$ Integral relaxation.

What if minors, in absolute value, are still bounded, but not by 1?

Bounded subdeterminants

Δ-modular Integer Programming

Given a constant $\Delta \in \mathbb{Z}_{>0}$, can integer linear programs

$$\min\{c^\top x \colon Ax \leq b,\ x \in \mathbb{Z}^n\}$$

with Δ-modular constraint matrix A be solved efficiently?

- $A \in \mathbb{Z}^{m \times n}$ is Δ-modular if
  - → $\operatorname{rank}(A) = n$, and
  - → absolute values of $n \times n$ subdeterminants are bounded by Δ
- Δ-modularity is more general than *total* Δ-modularity

Bounded subdeterminants

Δ-modular Integer Programming

Given a constant $\Delta \in \mathbb{Z}_{>0}$, can integer linear programs

$$\min\{c^\top x \colon Ax \leq b,\ x \in \mathbb{Z}^n\}$$

with Δ-modular constraint matrix A be solved efficiently?

- $A \in \mathbb{Z}^{m \times n}$ is Δ-modular if
  - → $\operatorname{rank}(A) = n$, and
  - → absolute values of $n \times n$ subdeterminants are bounded by Δ
- Δ-modularity is more general than *total* Δ-modularity

Known results

✓ $\Delta = 1$: easy

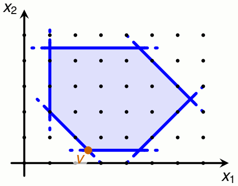
✓ $\Delta = 2$: Bimodular Integer Programming (BIP)

[Artmann, Weismantel, and Zenklusen, STOC 2017]

✓ Arbitrary constant Δ, at most 2 non-zeros per row

[Fiorini, Joret, Weltge, and Yuditsky, FOCS 2021]

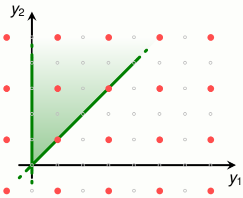
The approach to BIP



Bimodular Integer Program (BIP)

$$\min\{c^\top x \colon Ax \leqslant b, x \in \mathbb{Z}^n\}$$

A bimodular.



Conic Parity TU Problem (CPTU)

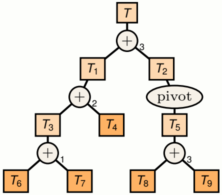
$$\min\{\tilde{c}^\top y \colon Ty \leqslant 0, y \in \mathbb{Z}^n, y(S) \text{ odd}\}$$

T totally unimodular, $S \subseteq [n]$.

Theorem

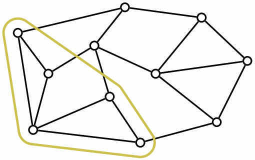
BIP can be solved in strongly polynomial time.

[Artmann, Weismantel, and Zenklusen, STOC 2017]



Seymour's TU decomposition

Reduction to base block problems.



Base block problems

Interpretation as parity-constrained cut and circulation problems.

Challenge: Generalize!

Bimodular Integer Program (BIP)

$\min\{c^\top x \colon Ax \leqslant b, x \in \mathbb{Z}^n\}$
A bimodular.

Conic Parity TU Problem (CPTU)

$\min\{\tilde{c}^\top y \colon Ty \leqslant 0, y \in \mathbb{Z}^n, y(S) \text{ odd}\}$
T totally unimodular, $S \subseteq [n]$.

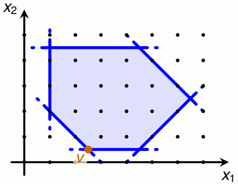
Seymour's TU decomposition

Reduction to
base block problems.

Base block problems

Interpretation as parity-constrained
cut and circulation problems

Challenge: Generalize!



Δ-modular integer programming

$$\min\{c^\top x \colon Ax \leqslant b, x \in \mathbb{Z}^n\}$$

A is Δ-modular.

Conic Parity TU Problem (CPTU)

$$\min\{\tilde{c}^\top y \colon Ty \leqslant 0, y \in \mathbb{Z}^n, y(S) \text{ odd}\}$$

T totally unimodular, $S \subseteq [n]$.

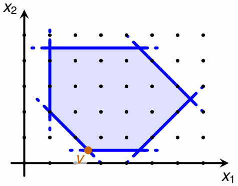
Seymour's TU decomposition

Reduction to
base block problems.

Base block problems

Interpretation as parity-constrained
cut and circulation problems

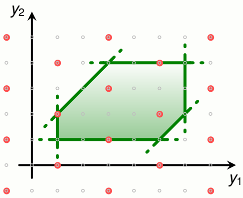
Challenge: Generalize!



Δ-modular integer programming

$\min\{c^\top x \colon Ax \leqslant b, x \in \mathbb{Z}^n\}$

A is Δ-modular.



Congruency-Constr. TU Prb. (CCTU)

$$\min\left\{\tilde{c}^\top y \colon \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array}\right\}$$

T totally unimodular, modulus m.

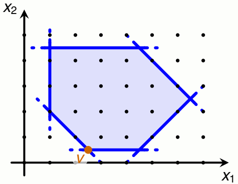
Seymour's TU decomposition

Reduction to
base block problems.

Base block problems

Interpretation as parity-constrained
cut and circulation problems

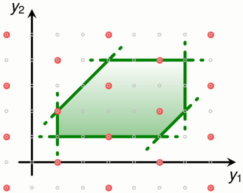
Challenge: Generalize!



Δ-modular integer programming

$$\min\{c^\top x \colon Ax \leqslant b, x \in \mathbb{Z}^n\}$$

A is Δ-modular.



Congruency-Constr. TU Prb. (CCTU)

$$\min\left\{\tilde{c}^\top y \colon \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array}\right\}$$

T totally unimodular, modulus m.

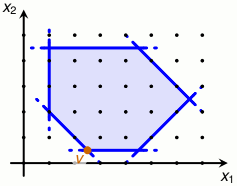
Seymour's TU decomposition

Reduction to base block problems.

