On Symmetric Circuits and FPC

Matthew Anderson Anuj Dawar
University of Cambridge Computer Laboratory

27 June 2013

Context / Motivation

A circuit is symmetric if every permutation of its inputs induces an automorphisms of the circuit.

Context / Motivation

A circuit is symmetric if every permutation of its inputs induces an automorphisms of the circuit.

[Denenberg-Gurevich-Shelah '86]
Characterises first-order logic FO by uniform constant-depth poly-size symmetric Boolean circuits.

[Otto '97]
Characterises infinitary logic L_∞ by certain uniform symmetric classes of infinite Boolean circuits.

Context / Motivation

A circuit is symmetric if every permutation of its inputs induces an automorphisms of the circuit.

> [Denenberg-Gurevich-Shelah '86]
> Characterises first-order logic FO by uniform constant-depth poly-size symmetric Boolean circuits.
>
> [Otto '97]
> Characterises infinitary logic L_∞ by certain uniform symmetric classes of infinite Boolean circuits.

Theorem

P-*uniform poly-size symmetric threshold circuits* $=$ FPC.

Structures / Logic

Vocabulary τ

Finite τ-structures $\mathrm{fin}[\tau]$

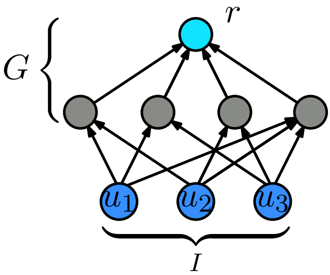
FPC Inflationary fixed-point logic extended with the ability to express the size of definable sets.

- Assume standard syntax and semantics.
- Expresses properties invariant to isomorphisms of structures.

Colored DAGs

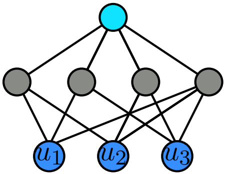
A $\mathbb{C}$-Colored Directed Acyclic Graph (CDAG) over a set U:

- Gates G
- Inputs I
- Directed edges E – form acyclic graph on $G \uplus I$ with leaves I with a single root gate r.
- Coloring $\xi : G \uplus I \to \mathbb{C}$
- Input Tuples $\lambda : I \to U^k$



Circuits

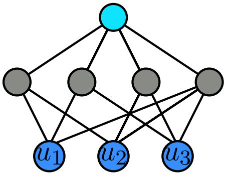
A CDAG is an abstraction of a Boolean circuit on τ-structures:



Circuits

A CDAG is an abstraction of a Boolean circuit on τ-structures:

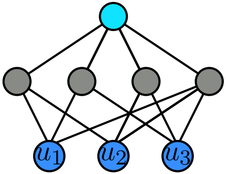
- Let $\mathbb{B} = \{\mathsf{AND}, \mathsf{OR}, \mathsf{NOT}\}$.
- Let $\mathbb{C} = \mathbb{B} \uplus \tau$.



Circuits

A CDAG is an abstraction of a Boolean circuit on τ-structures:

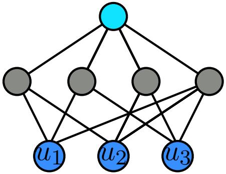
- Let $\mathbb{B} = \{\mathsf{AND}, \mathsf{OR}, \mathsf{NOT}\}$.
- Let $\mathbb{C} = \mathbb{B} \uplus \tau$.
- Color each gate with an element of $\mathbb{B}$ and each input with a relation from τ.



Circuits

A CDAG is an abstraction of a Boolean circuit on τ-structures:

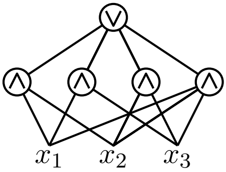
- Let $\mathbb{B} = \{\mathsf{AND}, \mathsf{OR}, \mathsf{NOT}\}$.
- Let $\mathbb{C} = \mathbb{B} \uplus \tau$.
- Color each gate with an element of $\mathbb{B}$ and each input with a relation from τ.
- Let U be the domain of the structure.



Circuits

A CDAG is an abstraction of a Boolean circuit on τ-structures:

- Let $\mathbb{B} = \{\mathsf{AND}, \mathsf{OR}, \mathsf{NOT}\}$.
- Let $\mathbb{C} = \mathbb{B} \uplus \tau$.
- Color each gate with an element of $\mathbb{B}$ and each input with a relation from τ.
- Let U be the domain of the structure.

