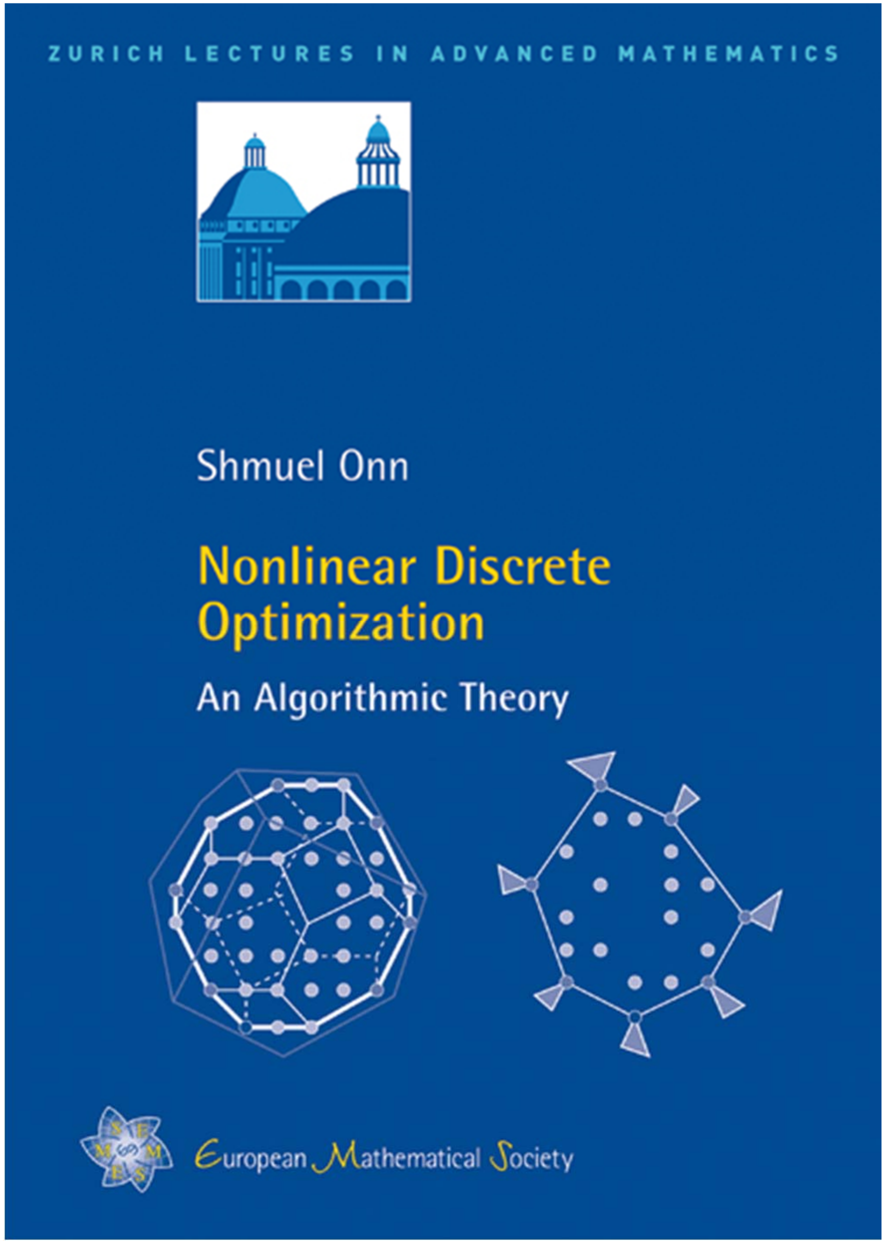
Shifted
Combinatorial Optimization

Shmuel Onn

Technion – Israel Institute of Technology



Acknowledgment and little advertisement

Monograph developed at ETH while delivering the Nachdiplom Lectures

Theory of Graver bases for integer programming

Available electronically from my homepage
(with kind permission of EMS)

1. Shifted Combinatorial Optimization

2. Multicommodity Flows

Standard Combinatorial Optimization

Standard Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $w \in Z^d$, solve

$$\max \{ ws \ : \ s \in S \}$$

Standard Combinatorial Optimization

Standard Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $w \in Z^d$, solve

$$\max \{ ws \ : \ s \in S \}$$

For instance:

- (indicators of) Perfect matchings in a graph with d edges

Standard Combinatorial Optimization

Standard Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $w \in Z^d$, solve

$$\max \{ ws \ : \ s \in S \}$$

For instance:

- (indicators of) Perfect matchings in a graph with d edges

- Dipaths from u to v in a digraph

Standard Combinatorial Optimization

Standard Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $w \in Z^d$, solve

$$\max\{\, ws \;:\; s \in S \,\}$$

For instance:

- (indicators of) Perfect matchings in a graph with d edges
- Dipaths from u to v in a digraph
- Set $S = \{s \in \{0,1\}^d : As \leq b\}$ given by inequalities (and general IP)