Base block problems

Interpretation as parity-constrained cut and circulation problems

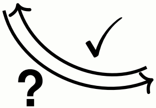
Challenge: Generalize!

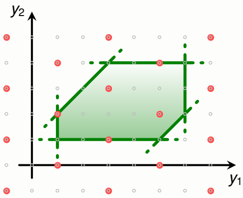


Δ-modular integer programming

$$\min\{c^\top x \colon Ax \leqslant b, x \in \mathbb{Z}^n\}$$

A is Δ-modular.





Congruency-Constr. TU Prb. (CCTU)

$$\min\left\{\tilde{c}^\top y \colon \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array}\right\}$$

T totally unimodular, modulus m.

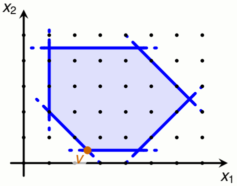
Seymour's TU decomposition

Reduction to base block problems.

Base block problems

Interpretation as parity-constrained cut and circulation problems

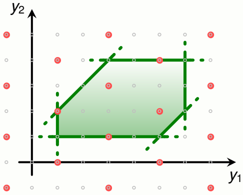
Challenge: Generalize!



Δ-modular integer programming

$$\min\{c^\top x : Ax \leqslant b, x \in \mathbb{Z}^n\}$$

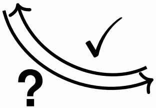
A is Δ-modular.



Congruency-Constr. TU Prb. (CCTU)

$$\min \left\{ \tilde{c}^\top y : \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array} \right\}$$

T totally unimodular, modulus m.



Equivalence:
strictly Δ-modular IP
(subdets in $\{0, \pm\Delta\}$)
if Δ prime

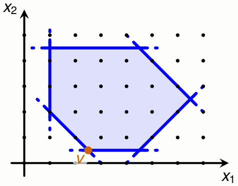
Seymour's TU decomposition

Reduction to
base block problems.

Base block problems

Interpretation as parity-constrained
cut and circulation problems

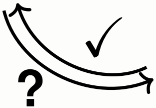
Challenge: Generalize!



Δ-modular integer programming

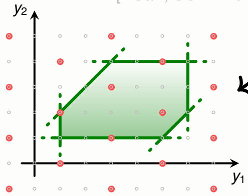
$\min\{c^\top x \colon Ax \leqslant b, x \in \mathbb{Z}^n\}$

A is Δ-modular.



Equivalence:
strictly Δ-modular IP
(subdets in $\{0, \pm\Delta\}$)
if Δ prime

Structural results:
$\rightarrow$ proximity
$\rightarrow$ flatness or feasibility
[NSaZ, SODA 2022]



Congruency-Constr. TU Prb. (CCTU)

$$\min \left\{ \tilde{c}^\top y \colon \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array} \right\}$$

T totally unimodular, modulus m.

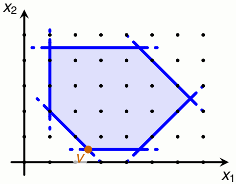
Seymour's TU decomposition

Reduction to
base block problems.

Base block problems

Interpretation as parity-constrained
cut and circulation problems

Challenge: Generalize!



Δ-modular integer programming

$\min\{c^\top x \colon Ax \leqslant b, x \in \mathbb{Z}^n\}$

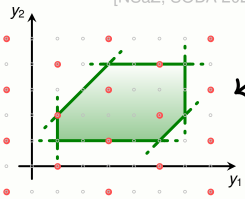
A is Δ-modular.

Structural results:

→ proximity

→ flatness or feasibility

[NSaZ, SODA 2022]



Congruency-Constr. TU Prb. (CCTU)

$$\min\left\{\tilde{c}^\top y \colon \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array}\right\}$$

T totally unimodular, modulus m.

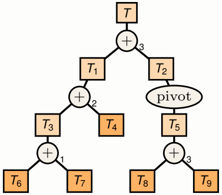
?

Equivalence:

strictly Δ-modular IP

(subdets in $\{0, \pm\Delta\}$)

if Δ prime



Seymour's TU decomposition

Reduction gets much more involved!

Generalization: $m = 3$, feasibility

→ hierarchy of problems

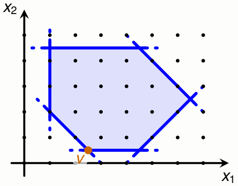
→ powerful tools, e.g., Cauchy-Davenport

[NSaZ, SODA 2022]

Base block problems

Interpretation as parity-constrained cut and circulation problems

Challenge: Generalize!



Δ-modular integer programming

$\min\{c^\top x : Ax \leqslant b, x \in \mathbb{Z}^n\}$

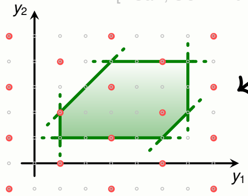
A is Δ-modular.

Structural results:

$\rightarrow$ proximity

$\rightarrow$ flatness or feasibility

[NSaZ, SODA 2022]



Congruency-Constr. TU Prb. (CCTU)

$$\min\left\{\tilde{c}^\top y : \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array}\right\}$$

T totally unimodular, modulus m.

✓

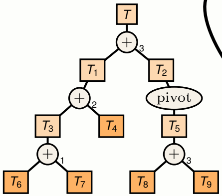
?

Equivalence:

strictly Δ-modular IP

(subdets in $\{0, \pm\Delta\}$)

if Δ prime



Seymour's TU decomposition

Reduction gets much more involved!

Generalization: $m = 3$, feasibility

$\rightarrow$ hierarchy of problems

$\rightarrow$ powerful tools, e.g., Cauchy-Davenport

[NSaZ, SODA 2022]

Cut baseblock:

$\rightarrow$ optim. for prime power m

[NSuZ, SODA 2018]

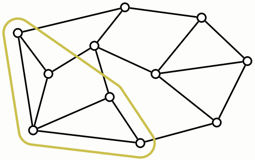
$\rightarrow$ feasibility for general m

[NNSaZ, 2022+]

Circulation baseblock:

$\rightarrow$ rand. alg. for unary enc. obj.

