Generation of Monotone Graph Structures

Endre Boros*

MSIS Department and RUTCOR, Rutgers University

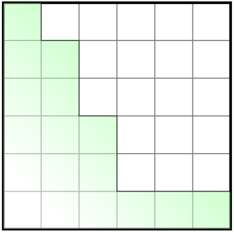
AGTAC, Koper, June 16-19, 2015

*Based on joint results with K. Elbassioni, V. Gurvich, L. Khachiyan (1952-2005), and K. Makino

In Memory of Leo Khachiyan (1952-2005)

Outline

Monotone generation

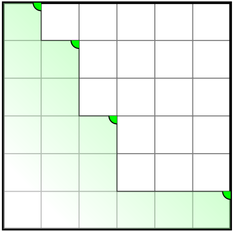


Consider a monotone property Π in a lattice represented by a membership oracle

- $\mathbf{Max}(\Pi) = \{$ max'l elements $\mathbf{v} \in \Pi\}$.
- $\mathbf{Min}(\overline{\Pi}) = \{$ min'l elements $\mathbf{v} \notin \Pi\}$.

Given Π, generate

Monotone generation



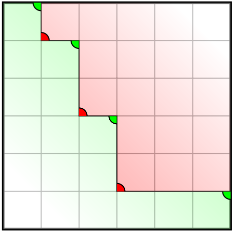
Consider a monotone property Π in a lattice represented by a membership oracle

- $\mathbf{Max}(\Pi) = \{$ max'l elements $\mathbf{v} \in \Pi\}$.
- $\mathbf{Min}(\overline{\Pi}) = \{$ min'l elements $\mathbf{v} \notin \Pi\}$.

Given Π, generate

- $\mathbf{Max}(\Pi)$ (or $\mathbf{Min}(\overline{\Pi})$ or both).

Monotone generation



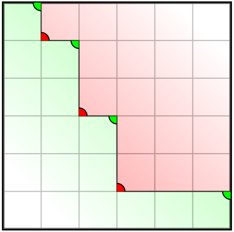
Consider a monotone property Π in a lattice represented by a membership oracle

- $\mathbf{Max}(\Pi) \;=\; \{$ max'l elements $\mathbf{v} \in \Pi\}$.
- $\mathbf{Min}(\overline{\Pi}) \;=\; \{$ min'l elements $\mathbf{v} \notin \Pi\}$.

Given Π, generate

- $\mathbf{Max}(\Pi)$ (or $\mathbf{Min}(\overline{\Pi})$ or both).
- Typically $\mathbf{size}(\Pi) \ll |\mathbf{Max}(\Pi)|$.
- How to measure efficiency of generation?

Monotone generation



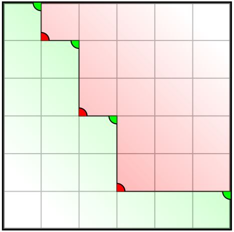
Consider a monotone property $\mathbf{\Pi}$ in a lattice represented by a membership oracle

- $\mathbf{Max}(\mathbf{\Pi}) = \{$ max'l elements $\mathbf{v} \in \mathbf{\Pi}\}$.
- $\mathbf{Min}(\overline{\mathbf{\Pi}}) = \{$ min'l elements $\mathbf{v} \notin \mathbf{\Pi}\}$.

Given $\mathbf{\Pi}$, generate

- $\mathbf{Max}(\mathbf{\Pi})$ (or $\mathbf{Min}(\overline{\mathbf{\Pi}})$ or both).
- Typically $\mathbf{size}(\mathbf{\Pi}) \ll |\mathbf{Max}(\mathbf{\Pi})|$.
- How to measure efficiency of generation?

Monotone generation



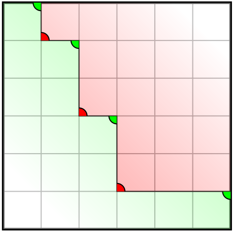
Consider a monotone property Π in a lattice represented by a membership oracle

- $\mathbf{Max}(\Pi) = \{$ max'l elements $\mathbf{v} \in \Pi\}$.
- $\mathbf{Min}(\overline{\Pi}) = \{$ min'l elements $\mathbf{v} \notin \Pi\}$.

Given Π, generate

- $\mathbf{Max}(\Pi)$ (or $\mathbf{Min}(\overline{\Pi})$ or both).
- Typically $\mathbf{size}(\Pi) \ll |\mathbf{Max}(\Pi)|$.
- How to measure efficiency of generation?

Monotone generation



Consider a monotone property Π in a lattice represented by a membership oracle

- $\mathbf{Max}(\Pi) = \{$ max'l elements $\mathbf{v} \in \Pi\}$.
- $\mathbf{Min}(\overline{\Pi}) = \{$ min'l elements $\mathbf{v} \notin \Pi\}$.

Given Π, generate

- $\mathbf{Max}(\Pi)$ (or $\mathbf{Min}(\overline{\Pi})$ or both).
- Typically $\mathbf{size}(\Pi) \ll |\mathbf{Max}(\Pi)|$.
- How to measure efficiency of generation?

Outline

Complexity of generation

Sequential generation

- Given a monotone system $\mathbf{\Pi}$ of input size $|\mathbf{\Pi}| = \mathbf{N}$, an algorithm $\mathfrak{A}$ generates one-by-one the elements

$$\mathbf{Max}(\mathbf{\Pi}) = \{\mathbf{v_0}, \mathbf{v_1}, \ldots, \mathbf{v_{M-1}}\},$$

outputting $\mathbf{v_k}$ at time $\mathbf{t_k}$ $\quad (\mathbf{t_0} \leq \mathbf{t_1} \leq \cdots \leq \mathbf{t_M})$.

- Algorithm $\mathfrak{A}$ is said to work
  - in total polynomial time, if $\mathbf{t_M} \leq \mathbf{poly(N, M)}$
  - in incremental polynomial time, if

$$\mathbf{t_k} \leq \mathbf{poly(N, k)} \quad \text{for all} \quad \mathbf{k} \leq \mathbf{M}$$

  - with polynomial delay, if

$$\mathbf{t_{k+1}} - \mathbf{t_k} \leq \mathbf{poly(N)} \quad \text{for all} \quad \mathbf{k} < \mathbf{M}$$

Complexity of generation

Sequential generation

- Given a monotone system $\mathbf{\Pi}$ of input size $|\mathbf{\Pi}| = \mathbf{N}$, an algorithm $\mathfrak{A}$ generates one-by-one the elements

$$\mathbf{Max}(\mathbf{\Pi}) = \{\mathbf{v_0}, \mathbf{v_1}, ..., \mathbf{v_{M-1}}\},$$

outputting $\mathbf{v_k}$ at time $\mathbf{t_k}$ $\quad (\mathbf{t_0} \leq \mathbf{t_1} \leq \cdots \leq \mathbf{t_M})$.