Standard Combinatorial Optimization

Standard Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $w \in Z^d$, solve

$$\max \{ ws \; : \; s \in S \}$$

For instance:

- (indicators of) Perfect matchings in a graph with d edges
- Dipaths from u to v in a digraph
- Set $S = \{s \in \{0,1\}^d : As \le b\}$ given by inequalities (and general IP)
- Bases in a matroid over [d] given by independence oracle

Standard Combinatorial Optimization

Standard Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $w \in Z^d$, solve

$$\max \{ ws \ : \ s \in S \}$$

For instance:

- (indicators of) Perfect matchings in a graph with d edges
- Dipaths from u to v in a digraph
- Set $S = \{s \in \{0,1\}^d : As \leq b\}$ given by inequalities (and general IP)
- Bases in a matroid over [d] given by independence oracle
- Intersection of matroids given by oracles

Standard Combinatorial Optimization

Standard Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $w \in Z^d$, solve

$$\max \{ ws \ : \ s \in S \}$$

For instance:

- (indicators of) Perfect matchings in a graph with d edges
- Dipaths from u to v in a digraph
- Set $S = \{s \in \{0,1\}^d : As \leq b\}$ given by inequalities (and general IP)
- Bases in a matroid over [d] given by independence oracle
- Intersection of matroids given by oracles
- Set $S = \{s^1, s^2, \ldots, s^m\} \subseteq \{0,1\}^d$ given explicitly

Shifted Combinatorial Optimization

For $S \subseteq \{0,1\}^d$ consider matrices consisting of n choices from S,

$$S^n = \{ x \in \{0,1\}^{d \times n} \; : \; x^k \in S, \quad k=1,\dots,n \}$$

Shifted Combinatorial Optimization

For $S \subseteq \{0,1\}^d$ consider matrices consisting of n choices from S,

$$S^n = \{ x \in \{0,1\}^{d \times n} \;:\; x^k \in S, \quad k=1,\ldots,n \}$$

The shift of matrix $x \in R^{d \times n}$ is the matrix $\underline{x}$ obtained from x by permuting each row of x to be nonincreasing

Shifted Combinatorial Optimization

For $S \subseteq \{0,1\}^d$ consider matrices consisting of n choices from S,

$$S^n = \{\ x \in \{0,1\}^{d \times n} \ :\ x^k \in S, \quad k=1,\ldots,n\ \}$$

The shift of matrix $x \in R^{d \times n}$ is the matrix $\underline{x}$ obtained from x by permuting each row of x to be nonincreasing

Shifted Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $c \in Z^{d \times n}$, solve

$$\max \{\ c\underline{x} \ :\ x \in S^n\ \}$$

One Motivation: Minimizing Vulnerable Elements

Element $i \in [d]$ is r-vulnerable in x if used by at least r choices x^k

One Motivation: Minimizing Vulnerable Elements

Element $i \in [d]$ is r-vulnerable in x if used by at least r choices x^k

Then $\underline{x}^r$ is exactly the indicator of the r-vulnerable elements in x

One Motivation: Minimizing Vulnerable Elements

Element $i \in [d]$ is r-vulnerable in x if used by at least r choices x^k

Then $\underline{x}^r$ is exactly the indicator of the r-vulnerable elements in x

Maximizing $c\underline{x}$ for suitable c may allow the survival of some of the n choices under various attacks of vulnerable elements by adversary

One Motivation: Minimizing Vulnerable Elements

Element $i \in [d]$ is r-vulnerable in x if used by at least r choices x^k

Then $\underline{x}^r$ is exactly the indicator of the r-vulnerable elements in x

Maximizing $c\underline{x}$ for suitable c may allow the survival of some of the n choices under various attacks of vulnerable elements by adversary

For instance, by suitable shifting we can:

- minimize the number of r-vulnerable elements

- lexicographically minimize n-vulnerables, $(n-1)$-vulnerables, and so on

One Motivation: Minimizing Vulnerable Elements

Element $i \in [d]$ is r-vulnerable in x if used by at least r choices x^k

Then $\underline{x}^r$ is exactly the indicator of the r-vulnerable elements in x

Maximizing $c\underline{x}$ for suitable c may allow the survival of some of the n choices under various attacks of vulnerable elements by adversary

For instance, by suitable shifting we can:

- minimize the number of r-vulnerable elements

  (set $c_{i,r} = -1$, $c_{i,j} = 0$, $i=1,\dots,d$, $j \neq r$)

- lexicographically minimize n-vulnerables, $(n-1)$-vulnerables, and so on

  (set $c_{i,j} = -(d+1)^{(j-1)}$, $i=1,\dots,d$, $j=1,\dots,n$)

Special Case: Minimum Shared Edges Problem

Special case of choosing n dipaths from u to v in a digraph to minimize 2-vulnerable edges (shared edges) studied recently a lot, for example:

Special Case: Minimum Shared Edges Problem

Special case of choosing n dipaths from u to v in a digraph to minimize 2-vulnerable edges (shared edges) studied recently a lot, for example:

- For n variable it is NP-hard (O-S-Z 2013)
- For n fixed it is polynomial time solvable (A-E-N-Y-Z 2014)
- For n parameter it is fixed-parameter tractable (F-K-N-S 2015)

Matchings, Dipaths, Inequalities

Shifted Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $c \in Z^{d \times n}$, solve

$$\max \{ c\underline{x} \; : \; x \in S^n \}$$

Matchings, Dipaths, Inequalities

Shifted Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $c \in Z^{d \times n}$, solve

$$\max \{ c\underline{x} \; : \; x \in S^n \}$$

Proposition: for perfect matchings the shifting problem is NP-hard even for n=2 with shifted c and cubic graphs

Matchings, Dipaths, Inequalities

Shifted Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $c \in Z^{d \times n}$, solve

$$\max \{ c\underline{x} \; : \; x \in S^n \}$$

Proposition: for perfect matchings the shifting problem is NP-hard even for n=2 with shifted c and cubic graphs

matrix c is shifted if $c = \underline{c}$, that is, its rows are nonincreasing

Matchings, Dipaths, Inequalities

Shifted Combinatorial Optimization: given $S \subseteq \{0,1\}^d$ and $c \in Z^{d \times n}$, solve

$$\max \{ c\underline{x} \; : \; x \in S^n \}$$

Proposition: for perfect matchings the shifting problem is NP-hard even for n=2 with shifted c and cubic graphs

Proof: Use shifting to find n=2 matchings with minimum shared edges.
If they are disjoint then the cubic graph is 3-edge-colorable.

(set $c_{i,1} = 0$, $c_{i,2} = -1$, $i=1, \ldots, d$)

Shmuel Onn

Matchings, Dipaths, Inequalities

Theorem 1: Shifting with shifted $c=\underline{c}$ is polynomial time solvable for any set $S=\{s\in\{0,1\}^d : As\leq b\}$ with A totally unimodular and b integer.

Matchings, Dipaths, Inequalities

Theorem 1: Shifting with shifted $c=\underline{c}$ is polynomial time solvable for any set $S=\{s\in\{0,1\}^d : As\le b\}$ with A totally unimodular and b integer.

Specifically, the Shifted and Lexicographic problems are solvable for:

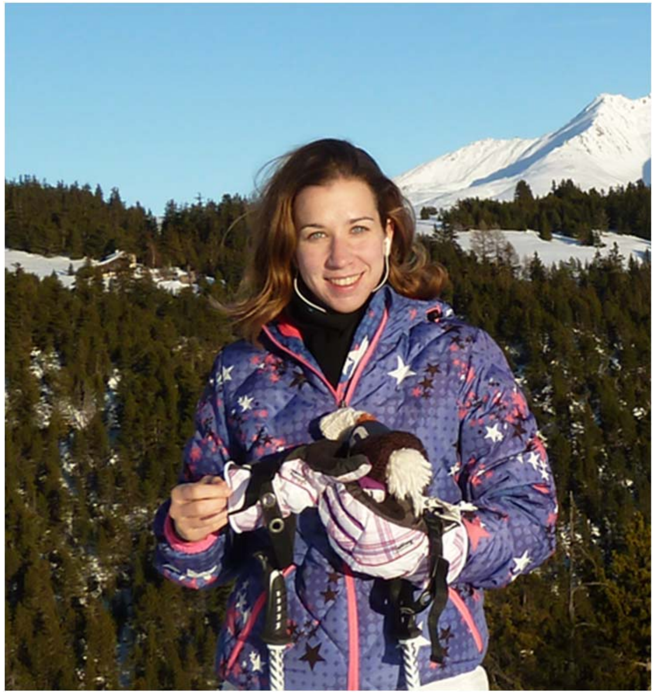
- perfect matchings in bipartite graph
- dipaths from u to v in directed graph

Matchings, Dipaths, Inequalities

Theorem 1: Shifting with shifted $c=\underline{c}$ is polynomial time solvable for any set $S=\{s\in\{0,1\}^d : As\leq b\}$ with A totally unimodular and b integer.