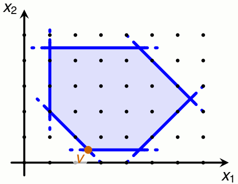
[NSaZ, SODA 2022]



Base block problems

Interpretation as congruency-constrained cut and circulation problems

Challenge: Generalize!



Δ-modular integer programming

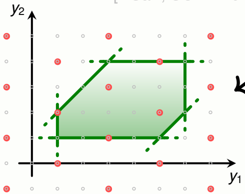
$\min\{c^\top x \colon Ax \leqslant b, x \in \mathbb{Z}^n\}$

A is Δ-modular.

✓ ?

Equivalence:
strictly Δ-modular IP
(subdets in $\{0, \pm\Delta\}$)
if Δ prime

Structural results:
$\rightarrow$ proximity
$\rightarrow$ flatness or feasibility
[NSaZ, SODA 2022]



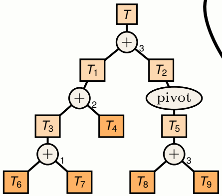
Congruency-Constr. TU Prb. (CCTU)

$$\min\left\{\tilde{c}^\top y \colon \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array}\right\}$$

T totally unimodular, modulus m.

Corollary

Strongly polynomial randomized algorithm for checking feasibility of strictly 3-modular IPs.



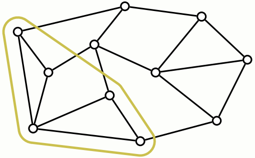
Seymour's TU decomposition

Reduction gets much more involved!

Generalization: $m = 3$, feasibility
$\rightarrow$ hierarchy of problems
$\rightarrow$ powerful tools, e.g., Cauchy-Davenport
[NSaZ, SODA 2022]

Cut baseblock:
$\rightarrow$ optim. for prime power m
[NSuZ, SODA 2018]
$\rightarrow$ feasibility for general m
[NNSaZ, 2022+]

Circulation baseblock:
$\rightarrow$ rand. alg. for unary enc. obj.
[NSaZ, SODA 2022]



Base block problems

Interpretation as congruency-constrained cut and circulation problems

Challenge: Generalize!

Δ-modular integer programming

$\min\{c^\top x : Ax \leqslant b, x \in \mathbb{Z}^n\}$
A is Δ-modular.

Structural results:
$\to$ proximity
$\to$ flatness or feasibility
[NSaZ, SODA 2022]

Congruency-Constr. TU Prb. (CCTU)

$$\min\left\{\tilde{c}^\top y : \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array}\right\}$$

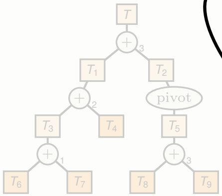
T totally unimodular, modulus m.

?

Equivalence:
strictly Δ-modular IP
(subdets in $\{0, \pm\Delta\}$)
if Δ prime

Corollary

Strongly polynomial randomized algorithm for checking feasibility of strictly 3-modular IPs.



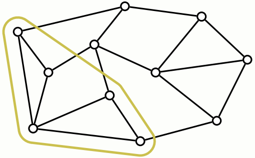
Seymour's TU decomposition

Reduction gets much more involved!

Generalization: $m = 3$, feasibility
$\to$ hierarchy of problems
$\to$ powerful tools, e.g., Cauchy-Davenport
[NSaZ, SODA 2022]

Cut baseblock:
$\to$ optim. for prime power m
[NSuZ, SODA 2018]
$\to$ feasibility for general m
[NNSaZ, 2022+]

Circulation baseblock:
$\to$ rand. alg. for unary enc. obj.
[NSaZ, SODA 2022]



Base block problems

Interpretation as congruency-constrained cut and circulation problems

Base Block Problems

Seymour's decomposition — network matrices — the two cases

Seymour's decomposition

Theorem: Seymour's decomposition [Seymour, 1980]

For every TU matrix $T \in \mathbb{Z}^{k \times n}$, one of the following applies:

(i) T is, possibly after row/column permutations and a pivot, of the form

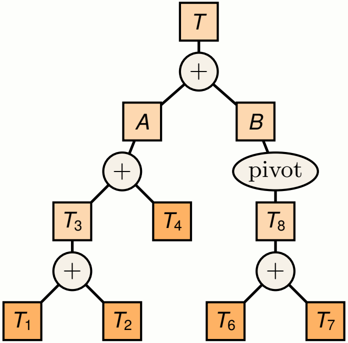
$$\begin{pmatrix} A & ef^\top \\ gh^\top & B \end{pmatrix},$$

where $\begin{pmatrix} A & e & e \\ h^\top & 0 & 1 \end{pmatrix}$ and $\begin{pmatrix} B & g & g \\ f^\top & 0 & 1 \end{pmatrix}$ are TU.

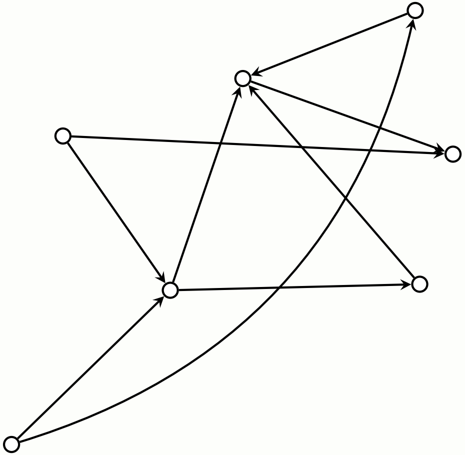
(ii) T is essentially equal to one of

$$\begin{pmatrix} 1 & -1 & 0 & 0 & -1 \\ -1 & 1 & -1 & 0 & 0 \\ 0 & -1 & 1 & -1 & 0 \\ 0 & 0 & -1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 1 \end{pmatrix} \text{ and } \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

(iii) T or $T^\top$ is a network matrix.