- Algorithm $\mathfrak{A}$ is said to work
  - in **total polynomial time**, if $\mathbf{t_M} \leq \mathbf{poly}(\mathbf{N}, \mathbf{M})$
  - in **incremental polynomial time**, if

$$\mathbf{t_k} \leq \mathbf{poly}(\mathbf{N}, \mathbf{k}) \quad \text{for all} \quad \mathbf{k} \leq \mathbf{M}$$

  - with **polynomial delay**, if

$$\mathbf{t_{k+1}} - \mathbf{t_k} \leq \mathbf{poly}(\mathbf{N}) \quad \text{for all} \quad \mathbf{k} < \mathbf{M}$$

Complexity of generation

Sequential generation

- Given a monotone system $\mathbf{\Pi}$ of input size $|\mathbf{\Pi}| = \mathbf{N}$, an algorithm $\mathfrak{A}$ generates one-by-one the elements

$$\mathbf{Max}(\mathbf{\Pi}) = \{\mathbf{v_0}, \mathbf{v_1}, ..., \mathbf{v_{M-1}}\},$$

  outputting $\mathbf{v_k}$ at time $\mathbf{t_k}$ $\quad(\mathbf{t_0} \leq \mathbf{t_1} \leq \cdots \leq \mathbf{t_M})$.
- Algorithm $\mathfrak{A}$ is said to work
  - in **total polynomial time**, if $\mathbf{t_M} \leq \mathbf{poly}(\mathbf{N}, \mathbf{M})$
  - in **incremental polynomial time**, if

$$\mathbf{t_k} \leq \mathbf{poly}(\mathbf{N}, \mathbf{k}) \quad \text{for all} \quad \mathbf{k} \leq \mathbf{M}$$

  - with **polynomial delay**, if

$$\mathbf{t_{k+1}} - \mathbf{t_k} \leq \mathbf{poly}(\mathbf{N}) \quad \text{for all} \quad \mathbf{k} < \mathbf{M}$$

Complexity of generation

Sequential generation

- Given a monotone system $\mathbf{\Pi}$ of input size $|\mathbf{\Pi}| = \mathbf{N}$, an algorithm $\mathfrak{A}$ generates one-by-one the elements

$$\mathbf{Max}(\mathbf{\Pi}) = \{\mathbf{v_0}, \mathbf{v_1}, ..., \mathbf{v_{M-1}}\},$$

  outputting $\mathbf{v_k}$ at time $\mathbf{t_k}$ $\quad (\mathbf{t_0} \leq \mathbf{t_1} \leq \cdots \leq \mathbf{t_M})$.
- Algorithm $\mathfrak{A}$ is said to work
  - in **total polynomial time**, if $\mathbf{t_M} \leq \mathbf{poly}(\mathbf{N}, \mathbf{M})$
  - in **incremental polynomial time**, if

$$\mathbf{t_k} \leq \mathbf{poly}(\mathbf{N}, \mathbf{k}) \quad \text{for all} \quad \mathbf{k} \leq \mathbf{M}$$

  - with **polynomial delay**, if

$$\mathbf{t_{k+1}} - \mathbf{t_k} \leq \mathbf{poly}(\mathbf{N}) \quad \text{for all} \quad \mathbf{k} < \mathbf{M}$$

Outline

Hardness of generation

NEXT($\mathbf{\Pi}, \mathcal{M}$)

Given a monotone system $\mathbf{\Pi}$ and $\mathcal{M} \subseteq \mathbf{Max}(\mathbf{\Pi})$, decide if $\mathcal{M} = \mathbf{Max}(\mathbf{\Pi})$, and if not, find $\mathbf{v} \in \mathbf{Max}(\mathbf{\Pi}) \setminus \mathcal{M}$.

Theorem (Ms. Folklore, Bronze Age)

$\mathbf{Max}(\mathbf{\Pi})$ *can be generated in incremental polynomial time if and only if problem NEXT*($\mathbf{\Pi}, \mathcal{M}$) *can be solved in polynomial time for all* $\mathcal{M} \subseteq \mathbf{Max}(\mathbf{\Pi})$.

... (Lawler, Lenstra, and Rinnooy Kann, 1980) ...

Generation is hard if NEXT($\mathbf{\Pi}, \mathcal{M}$) is NP-hard.

Hardness of generation

NEXT($\mathbf{\Pi}, \mathcal{M}$)

Given a monotone system $\mathbf{\Pi}$ and $\mathcal{M} \subseteq \mathbf{Max}(\mathbf{\Pi})$, decide if $\mathcal{M} = \mathbf{Max}(\mathbf{\Pi})$, and if not, find $\mathbf{v} \in \mathbf{Max}(\mathbf{\Pi}) \setminus \mathcal{M}$.

Theorem (Ms. Folklore, Bronze Age)

$\mathbf{Max}(\mathbf{\Pi})$ *can be generated in incremental polynomial time if and only if problem NEXT($\mathbf{\Pi}, \mathcal{M}$) can be solved in polynomial time for all* $\mathcal{M} \subseteq \mathbf{Max}(\mathbf{\Pi})$.

... (Lawler, Lenstra, and Rinnooy Kann, 1980) ...

Generation is hard if NEXT($\mathbf{\Pi}, \mathcal{M}$) is NP-hard.

Hardness of generation

NEXT($\mathbf{\Pi}, \mathcal{M}$)

Given a monotone system $\mathbf{\Pi}$ and $\mathcal{M} \subseteq \mathbf{Max}(\mathbf{\Pi})$, decide if $\mathcal{M} = \mathbf{Max}(\mathbf{\Pi})$, and if not, find $\mathbf{v} \in \mathbf{Max}(\mathbf{\Pi}) \setminus \mathcal{M}$.

Theorem (Ms. Folklore, Bronze Age)

$\mathbf{Max}(\mathbf{\Pi})$ *can be generated in incremental polynomial time if and only if problem NEXT*($\mathbf{\Pi}, \mathcal{M}$) *can be solved in polynomial time for all* $\mathcal{M} \subseteq \mathbf{Max}(\mathbf{\Pi})$.

... (Lawler, Lenstra, and Rinnooy Kann, 1980) ...

Generation is hard if NEXT($\mathbf{\Pi}, \mathcal{M}$) is NP-hard.

Hardness of generation

NEXT($\mathbf{\Pi}, \mathcal{M}$)

Given a monotone system $\mathbf{\Pi}$ and $\mathcal{M} \subseteq \mathbf{Max}(\mathbf{\Pi})$, decide if $\mathcal{M} = \mathbf{Max}(\mathbf{\Pi})$, and if not, find $\mathbf{v} \in \mathbf{Max}(\mathbf{\Pi}) \setminus \mathcal{M}$.

Theorem (Ms. Folklore, Bronze Age)

$\mathbf{Max}(\mathbf{\Pi})$ *can be generated in incremental polynomial time if and only if problem NEXT*($\mathbf{\Pi}, \mathcal{M}$) *can be solved in polynomial time for all* $\mathcal{M} \subseteq \mathbf{Max}(\mathbf{\Pi})$.

... (Lawler, Lenstra, and Rinnooy Kann, 1980) ...

Generation is hard if NEXT($\mathbf{\Pi}, \mathcal{M}$) is NP-hard.