Specifically, the Shifted and Lexicographic problems are solvable for:

- perfect matchings in bipartite graph
- dipaths from u to v in directed graph



Reference: The unimodular intersection problem (Kaibel, Onn, Pauline Sarrabezolles), 2015

Outline of Proof of Theorem 1

The problem is: $\max\{c\underline{x} : x \in \{0,1\}^{d \times n},\ Ax^k \leq b,\ k=1,\ldots,n\}$

Outline of Proof of Theorem 1

The problem is: $\max\{c\underline{x} : x \in \{0,1\}^{d \times n},\ Ax^k \leq b,\ k=1,\ldots,n\}$

1. Obtain in polytime optimal y for the auxiliary linear integer program:

$$\max\{cy : y \in \{0,1\}^{d \times n},\ A\sum y^k \leq nb\}$$

Outline of Proof of Theorem 1

The problem is: $\max\{c\underline{x} : x \in \{0,1\}^{d \times n},\ Ax^k \leq b,\ k=1,\ldots,n\}$

1. Obtain in polytime optimal y for the auxiliary linear integer program:

$$\max\{cy : y \in \{0,1\}^{d \times n},\ A\sum y^k \leq nb\}$$

2. By decomposition lemma for A TU can obtain in polytime z satisfying:

$$z \in \{0,1\}^{d \times n},\ \sum z^k = \sum y^k,\ Az^k \leq b,\ k=1,\ldots,n$$

Outline of Proof of Theorem 1

The problem is: $\max\{c\underline{x} \;:\; x \in \{0,1\}^{d \times n},\; Ax^k \le b,\; k=1,\dots,n\}$

1. Obtain in polytime optimal y for the auxiliary linear integer program:

$$\max\{cy \;:\; y \in \{0,1\}^{d \times n},\; A\sum y^k \le nb\}$$

2. By decomposition lemma for A TU can obtain in polytime z satisfying:

$$z \in \{0,1\}^{d \times n},\; \sum z^k = \sum y^k,\; Az^k \le b,\; k=1,\dots,n$$

3. Any feasible x is no better than z in the original program, since

$$c\underline{z} = c\underline{y} \ge cy \ge c\underline{x}$$

since $\underline{z} = \underline{y}$, $c=\underline{c}$ is shifted, and $\underline{x}$ is feasible in the auxiliary program.

Matroids, Matroid Intersections

Theorem 2: Shifting with shifted $c=\underline{c}$ is polynomial time solvable for:

- Bases or independent sets in a matroid given by independence oracle
- Intersection of strongly orderable matroids (transversals, gammoids)

Matroids, Matroid Intersections

Theorem 2: Shifting with shifted $c=\underline{c}$ is polynomial time solvable for:

- Bases or independent sets in a matroid given by independence oracle
- Intersection of strongly orderable matroids (transversals, gammoids)

Reference: Shifted matroid optimization
(Levin, Onn), 2016

Matroids, Matroid Intersections

Theorem 2: Shifting with shifted $c=\underline{c}$ is polynomial time solvable for:

- Bases or independent sets in a matroid given by independence oracle
- Intersection of strongly orderable matroids (transversals, gammoids)

Proof: uses matroid union, matroid intersection, and other ingredients.

Reference: Shifted matroid optimization
(Levin, Onn), 2016

Sets Given Explicitly

Reference: Shifted combinatorial optimization
(Gajarský, Hlinený, Martin Koutecký, Onn), in preparation

Sets Given Explicitly

Proposition: shifting is NP-hard already for explicitly given sets $S = \{s^1, s^2, ..., s^m\} \subseteq \{0,1\}^d$ and shifted $c = \underline{c} \in \{0,1\}^{d \times n}$

Sets Given Explicitly

Proposition: shifting is NP-hard already for explicitly given sets $S = \{s^1, s^2, \ldots, s^m\} \subseteq \{0,1\}^d$ and shifted $c = \underline{c} \in \{0,1\}^{d \times n}$

Proof: Given graph with $d=3n$ vertices, let S be the set of indicators of triples of vertices inducing triangles.

Shift to find n triangles covering maximum vertices. If all are covered then they partition the vertex set.

(set $c_{i,1} = 1$, $c_{i,k} = 0$, $i=1, \ldots, d$, $k=2, \ldots, n$)

Sets Given Explicitly

Theorem 3: Consider shifting over $S = \{s^1, \ldots, s^m\} \subseteq R^d$ and $c \in R^{d \times n}$.

Sets Given Explicitly

Theorem 3: Consider shifting over $S = \{s^1, \ldots, s^m\} \subseteq R^d$ and $c \in R^{d \times n}$.

- it reduces to nonlinear integer programming over the simplex,

$$\max \{ f_c(n_1, \ldots, n_m) \ : \ n_1, \ldots, n_m \in Z_+ , \quad n_1 + \ldots + n_m = n \}$$

Sets Given Explicitly

Theorem 3: Consider shifting over $S = \{s^1, \ldots, s^m\} \subseteq R^d$ and $c \in R^{d \times n}$.

- it reduces to nonlinear integer programming over the simplex,

$$\max \{ f_c(n_1, \ldots, n_m) \; : \; n_1, \ldots, n_m \in Z_+ , \quad n_1 + \ldots + n_m = n \}$$

- when c is shifted f_c is concave and when $-c$ is shifted f_c is convex

Sets Given Explicitly

Theorem 3: Consider shifting over $S=\{s^1, \ldots, s^m\} \subseteq R^d$ and $c \in R^{d \times n}$.