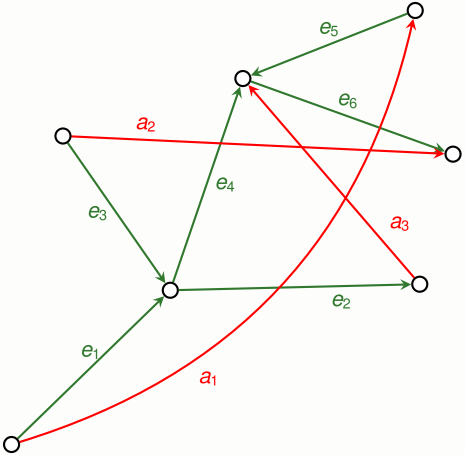


Network matrices



$$\left\{ \begin{array}{rrr} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{array} \right\}$$

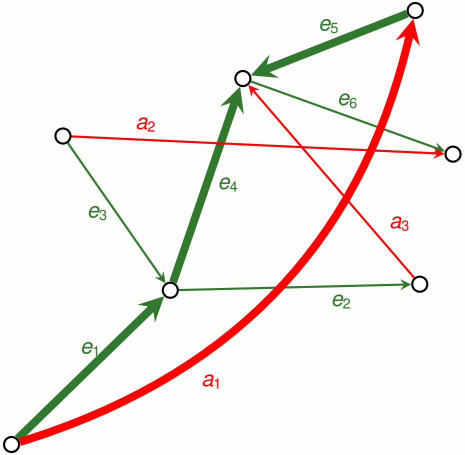
Network matrices



Tree $T = (V, E)$, extra arcs A.

$$
\begin{array}{c c}
 & \begin{array}{ccc} a_1 & a_2 & a_3 \end{array} \\
\begin{array}{c} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{array} &
\left\{ \begin{array}{rrr}
1 & 0 & 0 \\
0 & 0 & -1 \\
0 & 1 & 0 \\
1 & 1 & 1 \\
-1 & 0 & 0 \\
0 & 1 & 0
\end{array} \right\}
\end{array}
$$

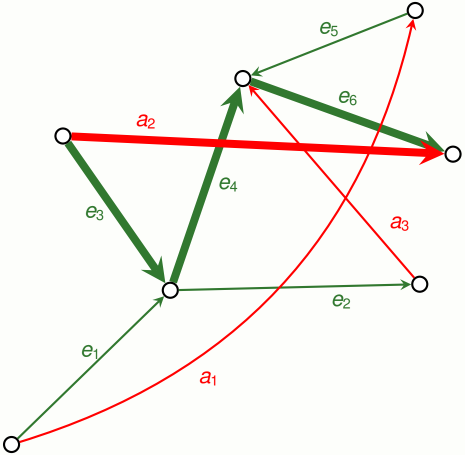
Network matrices



Tree $T = (V, E)$, extra arcs A.

	a_1	a_2	a_3
e_1	1	0	0
e_2	0	0	-1
e_3	0	1	0
e_4	1	1	1
e_5	-1	0	0
e_6	0	1	0

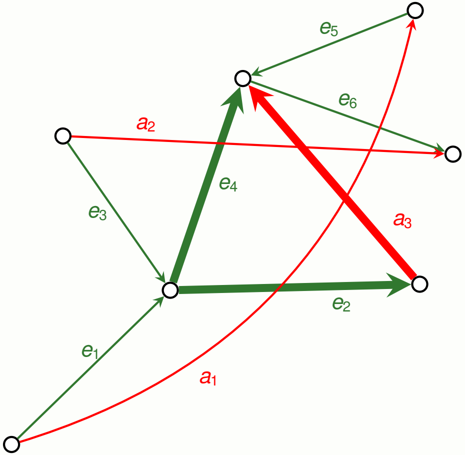
Network matrices



Tree $T = (V, E)$, extra arcs A.

	a_1	a_2	a_3
e_1	1	0	0
e_2	0	0	−1
e_3	0	1	0
e_4	1	1	1
e_5	−1	0	0
e_6	0	1	0

Network matrices



Tree $T = (V, E)$, extra arcs A.

	a_1	a_2	a_3
e_1	1	0	0
e_2	0	0	-1
e_3	0	1	0
e_4	1	1	1
e_5	-1	0	0
e_6	0	1	0

CCTU with transposed network matrices

“the cut baseblock”

CCTU with transposed network constraint matrices

CCTU with transp. netw. matrix

$$\min\{\tilde{c}^\top y \colon Ty \leqslant b, \gamma^\top y \equiv r \pmod{m}\}$$

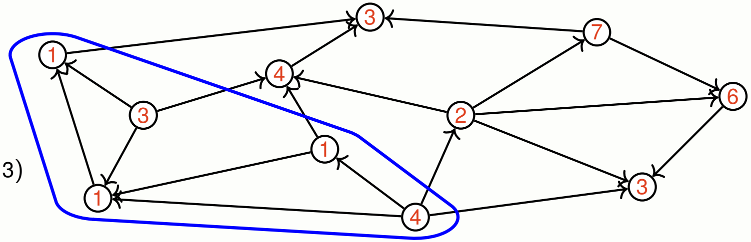
with transposed network matrix T

Congruency-constrained min (di-)cut

$$\min_{C \subsetneq V}\{|\delta^+(C)| \colon \delta^-(C) = \emptyset, \gamma(C) \equiv r \pmod{m}\}$$

on digraph $G = (V, A)$ with $\gamma \colon V \to \mathbb{Z}$.