Outline

Prime example for monotone generation

Hypergraph transversals

Let $|U| = m$ and $\mathcal{H} \subseteq 2^U$ be a hypergraph. Associate to it a property $\mathbf{\Pi} = \mathbf{\Pi}_{\mathcal{H}} \subseteq 2^U$ by

$$S \in \mathbf{\Pi} \quad \Leftrightarrow \quad S \textit{ is independent} \quad \Leftrightarrow \quad H \not\subseteq S \quad \forall H \in \mathcal{H}$$

- $\mathcal{H}^* = Max(\mathbf{\Pi}_{\mathcal{H}})$ is the family of maximal independent sets of $\mathcal{H}$.
- $\mathcal{H}^d = \{U \setminus S \mid S \in Max(\mathbf{\Pi}_{\mathcal{H}})\}$ is the family of minimal transversals of $\mathcal{H}$.
- $\mathcal{H} \to \mathcal{H}^d$ (or $\mathcal{H} \to \mathcal{H}^*$) are known as the hypergraph transversal or monotone dualization problems.

Prime example for monotone generation

Hypergraph transversals

Let $|U| = m$ and $\mathcal{H} \subseteq 2^U$ be a hypergraph. Associate to it a property $\mathbf{\Pi} = \mathbf{\Pi}_{\mathcal{H}} \subseteq 2^U$ by

$$S \in \mathbf{\Pi} \iff S \textit{ is independent} \iff H \not\subseteq S \quad \forall H \in \mathcal{H}$$

$$\iff \overline{S} \textit{ is a transversal} \iff \overline{S} \cap H \neq \emptyset \quad \forall H \in \mathcal{H}$$

- $\mathcal{H}^* = Max(\mathbf{\Pi}_{\mathcal{H}})$ is the family of maximal independent sets of $\mathcal{H}$.
- $\mathcal{H}^d = \{U \setminus S \mid S \in Max(\mathbf{\Pi}_{\mathcal{H}})\}$ is the family of minimal transversals of $\mathcal{H}$.
- $\mathcal{H} \to \mathcal{H}^d$ (or $\mathcal{H} \to \mathcal{H}^*$) are known as the hypergraph transversal or monotone dualization problems.

Prime example for monotone generation

Hypergraph transversals

Let $|U| = m$ and $\mathcal{H} \subseteq 2^U$ be a hypergraph. Associate to it a property $\mathbf{\Pi} = \mathbf{\Pi}_{\mathcal{H}} \subseteq 2^U$ by

$$\begin{aligned} S \in \mathbf{\Pi} \quad &\Leftrightarrow \quad S \textit{ is independent} &&\Leftrightarrow \quad H \not\subseteq S &&\forall H \in \mathcal{H} \\ &\Leftrightarrow \quad \overline{S} \textit{ is a transversal} &&\Leftrightarrow \quad \overline{S} \cap H \neq \emptyset &&\forall H \in \mathcal{H} \end{aligned}$$

- $\mathcal{H}^* = Max(\mathbf{\Pi}_{\mathcal{H}})$ is the family of maximal independent sets of $\mathcal{H}$.
- $\mathcal{H}^d = \{U \setminus S \mid S \in Max(\mathbf{\Pi}_{\mathcal{H}})\}$ is the family of minimal transversals of $\mathcal{H}$.
- $\mathcal{H} \to \mathcal{H}^d$ (or $\mathcal{H} \to \mathcal{H}^*$) are known as the hypergraph transversal or monotone dualization problems.

Prime example for monotone generation

Hypergraph transversals

Let $|U| = m$ and $\mathcal{H} \subseteq 2^U$ be a hypergraph. Associate to it a property $\mathbf{\Pi} = \mathbf{\Pi}_{\mathcal{H}} \subseteq 2^U$ by

$$\begin{aligned} S \in \mathbf{\Pi} \quad &\Leftrightarrow \quad S \textit{ is independent} \quad &\Leftrightarrow \quad H \not\subseteq S \quad \forall H \in \mathcal{H} \\ &\Leftrightarrow \quad \overline{S} \textit{ is a transversal} \quad &\Leftrightarrow \quad \overline{S} \cap H \neq \emptyset \quad \forall H \in \mathcal{H} \end{aligned}$$

- $\mathcal{H}^* = Max(\mathbf{\Pi}_{\mathcal{H}})$ is the family of maximal independent sets of $\mathcal{H}$.
- $\mathcal{H}^d = \{U \setminus S \mid S \in Max(\mathbf{\Pi}_{\mathcal{H}})\}$ is the family of minimal transversals of $\mathcal{H}$.
- $\mathcal{H} \rightarrow \mathcal{H}^d$ (or $\mathcal{H} \rightarrow \mathcal{H}^*$) are known as the hypergraph transversal or monotone dualization problems.

Prime example for monotone generation

Hypergraph transversals

Let $|U| = m$ and $\mathcal{H} \subseteq 2^U$ be a hypergraph. Associate to it a property $\mathbf{\Pi} = \mathbf{\Pi}_{\mathcal{H}} \subseteq 2^U$ by

$$
\begin{aligned}
S \in \mathbf{\Pi} \quad &\Leftrightarrow \quad S \textit{ is independent} \quad &&\Leftrightarrow \quad H \not\subseteq S \quad &&\forall H \in \mathcal{H} \\
&\Leftrightarrow \quad \overline{S} \textit{ is a transversal} \quad &&\Leftrightarrow \quad \overline{S} \cap H \neq \emptyset \quad &&\forall H \in \mathcal{H}
\end{aligned}
$$

- $\mathcal{H}^* = Max(\mathbf{\Pi}_{\mathcal{H}})$ is the family of maximal independent sets of $\mathcal{H}$.
- $\mathcal{H}^d = \{U \setminus S \mid S \in Max(\mathbf{\Pi}_{\mathcal{H}})\}$ is the family of minimal transversals of $\mathcal{H}$.
- $\mathcal{H} \to \mathcal{H}^d$ (or $\mathcal{H} \to \mathcal{H}^*$) are known as the hypergraph transversal or monotone dualization problems.

Generating hypergraph transversals

Theorem (Fredmand and Khachiyan, 1996)

For any hypergraph $\mathcal{H}$ and an arbitrary family $\mathcal{M} \subseteq \mathcal{H}^d$ of its minimal transversals, problem NEXT($\mathcal{H}, \mathcal{M}$) can be solved in $O\left((|\mathcal{H}| + |\mathcal{H}^d|)^{o(\log |\mathcal{H}| + |\mathcal{H}^d|)}\right)$ time.

Claim (Eiter and Gottlob, 1995)

If for all hyperedges $H \in \mathcal{H}$ we have $|H| \leq k$, where k is fixed, then $\mathcal{H}^d$ can be generated in incremental polynomial time.

Claim (Boros, Elbassioni, Gurvich, and Khachiyan, 2004)

If any ℓ hyperedges of $\mathcal{H}$ intersect in at most k points, where k, ℓ are fixed, then $\mathcal{H}^d$ can be generated in incremental polynomial time.

Generating hypergraph transversals

Theorem (Fredmand and Khachiyan, 1996)

For any hypergraph $\mathcal{H}$ and an arbitrary family $\mathcal{M} \subseteq \mathcal{H}^d$ of its minimal transversals, problem NEXT($\mathcal{H}, \mathcal{M}$) can be solved in $O\left((|\mathcal{H}| + |\mathcal{H}^d|)^{o(\log |\mathcal{H}| + |\mathcal{H}^d|)}\right)$ time.