- it reduces to nonlinear integer programming over the simplex,

$$\max \{ f_c(n_1, \ldots, n_m) \; : \; n_1, \ldots, n_m \in Z_+ , \quad n_1 + \ldots + n_m = n \}$$

- when c is shifted f_c is concave and when $-c$ is shifted f_c is convex

- with m fixed and c arbitrary it is polynomial time solvable

Sets Given Explicitly

Theorem 3: Consider shifting over $S = \{s^1, \ldots, s^m\} \subseteq R^d$ and $c \in R^{d \times n}$.

- it reduces to nonlinear integer programming over the simplex,

$$\max \{ f_c(n_1, \ldots, n_m) \ : \ n_1, \ldots, n_m \in Z_+ , \quad n_1 + \ldots + n_m = n \}$$

- when c is shifted f_c is concave and when $-c$ is shifted f_c is convex

- with m fixed and c arbitrary it is polynomial time solvable

- with m variable and $-c$ shifted it is polynomial time solvable

Sets Given Explicitly

Theorem 3: Consider shifting over $S = \{s^1, \ldots, s^m\} \subseteq R^d$ and $c \in R^{d \times n}$.

- it reduces to nonlinear integer programming over the simplex,

$$\max \{\ f_c(n_1, \ldots, n_m)\ :\ n_1, \ldots, n_m \in Z_+ ,\quad n_1 + \ldots + n_m = n\ \}$$

- when c is shifted f_c is concave and when $-c$ is shifted f_c is convex
- with m fixed and c arbitrary it is polynomial time solvable
- with m variable and $-c$ shifted it is polynomial time solvable
- with m parameter and c shifted it is fixed-parameter tractable

Sets Given Explicitly

Theorem 3: Consider shifting over $S = \{s^1, \ldots, s^m\} \subseteq R^d$ and $c \in R^{d \times n}$.

- it reduces to nonlinear integer programming over the simplex,

$$\max \{ f_c(n_1, \ldots, n_m) \ : \ n_1, \ldots, n_m \in Z_+ , \quad n_1 + \ldots + n_m = n \}$$

- when c is shifted f_c is concave and when $-c$ is shifted f_c is convex

- with m fixed and c arbitrary it is polynomial time solvable

- with m variable and $-c$ shifted it is polynomial time solvable

- with m parameter and c shifted it is fixed-parameter tractable

Proof: uses convex integer minimization, Oertel, Wagner, Weismantel, 2014

Shifted Linear Programming

Corollary: to solve the shifting problem over any S with nondecreasing c, solve the linear problem with $w := \sum c^k$, obtain $s \in S$, return $x := [s, \ldots, s]$

Proposition: shifted linear programming over a nonempty polytope $S = \{s \in R^d : As \leq b\}$ is NP-hard already for $n=2$ and shifted $c=\underline{c}$

Shmuel Onn

Beyond

Beyond

Some work in progress: (Gajarský, Hlinený, Koutecký, Onn)

- Shifting logic defined properties in graphs of bounded treewidth

Beyond

Some work in progress: (Gajarský, Hlinený, Koutecký, Onn)

- Shifting logic defined properties in graphs of bounded treewidth

Open directions: plenty !

Beyond

Some work in progress: (Gajarský, Hlinený, Koutecký, Onn)

- Shifting logic defined properties in graphs of bounded treewidth

Open directions: plenty !

- shift your favorite combinatorial optimization problem

Beyond

Some work in progress: (Gajarský, Hlinený, Koutecký, Onn)

- Shifting logic defined properties in graphs of bounded treewidth

Open directions: plenty !

- shift your favorite combinatorial optimization problem

- shifted integer programming (in particular defined by TU matrix)

Beyond

Some work in progress: (Gajarský, Hlinený, Koutecký, Onn)

- Shifting logic defined properties in graphs of bounded treewidth

Open directions: plenty !

- shift your favorite combinatorial optimization problem
- shifted integer programming (in particular defined by TU matrix)
- parametrized shifting by various parameters (in particular n)

Beyond

Some work in progress: (Gajarský, Hlinený, Koutecký, Onn)

- Shifting logic defined properties in graphs of bounded treewidth

Open directions: plenty !