$\gamma(C) \equiv 1 \pmod{3}$



Prime power moduli

Theorem

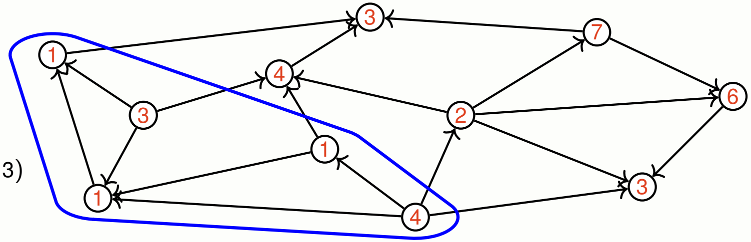
Congruency-constrained min-cut is poly-time solvable for constant prime power m.

[NSuZ, SODA 2018]

Algorithm

- Guess $m-1$ elements in- and outside OPT
- Solve corresp. unconstrained min cut problem
- Return best cong-constraint feasible solution

$\gamma(C) \equiv 1 \pmod 3$



Prime power moduli

Theorem

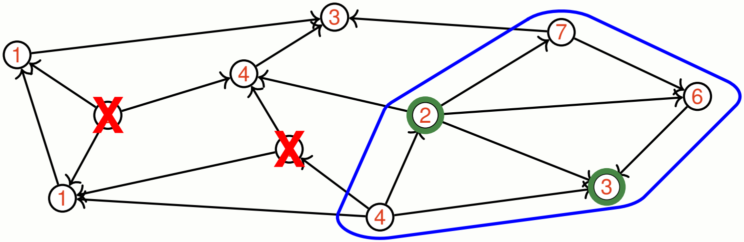
Congruency-constrained min-cut is poly-time solvable for constant prime power m.

[NSuZ, SODA 2018]

Algorithm

- Guess $m-1$ elements in- and outside OPT
- Solve corresp. unconstrained min cut problem
- Return best cong-constraint feasible solution

$\gamma(C) \not\equiv 1 \pmod{3}$



Prime power moduli

Theorem

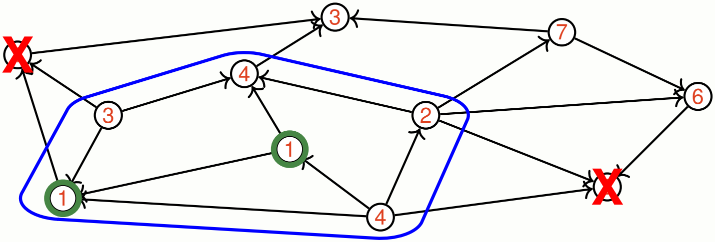
Congruency-constrained min-cut is poly-time solvable for constant prime power m.

[NSuZ, SODA 2018]

Algorithm

- Guess $m-1$ elements in- and outside OPT
- Solve corresp. unconstrained min cut problem
- Return best cong-constraint feasible solution

$\gamma(C) \equiv 1 \pmod 3$



Prime power moduli

Theorem

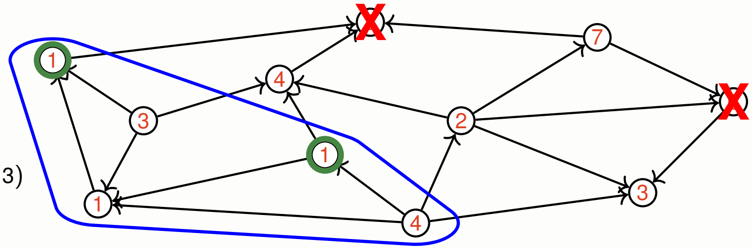
Congruency-constrained min-cut is poly-time solvable for constant prime power m.

[NSuZ, SODA 2018]

Algorithm

- Guess $m-1$ elements in- and outside OPT
- Solve corresp. unconstrained min cut problem
- Return best cong-constraint feasible solution

$\gamma(C) \equiv 1 \pmod 3$



Prime power moduli

Theorem

Congruency-constrained min-cut is poly-time solvable for constant prime power m.

[NSuZ, SODA 2018]

Algorithm

- Guess $m-1$ elements in- and outside OPT
- Solve corresp. unconstrained min cut problem
- Return best cong-constraint feasible solution

Proof idea:

- Failing guesses give structured set system.
- Such systems cannot exist for prime power m.

The feasibility problem

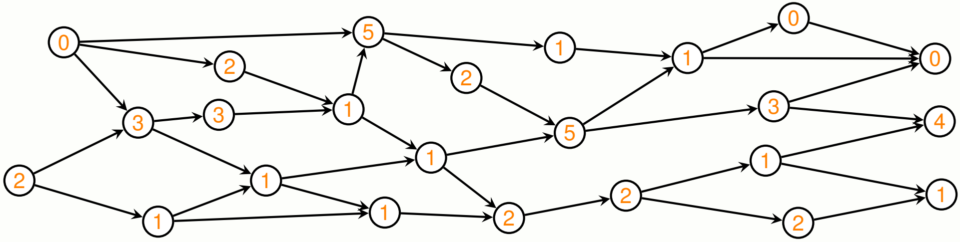
Congruency-constrained lattice feasibility

In $G = (V, A)$ with $\gamma \colon V \to \mathbb{Z}$, find
$C \subsetneq V \colon \delta^-(C) = \emptyset, \gamma(C) \equiv r \pmod{m}$.

Theorem

Congruency-constr. lattice feasibility can be decided in poly time for constant m.

[NNSaZ, 2022+]



The feasibility problem

Congruency-constrained lattice feasibility

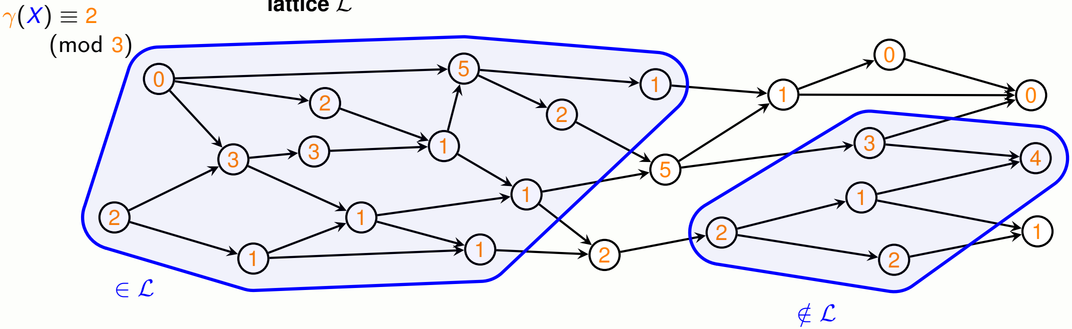
In $G = (V, A)$ with $\gamma\colon V \to \mathbb{Z}$, find
$C \subsetneq V\colon \delta^-(C) = \emptyset, \gamma(C) \equiv r \pmod{m}$.

lattice $\mathcal{L}$

Theorem

Congruency-constr. lattice feasibility can be decided in poly time for constant m.