Claim (Eiter and Gottlob, 1995)

If for all hyperedges $H \in \mathcal{H}$ we have $|H| \leq k$, where k is fixed, then $\mathcal{H}^d$ can be generated in incremental polynomial time.

Claim (Boros, Elbassioni, Gurvich, and Khachiyan, 2004)

If any ℓ hyperedges of $\mathcal{H}$ intersect in at most k points, where k, ℓ are fixed, then $\mathcal{H}^d$ can be generated in incremental polynomial time.

Generating hypergraph transversals

Theorem (Fredmand and Khachiyan, 1996)

For any hypergraph $\mathcal{H}$ and an arbitrary family $\mathcal{M} \subseteq \mathcal{H}^d$ of its minimal transversals, problem NEXT$(\mathcal{H}, \mathcal{M})$ can be solved in $O\left((|\mathcal{H}| + |\mathcal{H}^d|)^{o(\log|\mathcal{H}|+|\mathcal{H}^d|)}\right)$ time.

Claim (Eiter and Gottlob, 1995)

If for all hyperedges $H \in \mathcal{H}$ we have $|H| \leq k$, where k is fixed, then $\mathcal{H}^d$ can be generated in incremental polynomial time.

Claim (Boros, Elbassioni, Gurvich, and Khachiyan, 2004)

If any ℓ hyperedges of $\mathcal{H}$ intersect in at most k points, where k, ℓ are fixed, then $\mathcal{H}^d$ can be generated in incremental polynomial time.

Outline

Typical Monotone Systems

For a graph $G = (V, E)$, $b \in \mathbb{Z}_+^V$, $B \subseteq V \times V$, $U \subseteq V$

- Find all **maximal** subsets $F \subseteq E$ such that $d_F(v) \leq b(v)$ for all $v \in V$.
- Find all **minimal** subsets $F \subseteq E$ such that s and t are connected in (V, F) for all $(s, t) \in B$.
- Find all **maximal** subsets $F \subseteq E$ such that s and t are not connected for all $(s, t) \in B$.
- Find all **minimal** subsets $F \subseteq E$ such that U is within one connected component of (V, F).
- Find all **maximal** bipartite subgraphs of G.

Typical Monotone Systems

For a graph $G = (V, E)$, $b \in \mathbb{Z}_+^V$, $B \subseteq V \times V$, $U \subseteq V$

- Find all **maximal** subsets $F \subseteq E$ such that $d_F(v) \leq b(v)$ for all $v \in V$.
- Find all **minimal** subsets $F \subseteq E$ such that s and t are connected in (V, F) for all $(s, t) \in B$.
- Find all **maximal** subsets $F \subseteq E$ such that s and t are not connected for all $(s, t) \in B$.
- Find all **minimal** subsets $F \subseteq E$ such that U is within one connected component of (V, F).
- Find all **maximal** bipartite subgraphs of G.

Typical Monotone Systems

For a graph $G = (V, E)$, $b \in \mathbb{Z}_+^V$, $B \subseteq V \times V$, $U \subseteq V$

- Find all **maximal** subsets $F \subseteq E$ such that $d_F(v) \leq b(v)$ for all $v \in V$.
- Find all **minimal** subsets $F \subseteq E$ such that s and t are connected in (V, F) for all $(s, t) \in B$.
- Find all **maximal** subsets $F \subseteq E$ such that s and t are not connected for all $(s, t) \in B$.
- Find all **minimal** subsets $F \subseteq E$ such that U is within one connected component of (V, F).
- Find all **maximal** bipartite subgraphs of G.

Typical Monotone Systems

For a graph $G = (V, E)$, $b \in \mathbb{Z}_+^V$, $B \subseteq V \times V$, $U \subseteq V$

- Find all **maximal** subsets $F \subseteq E$ such that $d_F(v) \leq b(v)$ for all $v \in V$.
- Find all **minimal** subsets $F \subseteq E$ such that s and t are connected in (V, F) for all $(s, t) \in B$.
- Find all **maximal** subsets $F \subseteq E$ such that s and t are not connected for all $(s, t) \in B$.
- Find all **minimal** subsets $F \subseteq E$ such that U is within one connected component of (V, F).
- Find all **maximal** bipartite subgraphs of G.

Typical Monotone Systems

For a graph $G = (V, E)$, $b \in \mathbb{Z}_+^V$, $B \subseteq V \times V$, $U \subseteq V$

- Find all **maximal** subsets $F \subseteq E$ such that $d_F(v) \leq b(v)$ for all $v \in V$.
- Find all **minimal** subsets $F \subseteq E$ such that s and t are connected in (V, F) for all $(s, t) \in B$.
- Find all **maximal** subsets $F \subseteq E$ such that s and t are not connected for all $(s, t) \in B$.
- Find all **minimal** subsets $F \subseteq E$ such that U is within one connected component of (V, F).
- Find all **maximal** bipartite subgraphs of G.

Typical Monotone Systems

For a directed graph $D = (V, A)$, $w \in \mathbb{R}^A$

- Find all **minimal** subsets $F \subseteq A$ such that (V, F) is strongly connected.
- Find all **maximal** subsets $F \subseteq A$ such that (V, F) is acyclic.
- Find all subsets $C \subseteq A$ such that C is a simple directed cycle and $w(C) < 0$.

Typical Monotone Systems

For a directed graph $D = (V, A)$, $w \in \mathbb{R}^A$

- Find all **minimal** subsets $F \subseteq A$ such that (V, F) is strongly connected.
- Find all **maximal** subsets $F \subseteq A$ such that (V, F) is acyclic.
- Find all subsets $C \subseteq A$ such that C is a simple directed cycle and $w(C) < 0$.

Typical Monotone Systems

For a directed graph $D = (V, A)$, $w \in \mathbb{R}^A$

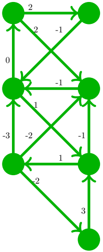
- Find all **minimal** subsets $F \subseteq A$ such that (V, F) is strongly connected.
- Find all **maximal** subsets $F \subseteq A$ such that (V, F) is acyclic.
- Find all subsets $C \subseteq A$ such that C is a simple directed cycle and $w(C) < 0$.

Typical Monotone Systems

For a directed graph $D = (V, A)$, $w \in \mathbb{R}^A$

- Find all **minimal** subsets $F \subseteq A$ such that (V, F) is strongly connected.
- Find all **maximal** subsets $F \subseteq A$ such that (V, F) is acyclic.
- Find all subsets $C \subseteq A$ such that C is a simple directed cycle and $w(C) < 0$. **Whoops! NOT MONOTONE!**

Negative cycle free subgraphs' polyhedron



Let $G = (V, E)$ be a directed graph, $w : E \to \mathbb{R}$, $x \in \mathbb{R}^V$, and consider the system of linear inequalities

$$\{x_i - x_j \leq w_{ij} \quad \forall\ (i,j) \in E\}$$

Min'l Infeasible Subsystems $\rightsquigarrow$ $\{\mathbf{C} \subseteq E \mid \mathbf{C}$ is a negative cycle $\}$

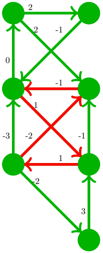
Theorem (Boros, Borys, Elbassioni, Gurvich and Khachiyan, 2005)

*Given a directed graph G with real weights on its arcs, generating all negative cycles of G is **NP-hard**.*

Corollary

Generating vertices of polyhedra is hard.