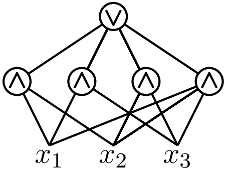


$\tau = \{X^1\}$

Circuits

A CDAG is an abstraction of a Boolean circuit on τ-structures:

- Let $\mathbb{B} = \{\text{AND}, \text{OR}, \text{NOT}\}$.
- Let $\mathbb{C} = \mathbb{B} \uplus \tau$.
- Color each gate with an element of $\mathbb{B}$ and each input with a relation from τ.
- Let U be the domain of the structure.



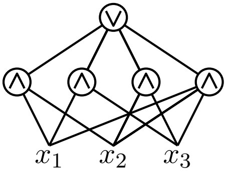
$\tau = \{X^1\}$

Each node in a circuit naturally evaluates to a Boolean value.

Circuits

A CDAG is an abstraction of a Boolean circuit on τ-structures:

- Let $\mathbb{B} = \{\mathsf{AND}, \mathsf{OR}, \mathsf{NOT}\}$.
- Let $\mathbb{C} = \mathbb{B} \uplus \tau$.
- Color each gate with an element of $\mathbb{B}$ and each input with a relation from τ.
- Let U be the domain of the structure.



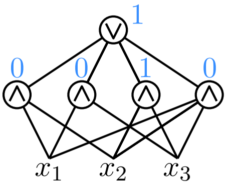
$\tau = \{X^1\}$ $X^{\mathcal{A}} = \{x_2, x_3\}$

Each node in a circuit naturally evaluates to a Boolean value.

Circuits

A CDAG is an abstraction of a Boolean circuit on τ-structures:

- Let $\mathbb{B} = \{\mathsf{AND}, \mathsf{OR}, \mathsf{NOT}\}$.
- Let $\mathbb{C} = \mathbb{B} \uplus \tau$.
- Color each gate with an element of $\mathbb{B}$ and each input with a relation from τ.
- Let U be the domain of the structure.



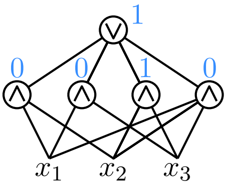
$\tau = \{X^1\}$ $X^{\mathcal{A}} = \{x_2, x_3\}$

Each node in a circuit naturally evaluates to a Boolean value.

Circuits

A CDAG is an abstraction of a Boolean circuit on τ-structures:

- Let $\mathbb{B} = \{\mathsf{AND}, \mathsf{OR}, \mathsf{NOT}\}$.
- Let $\mathbb{C} = \mathbb{B} \uplus \tau$.
- Color each gate with an element of $\mathbb{B}$ and each input with a relation from τ.
- Let U be the domain of the structure.



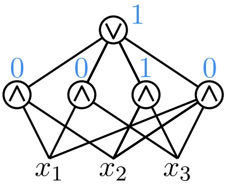
$\tau = \{X^1\}$ $X^{\mathcal{A}} = \{x_2, x_3\}$

Each node in a circuit naturally evaluates to a Boolean value.

- A circuit is invariant if the value computed at the root is independent of isomorphisms of the structure.

A CDAG is an abstraction of a Boolean circuit on τ-structures:

- Let $\mathbb{B} = \{\text{AND}, \text{OR}, \text{NOT}\}$.
- Let $\mathbb{C} = \mathbb{B} \uplus \tau$.
- Color each gate with an element of $\mathbb{B}$ and each input with a relation from τ.
- Let U be the domain of the structure.



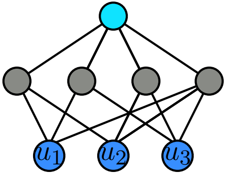
$\tau = \{X^1\}$ $X^{\mathcal{A}} = \{x_2, x_3\}$

Each node in a circuit naturally evaluates to a Boolean value.

- A circuit is invariant if the value computed at the root is independent of isomorphisms of the structure.
- A family of invariant Boolean circuits on τ-structures for all sizes of U defines a function $\text{fin}[\tau] \to \{0, 1\}$.

Symmetric CDAGs

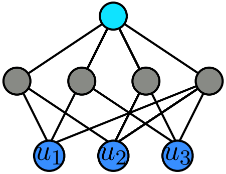
Let $C = (G, I, E, \xi, \lambda)$ be a CDAG over U.