[NNSaZ, 2022+]



The feasibility problem

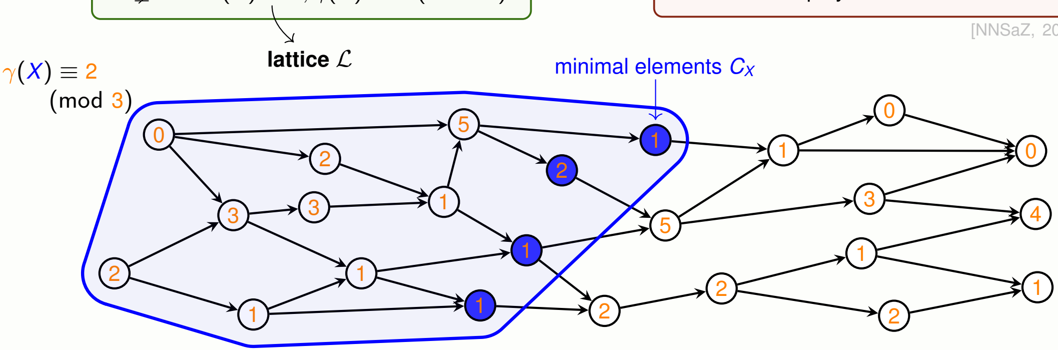
Congruency-constrained lattice feasibility

In $G = (V, A)$ with $\gamma\colon V \to \mathbb{Z}$, find $C \subsetneq V\colon \delta^-(C) = \emptyset, \gamma(C) \equiv r \pmod{m}$.

Theorem

Congruency-constr. lattice feasibility can be decided in poly time for constant m.

[NNSaZ, 2022+]



The feasibility problem

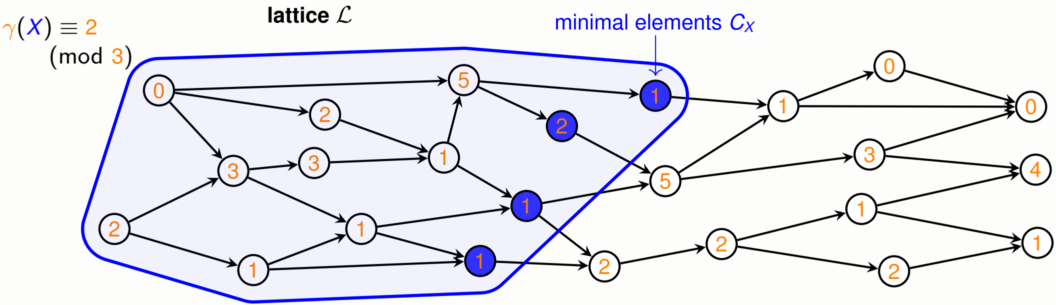
Congruency-constrained lattice feasibility

In $G = (V, A)$ with $\gamma \colon V \to \mathbb{Z}$, find
$C \subsetneq V \colon \delta^-(C) = \emptyset, \gamma(C) \equiv r \pmod{m}$.

Theorem

Congruency-constr. lattice feasibility can be decided in poly time for constant m.

[NNSaZ, 2022+]



There is a feasible $X \in \mathcal{L}$ with $|C_X| \leq m - 1$.

The feasibility problem

Congruency-constrained lattice feasibility

In $G = (V, A)$ with $\gamma \colon V \to \mathbb{Z}$, find $C \subsetneq V \colon \delta^-(C) = \emptyset, \gamma(C) \equiv r \pmod{m}$.

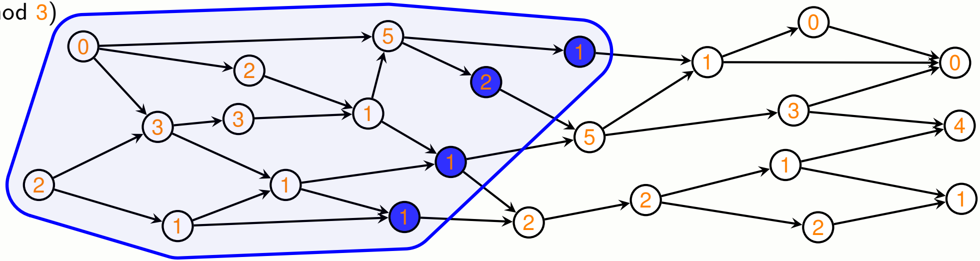
lattice $\mathcal{L}$

Theorem

Congruency-constr. lattice feasibility can be decided in poly time for constant m.

[NNSaZ, 2022+]

$\gamma(X) \equiv 2 \pmod{3}$



There is a feasible $X \in \mathcal{L}$ with $|C_X| \leq m - 1$.

Among any $\geq m$ numbers, there is a subset with sum $\equiv 0 \pmod{m}$.