Negative cycle free subgraphs' polyhedron



Let $G = (V, E)$ be a directed graph, $w : E \to \mathbb{R}$, $x \in \mathbb{R}^V$, and consider the system of linear inequalities

$$\{x_i - x_j \leq w_{ij} \quad \forall\ (i,j) \in E\}$$

Min'l Infeasible Subsystems $\leftrightsquigarrow$ $\{\mathbf{C} \subseteq E \mid \mathbf{C}$ is a negative cycle $\}$

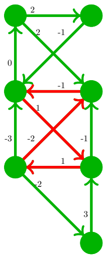
Theorem (Boros, Borys, Elbassioni, Gurvich and Khachiyan, 2005)

Given a directed graph G with real weights on its arcs, generating all negative cycles of G is ***NP-hard****.* Even if $w_{ij} \in \{\pm 1\}$ for all arcs $(i,j) \in E$.

Corollary

Generating vertices of polyhedra is hard.

Negative cycle free subgraphs' polyhedron



Let $G = (V, E)$ be a directed graph, $w : E \to \mathbb{R}$, $x \in \mathbb{R}^V$, and consider the system of linear inequalities

$$\{x_i - x_j \leq w_{ij} \quad \forall\ (i,j) \in E\}$$

Min'l Infeasible Subsystems $\leftrightsquigarrow$ $\{\mathbf{C} \subseteq E \mid \mathbf{C}$ is a negative cycle $\}$

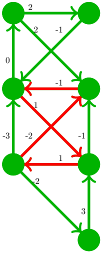
Theorem (Boros, Borys, Elbassioni, Gurvich and Khachiyan, 2005)

Given a directed graph G with real weights on its arcs, generating all negative cycles of G is ***NP-hard****. Even if $w_{ij} \in \{\pm 1\}$ for all arcs $(i,j) \in E$.*

Corollary

Generating vertices of polyhedra is hard.

Negative cycle free subgraphs' polyhedron



Let $G = (V, E)$ be a directed graph, $w : E \to \mathbb{R}$, $x \in \mathbb{R}^V$, and consider the system of linear inequalities

$$\{x_i - x_j \leq w_{ij} \quad \forall\ (i, j) \in E\}$$

$Min'l\ Infeasible\ Subsystems \leftrightsquigarrow \{\mathbf{C} \subseteq E \mid \mathbf{C} \text{ is a negative cycle } \}$

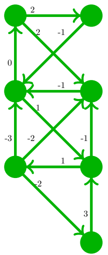
Theorem (Boros, Borys, Elbassioni, Gurvich and Khachiyan, 2005)

Given a directed graph G with real weights on its arcs, generating all negative cycles of G is ***NP-hard****. Even if $w_{ij} \in \{\pm 1\}$ for all arcs $(i, j) \in E$.*

Corollary

Generating vertices of polyhedra is hard.

Negative cycle free subgraphs' polyhedron



Let $G = (V, E)$ be a directed graph, $w : E \to \mathbb{R}$, $x \in \mathbb{R}^V$, and consider the system of linear inequalities

$$\{x_i - x_j \leq w_{ij} \quad \forall\ (i,j) \in E\}$$

Min'l Infeasible Subsystems $\leftrightsquigarrow$ $\{$**C** $\subseteq E \mid$ **C** is a negative cycle $\}$

Theorem (Boros, Borys, Elbassioni, Gurvich and Khachiyan, 2005)

Given a directed graph G with real weights on its arcs, generating all negative cycles of G is ***NP-hard****. Even if $w_{ij} \in \{\pm 1\}$ for all arcs $(i,j) \in E$.*

Corollary

Generating vertices of polyhedra is hard.

Recepie to prove Hardness of Generation

$\mathcal{I} \subseteq 2^V$ is an independence system if $Y \subseteq X \in \mathcal{I}$ implies $Y \in \mathcal{I}$

Theorem (Lawler, Lenstra, and Rinnooy Kan, 1980)

*If there is an algorithm generating the maximal independent sets of an arbitrary independence system represented by a **membership oracle** in **incremental polynomial time**, then **P=NP**.*

Given a CNF $C_1 \wedge C_2 \wedge \cdots \wedge C_m$

- Set $V = \{X_1, \bar{X}_1, \ldots, X_n, \bar{X}_n\}$
- and define $X \subseteq V$ independent if

Recepie to prove Hardness of Generation

$\mathcal{I} \subseteq 2^V$ is an independence system if $Y \subseteq X \in \mathcal{I}$ implies $Y \in \mathcal{I}$

Theorem (Lawler, Lenstra, and Rinnooy Kan, 1980)

If there is an algorithm generating the maximal independent sets of an arbitrary independence system represented by a ***membership oracle*** *in* ***incremental polynomial time****, then* ***P=NP****.*

Given a CNF $C_1 \wedge C_2 \wedge \cdots \wedge C_m$

- Set $V = \{X_1, \bar{X}_1, \ldots, X_n, \bar{X}_n\}$
- and define $X \subseteq V$ independent if

Recepie to prove Hardness of Generation

$\mathcal{I} \subseteq 2^V$ is an independence system if $Y \subseteq X \in \mathcal{I}$ implies $Y \in \mathcal{I}$

Theorem (Lawler, Lenstra, and Rinnooy Kan, 1980)

If there is an algorithm generating the maximal independent sets of an arbitrary independence system represented by a ***membership oracle*** *in* ***incremental polynomial time****, then* ***P=NP****.*

Given a CNF $C_1 \wedge C_2 \wedge \cdots \wedge C_m$

- Set $V = \{X_1, \bar{X}_1, \ldots, X_n, \bar{X}_n\}$
- and define $X \subseteq V$ **independent** if

Recepie to prove Hardness of Generation

$\mathcal{I} \subseteq 2^V$ is an independence system if $Y \subseteq X \in \mathcal{I}$ implies $Y \in \mathcal{I}$

Theorem (Lawler, Lenstra, and Rinnooy Kan, 1980)

If there is an algorithm generating the maximal independent sets of an arbitrary independence system represented by a ***membership oracle*** *in* ***incremental polynomial time****, then* ***P=NP****.*

Given a CNF $C_1 \wedge C_2 \wedge \cdots \wedge C_m$

- Set $V = \{X_1, \bar{X}_1, \ldots, X_n, \bar{X}_n\}$
- and define $X \subseteq V$ **independent** if
  - either there is an index j such that $X \cap \{X_j, \bar{X}_j\} = \emptyset$
  - or $X \cap C_i \neq \emptyset$ for all i, and $|X \cap \{X_j, \bar{X}_j\}| \leq 1$ for all j

Recepie to prove Hardness of Generation

$\mathcal{I} \subseteq 2^V$ is an independence system if $Y \subseteq X \in \mathcal{I}$ implies $Y \in \mathcal{I}$

Theorem (Lawler, Lenstra, and Rinnooy Kan, 1980)

If there is an algorithm generating the maximal independent sets of an arbitrary independence system represented by a ***membership oracle*** *in* ***incremental polynomial time****, then* ***P=NP****.*

Given a CNF $C_1 \wedge C_2 \wedge \cdots \wedge C_m$

- Set $V = \{X_1, \bar{X}_1, \ldots, X_n, \bar{X}_n\}$
- and define $X \subseteq V$ **independent** if
  - either there is an index j such that $X \cap \{X_j, \bar{X}_j\} = \emptyset$
  - or $X \cap C_i \neq \emptyset$ for all i, and $|X \cap \{X_j, \bar{X}_j\}| \leq 1$ for all j.