- shift your favorite combinatorial optimization problem
- shifted integer programming (in particular defined by TU matrix)
- parametrized shifting by various parameters (in particular n)
- approximated shifting

Beyond

Some work in progress: (Gajarský, Hlinený, Koutecký, Onn)

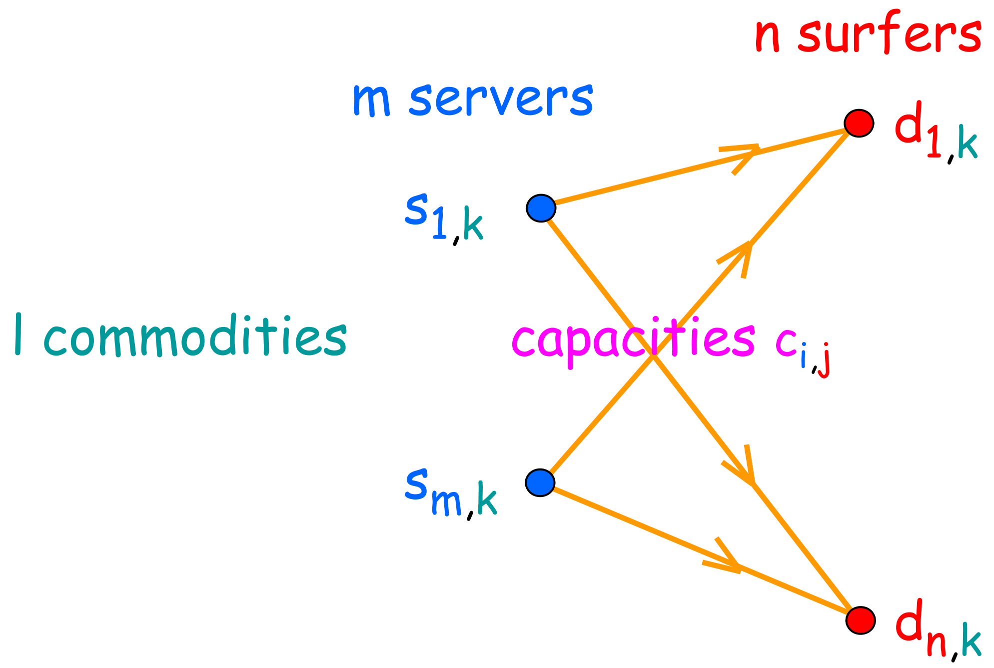
- Shifting logic defined properties in graphs of bounded treewidth

Open directions: plenty !

- shift your favorite combinatorial optimization problem
- shifted integer programming (in particular defined by TU matrix)
- parametrized shifting by various parameters (in particular n)
- approximated shifting
- shift your imagination !

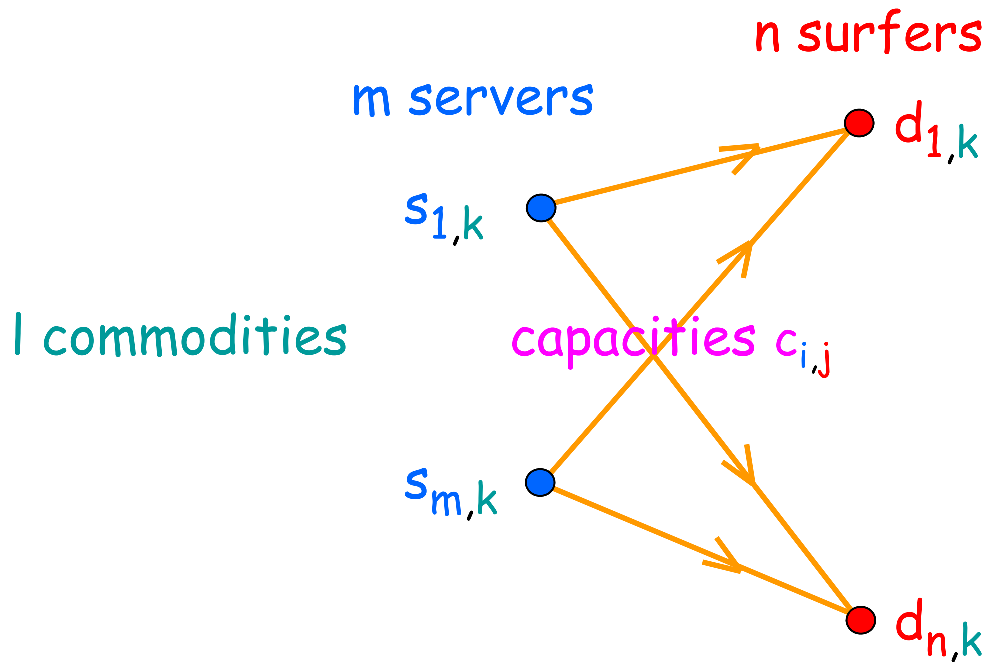
Multicommodity Flows

Find flow of l commodities from m servers to n surfers satisfying given supplies $s_{i,k}$, demands $d_{j,k}$ and capacities $c_{i,j}$ of total bit size L



Multicommodity Flows

Find flow of l commodities from m servers to n surfers satisfying given supplies $s_{i,k}$, demands $d_{j,k}$ and capacities $c_{i,j}$ of total bit size L