The feasibility problem

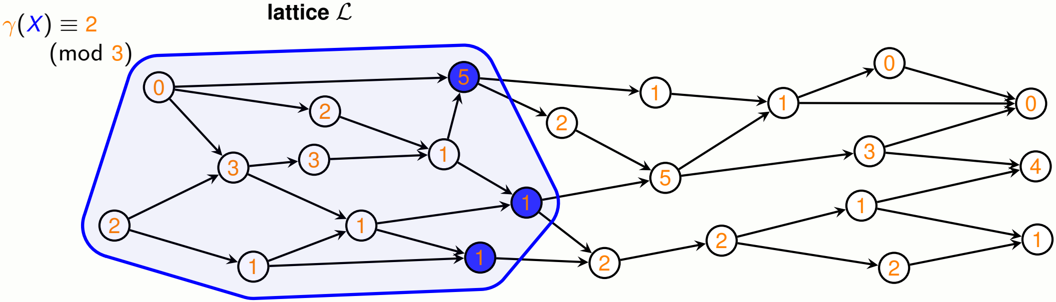
Congruency-constrained lattice feasibility

In $G = (V, A)$ with $\gamma\colon V \to \mathbb{Z}$, find $C \subsetneq V\colon \delta^-(C) = \emptyset, \gamma(C) \equiv r \pmod{m}$.

Theorem

Congruency-constr. lattice feasibility can be decided in poly time for constant m.

[NNSaZ, 2022+]



There is a feasible $X \in \mathcal{L}$ with $|C_X| \leq m - 1$.

Among any $\geq m$ numbers, there is a subset with sum $\equiv 0 \pmod{m}$.

The feasibility problem

Congruency-constrained lattice feasibility

In $G = (V, A)$ with $\gamma\colon V \to \mathbb{Z}$, find
$C \subsetneq V\colon \delta^-(C) = \emptyset, \gamma(C) \equiv r \pmod{m}$.

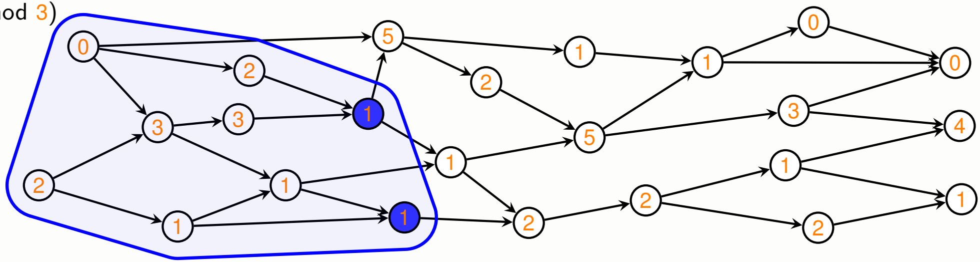
lattice $\mathcal{L}$

Theorem

Congruency-constr. lattice feasibility can be decided in poly time for constant m.

[NNSaZ, 2022+]

$\gamma(X) \equiv 2 \pmod{3}$



There is a feasible $X \in \mathcal{L}$ with $|C_X| \leq m - 1$.

Among any $\geq m$ numbers, there is a subset with sum $\equiv 0 \pmod{m}$.

CCTU with network matrices

"the circulation baseblock"

CCTU with network constraint matrices: Circulations

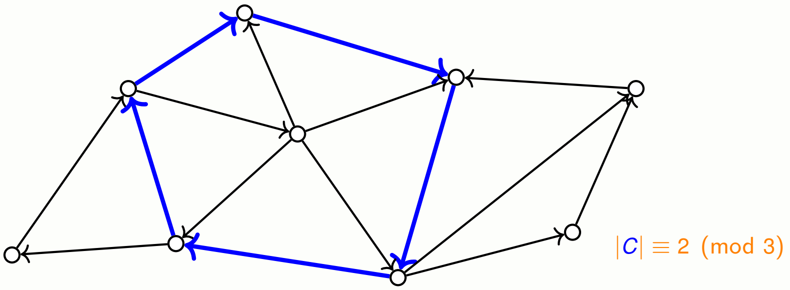
CCTU with transp. netw. matrix

$\min\{\tilde{c}^\top y \colon Ty \leqslant b, \gamma^\top y \equiv r \pmod{m}\}$
with network matrix T

⟷

Congruency-constrained circulation

In digraph $G = (V, A)$, find shortest circulation $C \subseteq A$ with $|C| \equiv r \pmod{m}$.



CCTU with network constraint matrices: Circulations

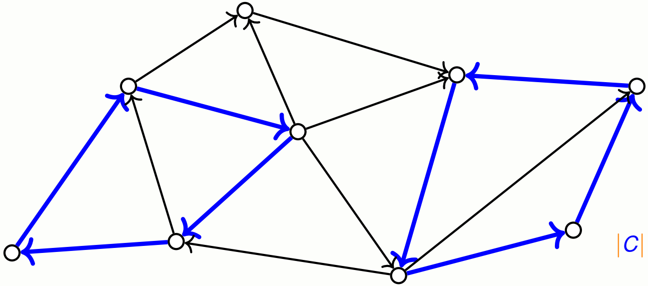
CCTU with transp. netw. matrix

$\min\{\tilde{c}^\top y \colon Ty \leqslant b, \gamma^\top y \equiv r \pmod{m}\}$
with network matrix T

$\rightsquigarrow$

Congruency-constrained circulation

In digraph $G = (V, A)$, find shortest circulation $C \subseteq A$ with $|C| \equiv r \pmod{m}$.



$|C| \equiv 2 \pmod{3}$

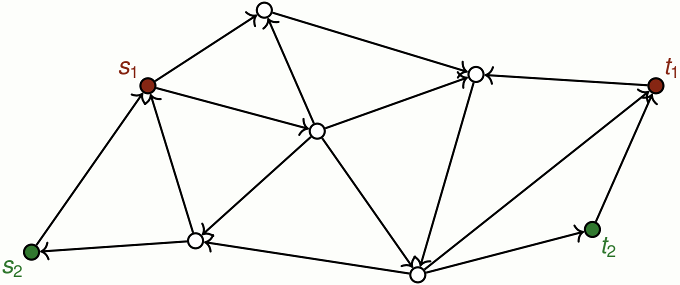
Circulations vs. cycles

Finding congruency-constrained *cycles* is hard for $m > 2$!