Recepie to prove Hardness of Generation

$\mathcal{I} \subseteq 2^V$ is an independence system if $Y \subseteq X \in \mathcal{I}$ implies $Y \in \mathcal{I}$

Theorem (Lawler, Lenstra, and Rinnooy Kan, 1980)

If there is an algorithm generating the maximal independent sets of an arbitrary independence system represented by a ***membership oracle*** *in* ***incremental polynomial time****, then* ***P=NP****.*

Given a CNF $C_1 \wedge C_2 \wedge \cdots \wedge C_m$

- Set $V = \{X_1, \bar{X}_1, \ldots, X_n, \bar{X}_n\}$
- and define $X \subseteq V$ **independent** if
  - either there is an index j such that $X \cap \{X_j, \bar{X}_j\} = \emptyset$
  - or $X \cap C_i \neq \emptyset$ for all i, and $|X \cap \{X_j, \bar{X}_j\}| \leq 1$ for all j.

Recepie to prove Hardness of Generation

$\mathcal{I} \subseteq 2^V$ is an independence system if $Y \subseteq X \in \mathcal{I}$ implies $Y \in \mathcal{I}$

Theorem (Lawler, Lenstra, and Rinnooy Kan, 1980)

*If there is an algorithm generating the maximal independent sets of an arbitrary independence system represented by a **membership oracle** in **incremental polynomial time**, then **P=NP**.*

Given a CNF $C_1 \wedge C_2 \wedge \cdots \wedge C_m$

- Set $V = \{X_1, \bar{X}_1, \ldots, X_n, \bar{X}_n\}$
- and define $X \subseteq V$ **independent** if
  - either there is an index j such that $X \cap \{X_j, \bar{X}_j\} = \emptyset$
  - or $X \cap C_i \neq \emptyset$ for all i, and $|X \cap \{X_j, \bar{X}_j\}| \leq 1$ for all j.

Examples When Generation is Hard

- Maximal infeasible solutions to a system of monotone inequalities, B, Elbassioni, Gurvich, Khachiyan and Makino, 2002.
- Maximal frequent item sets, B, Gurvich, Khachiyan and Makino, 2002.

Examples When Generation is Hard

- Maximal infeasible solutions to a system of monotone inequalities, B, Elbassioni, Gurvich, Khachiyan and Makino, 2002.
- Maximal frequent item sets, B, Gurvich, Khachiyan and Makino, 2002.

Outline

Recepie for Efficient Generation: Finding the first set ...

Assume we want to generate $\mathcal{F} = \mathbf{Min}(\mathbf{\Pi}) \subseteq 2^V$ where $\mathbf{\Pi}$ is a membership oracle for a monotone system.

- **Set** $V = \{v_1, v_2, ..., v_n\}$ and $F = V$. **If** $\mathbf{\Pi}(F) = 0$ **then** STOP $(F = \emptyset.)$
- **For** $i = 1, ..., n$ **do: if** $\mathbf{\Pi}(F \setminus \{v_i\}) = 1$ **then set** $F = F \setminus \{v_i\}$.
- **Output** F.

Recepie for Efficient Generation: Finding the first set ...

Assume we want to generate $\mathcal{F} = \mathbf{Min}(\mathbf{\Pi}) \subseteq 2^V$ where $\mathbf{\Pi}$ is a membership oracle for a monotone system.

- **Set** $V = \{v_1, v_2, ..., v_n\}$ and $F = V$. **If** $\mathbf{\Pi}(F) = 0$ **then** STOP $(F = \emptyset.)$
- **For** $i = 1, ..., n$ **do: if** $\mathbf{\Pi}(F \setminus \{v_i\}) = 1$ **then set** $F = F \setminus \{v_i\}$.
- **Output** F.

Recepie for Efficient Generation: Finding the first set ...

Assume we want to generate $\mathcal{F} = \mathbf{Min}(\mathbf{\Pi}) \subseteq 2^V$ where $\mathbf{\Pi}$ is a membership oracle for a monotone system.

- **Set** $V = \{v_1, v_2, ..., v_n\}$ and $F = V$. **If** $\mathbf{\Pi}(F) = 0$ **then** STOP ($F = \emptyset$.)
- **For** $i = 1, ..., n$ **do: if** $\mathbf{\Pi}(F \setminus \{v_i\}) = 1$ **then set** $F = F \setminus \{v_i\}$.
- **Output** F.

Recepie for Efficient Generation: Finding the first set ...

Assume we want to generate $\mathcal{F} = \mathbf{Min}(\mathbf{\Pi}) \subseteq 2^V$ where $\mathbf{\Pi}$ is a membership oracle for a monotone system.

- **Set** $V = \{v_1, v_2, ..., v_n\}$ and $F = V$. **If** $\mathbf{\Pi}(F) = 0$ **then** STOP ($F = \emptyset$.)
- **For** $i = 1, ..., n$ **do: if** $\mathbf{\Pi}(F \setminus \{v_i\}) = 1$ **then set** $F = F \setminus \{v_i\}$.
- **Output** F.

Recepie for Efficient Generation: Supergraphs

Define a directed graph $D = (W, A)$ such that

- $W = \mathcal{F}$
- There is a subset $\mathcal{F}_0 \subseteq \mathcal{F}$ "*easy to generate.*"
- For all $F \in W = \mathcal{F}$ the set $N^+(F) \subseteq W$ can be generated in incremental polynomial time.
- For all $F \in W \setminus \mathcal{F}_0$ there is an $\mathcal{F}_0 \to F$ path.

Theorem (Schwikowski and Speckenmeyer, 2002)

Then, $\mathcal{F}$ can be generated in incremental polynomial time.

Recepie for Efficient Generation: Supergraphs

Define a directed graph $D = (W, A)$ such that

- $W = \mathcal{F}$
- There is a subset $\mathcal{F}_0 \subseteq \mathcal{F}$ "*easy to generate.*"
- For all $F \in W = \mathcal{F}$ the set $N^+(F) \subseteq W$ can be generated in incremental polynomial time.
- For all $F \in W \setminus \mathcal{F}_0$ there is an $\mathcal{F}_0 \rightarrow F$ path.

Theorem (Schwikowski and Speckenmeyer, 2002)

Then, $\mathcal{F}$ can be generated in incremental polynomial time.