With $l=2$ or $m=3$ it is NP-complete so assume both l,m are parameters

Shmuel Onn

Multicommodity Flows

Find flow of l commodities from m servers to n surfers satisfying given supplies $s_{i,k}$, demands $d_{j,k}$ and capacities $c_{i,j}$ of total bit size L

2008: polynomial time $O(n^{g(l,m)} L)$ with Graver complexity $g(l,m)$ exponential in l,m (De Loera, Hemmecke, Onn, Weismantel) (theory of n-fold IP)

Multicommodity Flows

Find flow of l commodities from m servers to n surfers satisfying given supplies $s_{i,k}$, demands $d_{j,k}$ and capacities $c_{i,j}$ of total bit size L

2008: polynomial time $O(n^{g(l,m)} L)$ with Graver complexity $g(l,m)$ exponential in l,m (De Loera, Hemmecke, Onn, Weismantel) (theory of n-fold IP)

2013: fixed-parameter tractable $O(n^3 L)$ (Hemmecke, Onn, Lyubov Romanchuk)

Multicommodity Flows

Find flow of l commodities from m servers to n surfers satisfying given supplies $s_{i,k}$, demands $d_{j,k}$ and capacities $c_{i,j}$ of total bit size L

2008: polynomial time $O(n^{g(l,m)} L)$ with Graver complexity $g(l,m)$ exponential in l,m
(De Loera, Hemmecke, Onn, Weismantel) (theory of n-fold IP)

2013: fixed-parameter tractable $O(n^3 L)$
(Hemmecke, Onn, Lyubov Romanchuk)

2015: strongly polynomial $O(n^{g(l,m)})$
(De Loera, Hemmecke, Lee)

Multicommodity Flows

Find flow of l commodities from m servers to n surfers satisfying given supplies $s_{i,k}$, demands $d_{j,k}$ and capacities $c_{i,j}$ of total bit size L

2008: polynomial time $O(n^{g(l,m)} L)$ with Graver complexity $g(l,m)$ exponential in l,m
(De Loera, Hemmecke, Onn, Weismantel) (theory of n-fold IP)

2013: fixed-parameter tractable $O(n^3 L)$
(Hemmecke, Onn, Lyubov Romanchuk)
usually the fastest

2015: strongly polynomial $O(n^{g(l,m)})$
(De Loera, Hemmecke, Lee)

Multicommodity Flows

Find flow of l commodities from m servers to n surfers satisfying given supplies $s_{i,k}$, demands $d_{j,k}$ and capacities $c_{i,j}$ of total bit size L

2008: polynomial time $O(n^{g(l,m)} L)$ with Graver complexity $g(l,m)$ exponential in l,m
(De Loera, Hemmecke, Onn, Weismantel) (theory of n-fold IP)

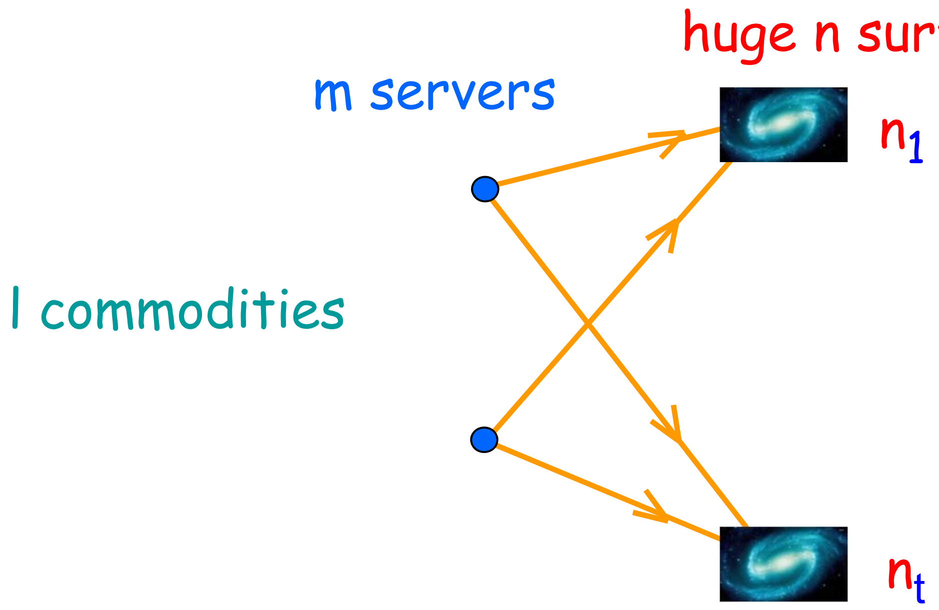
2013: fixed-parameter tractable $O(n^3 L)$
(Hemmecke, Onn, Lyubov Romanchuk)
usually the fastest

2015: strongly polynomial $O(n^{g(l,m)})$
(De Loera, Hemmecke, Lee)

Open: algorithm that is both fixed-parameter tractable and strongly polynomial ?