Circulations vs. cycles

Finding congruency-constrained *cycles* is hard for $m > 2$!

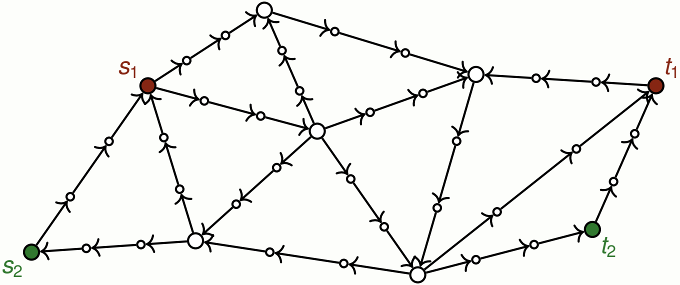
- Reduction to arc-disjoint paths:



Circulations vs. cycles

Finding congruency-constrained *cycles* is hard for $m > 2$!

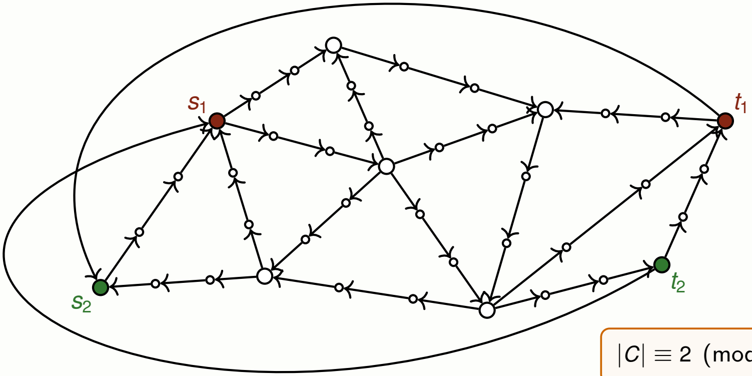
- Reduction to arc-disjoint paths:



Circulations vs. cycles

Finding congruency-constrained *cycles* is hard for $m > 2$!

- Reduction to arc-disjoint paths:



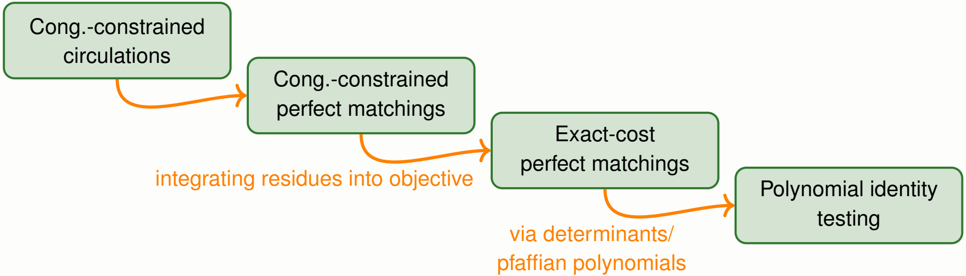
$|C| \equiv 2 \pmod 3 \iff$ paths exist

A reduction

Theorem

∃ strongly poly. randomized alg. for congruency-constrained circulations with unary encoded edge lengths and constant m.

- Our approach:



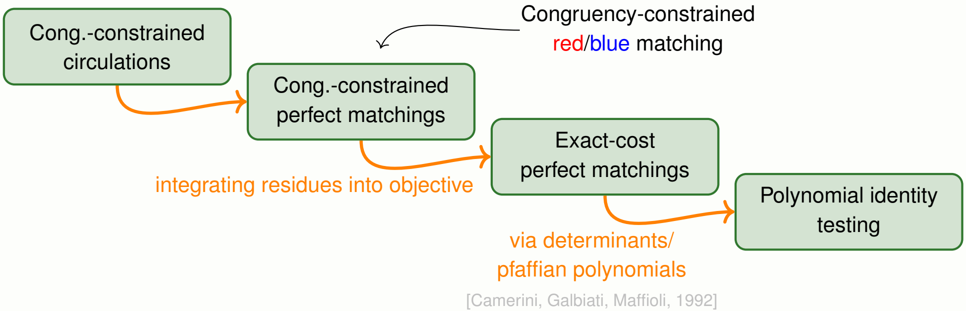
[Camerini, Galbiati, Maffioli, 1992]

A reduction

Theorem

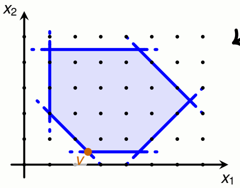
∃ strongly poly. randomized alg. for congruency-constrained circulations with unary encoded edge lengths and constant m.

- Our approach:



Open questions

Open questions

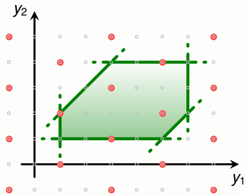


Structural results?

Δ-modular integer programming

$\min\{c^\top x\colon Ax \leqslant b, x \in \mathbb{Z}^n\}$

A is Δ-modular.



Congruency-Constr. TU Prb. (CCTU)

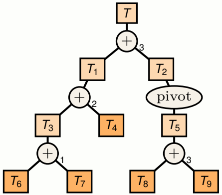
$$\min\left\{\tilde{c}^\top y\colon \begin{array}{c} Ty \leqslant b,\ y \in \mathbb{Z}^n, \\ \gamma^\top y \equiv r \pmod{m} \end{array}\right\}$$

T totally unimodular, modulus m.

?

- What problem should we reduce to?
- How to deal with several differing subdets?

Do we need to go through Seymour's decomposition?

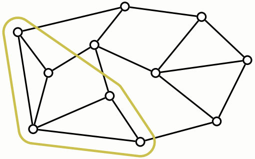


Seymour's TU decomposition

Reduction to base block problems.

- Optimization?
- General m?

- Beyond $m = p^\alpha$ for cut optimization?
- Deterministic approach for circulations?



Base block problems

Interpretation as parity-constrained cut and circulation problems.