Recepie for Efficient Generation: Supergraphs

Define a directed graph $D = (W, A)$ such that

- $W = \mathcal{F}$
- There is a subset $\mathcal{F}_0 \subseteq \mathcal{F}$ "*easy to generate.*"
- For all $F \in W = \mathcal{F}$ the set $N^+(F) \subseteq W$ can be generated in incremental polynomial time.
- For all $F \in W \setminus \mathcal{F}_0$ there is an $\mathcal{F}_0 \to F$ path.

Theorem (Schwikowski and Speckenmeyer, 2002)

Then, $\mathcal{F}$ can be generated in incremental polynomial time.

Recepie for Efficient Generation: Supergraphs

Define a directed graph $D = (W, A)$ such that

- $W = \mathcal{F}$
- There is a subset $\mathcal{F}_0 \subseteq \mathcal{F}$ "*easy to generate.*"
- For all $F \in W = \mathcal{F}$ the set $N^+(F) \subseteq W$ can be generated in incremental polynomial time.
- For all $F \in W \setminus \mathcal{F}_0$ there is an $\mathcal{F}_0 \to F$ path.

Theorem (Schwikowski and Speckenmeyer, 2002)

Then, $\mathcal{F}$ can be generated in incremental polynomial time.

Recepie for Efficient Generation: Supergraphs

Define a directed graph $D = (W, A)$ such that

- $W = \mathcal{F}$
- There is a subset $\mathcal{F}_0 \subseteq \mathcal{F}$ "*easy to generate.*"
- For all $F \in W = \mathcal{F}$ the set $N^+(F) \subseteq W$ can be generated in incremental polynomial time.
- For all $F \in W \setminus \mathcal{F}_0$ there is an $\mathcal{F}_0 \to F$ path.

Theorem (Schwikowski and Speckenmeyer, 2002)

Then, $\mathcal{F}$ can be generated in incremental polynomial time.

Examples When Supergraphs Work

- Minimal feedback arc-sets in directed graphs (Swikowski and Speckenmeyer, 2002)
- Minimal cut conjunctions in graphs (B, Borys, Elbassioni, Gurvich, Khachiyan, and Makino, 2006)
- Prefect 2-matchings (B, Elbassioni, and Gurvich, 2006)
- Minimal edge-dominating sets (Golovach, Heggernes, Kratsch, and Villager, 2012)

Examples When Supergraphs Work

- Minimal feedback arc-sets in directed graphs (Swikowski and Speckenmeyer, 2002)
- Minimal cut conjunctions in graphs (B, Borys, Elbassioni, Gurvich, Khachiyan, and Makino, 2006)
- Prefect 2-matchings (B, Elbassioni, and Gurvich, 2006)
- Minimal edge-dominating sets (Golovach, Heggernes, Kratsch, and Villager, 2012)

Examples When Supergraphs Work

- Minimal feedback arc-sets in directed graphs (Swikowski and Speckenmeyer, 2002)
- Minimal cut conjunctions in graphs (B, Borys, Elbassioni, Gurvich, Khachiyan, and Makino, 2006)
- Prefect 2-matchings (B, Elbassioni, and Gurvich, 2006)
- Minimal edge-dominating sets (Golovach, Heggernes, Kratsch, and Villager, 2012)

Examples When Supergraphs Work

- Minimal feedback arc-sets in directed graphs (Swikowski and Speckenmeyer, 2002)
- Minimal cut conjunctions in graphs (B, Borys, Elbassioni, Gurvich, Khachiyan, and Makino, 2006)
- Prefect 2-matchings (B, Elbassioni, and Gurvich, 2006)
- Minimal edge-dominating sets (Golovach, Heggernes, Kratsch, and Villager, 2012)

Outline

Special Cases of Supergraphs: Flashlight Principle

Assume that for all $X, Y \subseteq V$, $X \cap Y = \emptyset$ we can test in polynomial time if there exists a set $F \in \mathcal{F}$ such that $Y \subseteq F$ and $X \cap F = \emptyset$.

Theorem

Then $\mathcal{F}$ can be generated with polynomial delay.

- Bridges of graphs, Tarjan, 1974
- Paths, cuts in graphs, Read and Tarjan, 1975
- ...
- Reverse search, Avis and Fukuda, 1993.
- Blockers of perfect matchings, B, Elbassioni, and Gurvich, 2006.

Special Cases of Supergraphs: Flashlight Principle

Assume that for all $X, Y \subseteq V$, $X \cap Y = \emptyset$ we can test in polynomial time if there exists a set $F \in \mathcal{F}$ such that $Y \subseteq F$ and $X \cap F = \emptyset$.

Theorem

Then $\mathcal{F}$ can be generated with polynomial delay.

- Bridges of graphs, Tarjan, 1974
- Paths, cuts in graphs, Read and Tarjan, 1975
- ...
- Reverse search, Avis and Fukuda, 1993.
- Blockers of perfect matchings, B, Elbassioni, and Gurvich, 2006.

Special Cases of Supergraphs: Flashlight Principle

Assume that for all $X, Y \subseteq V$, $X \cap Y = \emptyset$ we can test in polynomial time if there exists a set $F \in \mathcal{F}$ such that $Y \subseteq F$ and $X \cap F = \emptyset$.

Theorem

Then $\mathcal{F}$ can be generated with polynomial delay.

- Bridges of graphs, Tarjan, 1974
- Paths, cuts in graphs, Read and Tarjan, 1975
- ...
- Reverse search, Avis and Fukuda, 1993.
- Blockers of perfect matchings, B, Elbassioni, and Gurvich, 2006.

Special Cases of Supergraphs: Flashlight Principle

Assume that for all $X, Y \subseteq V$, $X \cap Y = \emptyset$ we can test in polynomial time if there exists a set $F \in \mathcal{F}$ such that $Y \subseteq F$ and $X \cap F = \emptyset$.

Theorem

Then $\mathcal{F}$ can be generated with polynomial delay.

- Bridges of graphs, Tarjan, 1974
- Paths, cuts in graphs, Read and Tarjan, 1975
- ...
- Reverse search, Avis and Fukuda, 1993.
- Blockers of perfect matchings, B, Elbassioni, and Gurvich, 2006.

Special Cases of Supergraphs: Flashlight Principle

Assume that for all $X, Y \subseteq V$, $X \cap Y = \emptyset$ we can test in polynomial time if there exists a set $F \in \mathcal{F}$ such that $Y \subseteq F$ and $X \cap F = \emptyset$.

Theorem

Then $\mathcal{F}$ can be generated with polynomial delay.

- Bridges of graphs, Tarjan, 1974
- Paths, cuts in graphs, Read and Tarjan, 1975
- ...
- Reverse search, Avis and Fukuda, 1993.
- Blockers of perfect matchings, B, Elbassioni, and Gurvich, 2006.

Special Cases of Supergraphs: Flashlight Principle

Assume that for all $X, Y \subseteq V$, $X \cap Y = \emptyset$ we can test in polynomial time if there exists a set $F \in \mathcal{F}$ such that $Y \subseteq F$ and $X \cap F = \emptyset$.

Theorem

Then $\mathcal{F}$ can be generated with polynomial delay.