Multicommodity Flows

Huge version: surfers come in huge clouds of t types



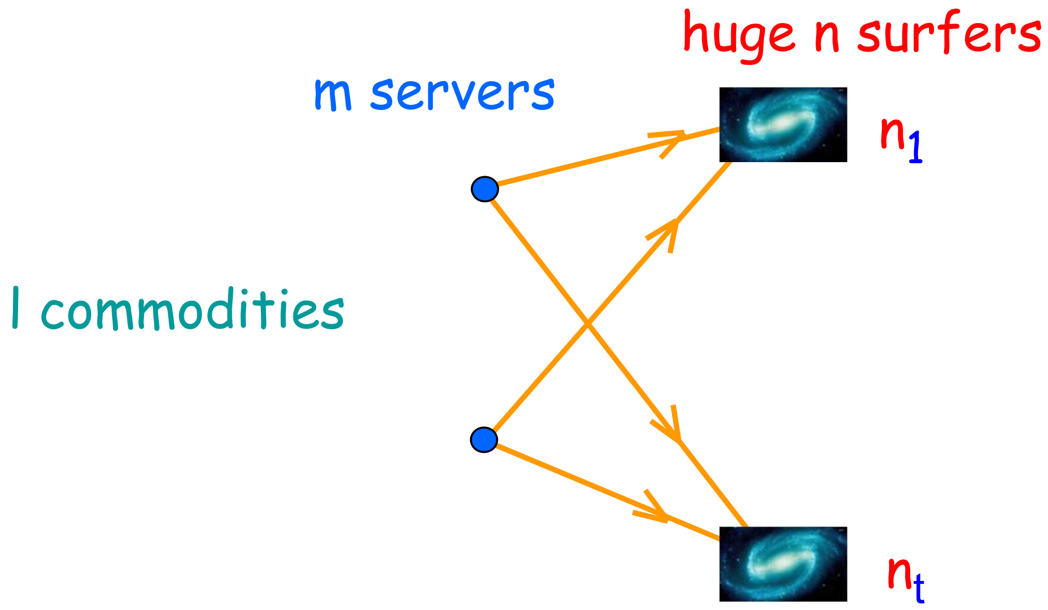
t surfer types

$n_1 + \ldots + n_t = n$

binary encoded

Multicommodity Flows

Huge version: surfers come in huge clouds of t types



t surfer types

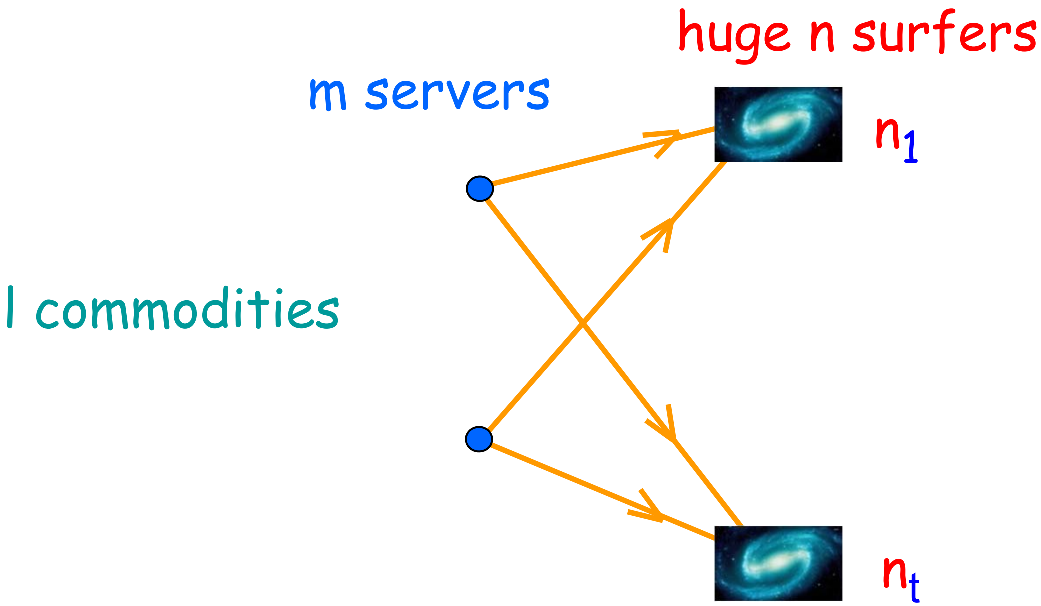
$n_1 + \ldots + n_t = n$

binary encoded

2016 (Onn): fixed-parameter tractable with parameters l,t, variable m, huge n

Multicommodity Flows

Huge version: surfers come in huge clouds of t types



t surfer types

$n_1 + \ldots + n_t = n$

binary encoded

2016 (Onn): fixed-parameter tractable with parameters l,t, variable m, huge n

Open: 4-dimensional huge tables are only known to be in NP intersect coNP