- Bridges of graphs, Tarjan, 1974
- Paths, cuts in graphs, Read and Tarjan, 1975
- ...
- Reverse search, Avis and Fukuda, 1993.
- Blockers of perfect matchings, B, Elbassioni, and Gurvich, 2006.

Special Cases of Supergraphs: Flashlight Principle

Assume that for all $X, Y \subseteq V$, $X \cap Y = \emptyset$ we can test in polynomial time if there exists a set $F \in \mathcal{F}$ such that $Y \subseteq F$ and $X \cap F = \emptyset$.

Theorem

Then $\mathcal{F}$ can be generated with polynomial delay.

- Bridges of graphs, Tarjan, 1974
- Paths, cuts in graphs, Read and Tarjan, 1975
- ...
- Reverse search, Avis and Fukuda, 1993.
- Blockers of perfect matchings, B, Elbassioni, and Gurvich, 2006.

Outline

Recepie for Efficient Generation: Joint Generation

Theorem (Gurvich and Khachiyan, 1999)

Given the membership oracle $\mathbf{\Pi}$ *for a monotone property over the finite set* V, $\mathcal{H} = \mathbf{Min}(\mathbf{\Pi})$, *then the family* $\mathcal{H} \cup \mathcal{H}^d$ *can be generated in incremental quasi-polynomial time.*

Corollary

If $|\mathcal{H}^d| \leq poly(|\mathcal{H}|, |V|, |\mathbf{\Pi}|)$, *then* $\mathcal{H}$ *can be generated in quasi-polynomial total time.*

Recepie for Efficient Generation: Joint Generation

Theorem (Gurvich and Khachiyan, 1999)

Given the membership oracle $\mathbf{\Pi}$ *for a monotone property over the finite set* V, $\mathcal{H} = \mathbf{Min}(\mathbf{\Pi})$, *then the family* $\mathcal{H} \cup \mathcal{H}^d$ *can be generated in incremental quasi-polynomial time.*

Corollary

If $|\mathcal{H}^d| \leq poly(|\mathcal{H}|, |V|, |\mathbf{\Pi}|)$, *then* $\mathcal{H}$ *can be generated in quasi-polynomial total time.*

Outline

Recepie for Efficient Generation: Dual Boundedness

$\mathcal{H} \subseteq 2^V$ is uniformly dual bounded if for all $\mathcal{F} \subseteq \mathcal{H}$ we have

$$|\mathcal{F}^d \cap \mathcal{H}^d| \leq poly(|\mathcal{F}|, |V|, |\mathbf{\Pi}|).$$

Theorem (B, Gurvich, Khachiyan and Makino, 2000)

If $\mathcal{H}$ is uniformly dual bounded, then it can be generated in incremental quasi-polynomial time.

Recepie for Efficient Generation: Dual Boundedness

$\mathcal{H} \subseteq 2^V$ is uniformly dual bounded if for all $\mathcal{F} \subseteq \mathcal{H}$ we have

$$|\mathcal{F}^d \cap \mathcal{H}^d| \leq poly(|\mathcal{F}|, |V|, |\mathbf{\Pi}|).$$

Theorem (B, Gurvich, Khachiyan and Makino, 2000)

If $\mathcal{H}$ is uniformly dual bounded, then it can be generated in incremental quasi-polynomial time.

Examples When Dual Boundedness Work

- Partial and multiple transversals to hypergarphs, B, Gurvich, Khachiyan and Makino, 2000.
- Maximal sets independent in m matroids over the same base, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Disjunction of sparse boxes in m databases, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal edges sets that make each of V_i, $i = 1, ..m$ connected, B, Elbassioni, Gurvich and Khachiyan, 2002.
- Minimal collections of events the union of which have a probability exceeding a threshold, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal feasible solutions to a system of monotone linear inequalities in binary variables, B. Elbassioni, Gurvich, Khachiyan and Makino, 2002.

Examples When Dual Boundedness Work

- Partial and multiple transversals to hypergarphs, B, Gurvich, Khachiyan and Makino, 2000.
- Maximal sets independent in m matroids over the same base, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Disjunction of sparse boxes in m databases, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal edges sets that make each of V_i, $i = 1, ..m$ connected, B, Elbassioni, Gurvich and Khachiyan, 2002.
- Minimal collections of events the union of which have a probability exceeding a threshold, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal feasible solutions to a system of monotone linear inequalities in binary variables, B. Elbassioni, Gurvich, Khachiyan and Makino, 2002.

Examples When Dual Boundedness Work

- Partial and multiple transversals to hypergarphs, B, Gurvich, Khachiyan and Makino, 2000.
- Maximal sets independent in m matroids over the same base, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Disjunction of sparse boxes in m databases, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal edges sets that make each of V_i, $i = 1, ..m$ connected, B, Elbassioni, Gurvich and Khachiyan, 2002.
- Minimal collections of events the union of which have a probability exceeding a threshold, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal feasible solutions to a system of monotone linear inequalities in binary variables, B. Elbassioni, Gurvich, Khachiyan and Makino, 2002.

Examples When Dual Boundedness Work

- Partial and multiple transversals to hypergarphs, B, Gurvich, Khachiyan and Makino, 2000.
- Maximal sets independent in m matroids over the same base, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Disjunction of sparse boxes in m databases, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal edges sets that make each of V_i, $i = 1, ..m$ connected, B, Elbassioni, Gurvich and Khachiyan, 2002.
- Minimal collections of events the union of which have a probability exceeding a threshold, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal feasible solutions to a system of monotone linear inequalities in binary variables, B. Elbassioni, Gurvich, Khachiyan and Makino, 2002.

Examples When Dual Boundedness Work

- Partial and multiple transversals to hypergarphs, B, Gurvich, Khachiyan and Makino, 2000.
- Maximal sets independent in m matroids over the same base, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Disjunction of sparse boxes in m databases, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal edges sets that make each of V_i, $i = 1, ..m$ connected, B, Elbassioni, Gurvich and Khachiyan, 2002.
- Minimal collections of events the union of which have a probability exceeding a threshold, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal feasible solutions to a system of monotone linear inequalities in binary variables, B. Elbassioni, Gurvich, Khachiyan and Makino, 2002.

Examples When Dual Boundedness Work

- Partial and multiple transversals to hypergarphs, B, Gurvich, Khachiyan and Makino, 2000.
- Maximal sets independent in m matroids over the same base, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Disjunction of sparse boxes in m databases, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal edges sets that make each of V_i, $i = 1, ..m$ connected, B, Elbassioni, Gurvich and Khachiyan, 2002.
- Minimal collections of events the union of which have a probability exceeding a threshold, B, Elbassioni, Gurvich, and Khachiyan, 2002.
- Minimal feasible solutions to a system of monotone linear inequalities in binary variables, B. Elbassioni, Gurvich, Khachiyan and Makino, 2002.

Recepie for Efficient Generation: Dual Boundedness

Theorem (B, Elbassioni, Gurvich, Khachiyan, and Makino, 2005)

Almost all monotone systems are uniformly dual bounded!


Thank you!

Congratulations to the Organizing Committee!!!

- Nastja Cepak
- Nina Chiarelli
- Tatiana Romina Hartinger
- Marcin Kamiński
- Martin Milanič