Symmetric CDAGs

Let $C = (G, I, E, \xi, \lambda)$ be a CDAG over U.

Let $\sigma \in \mathrm{Sym}_U$ be a permutation.



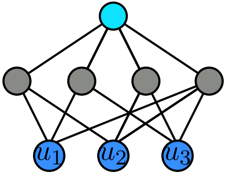
Symmetric CDAGs

Let $C = (G, I, E, \xi, \lambda)$ be a CDAG over U.

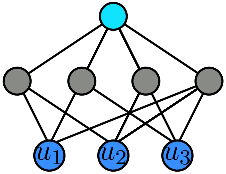
Let $\sigma \in \mathrm{Sym}_U$ be a permutation.

Consider a bijection π on the nodes of C that

- fixes the root r,



Symmetric CDAGs



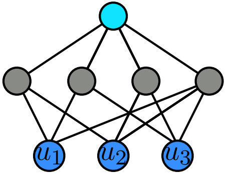
Let $C = (G, I, E, \xi, \lambda)$ be a CDAG over U.

Let $\sigma \in \mathrm{Sym}_U$ be a permutation.

Consider a bijection π on the nodes of C that

- fixes the root r,
- takes $i \in I$ to $\pi(i) \in I$ with
  1. $\xi(i) = \xi(\pi(i))$, and
  2. $\pi(\lambda(i)) = \pi(u_1, \ldots, u_k) := (\sigma(u_1), \ldots, \sigma(u_k)) = \lambda(\pi(i))$; and

Symmetric CDAGs



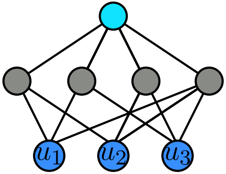
Let $C = (G, I, E, \xi, \lambda)$ be a CDAG over U.

Let $\sigma \in \mathrm{Sym}_U$ be a permutation.

Consider a bijection π on the nodes of C that

- fixes the root r,
- takes $i \in I$ to $\pi(i) \in I$ with
  1. $\xi(i) = \xi(\pi(i))$, and
  2. $\pi(\lambda(i)) = \pi(u_1, \ldots, u_k) := (\sigma(u_1), \ldots, \sigma(u_k)) = \lambda(\pi(i))$; and
- takes $g \in G$ to $\pi(g) \in G$ with
  1. $\xi(g) = \xi(\pi(g))$, and
  2. if $v \in G \uplus I$ has $(v, g) \in E$, then $(\pi(v), \pi(g)) \in E$.

Symmetric CDAGs



Let $C = (G, I, E, \xi, \lambda)$ be a CDAG over U.

Let $\sigma \in \mathrm{Sym}_U$ be a permutation.

Consider a bijection π on the nodes of C that

- fixes the root r,
- takes $i \in I$ to $\pi(i) \in I$ with
  1. $\xi(i) = \xi(\pi(i))$, and
  2. $\pi(\lambda(i)) = \pi(u_1, \ldots, u_k) := (\sigma(u_1), \ldots, \sigma(u_k)) = \lambda(\pi(i))$; and
- takes $g \in G$ to $\pi(g) \in G$ with
  1. $\xi(g) = \xi(\pi(g))$, and
  2. if $v \in G \uplus I$ has $(v, g) \in E$, then $(\pi(v), \pi(g)) \in E$.

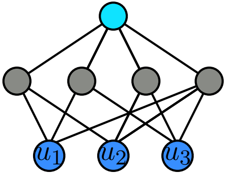
If π exists, σ induces an automorphism of C. (wlog., π is unique.)

Symmetric CDAGs

Let $C = (G, I, E, \xi, \lambda)$ be a CDAG over U.

Let $\sigma \in \mathrm{Sym}_U$ be a permutation.

Consider a bijection π on the nodes of C that

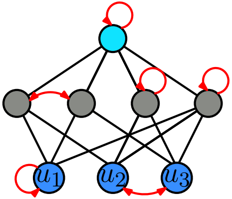


$\sigma = (u_1)(u_2 u_3)$

- fixes the root r,
- takes $i \in I$ to $\pi(i) \in I$ with
  1. $\xi(i) = \xi(\pi(i))$, and
  2. $\pi(\lambda(i)) = \pi(u_1, \ldots, u_k) := (\sigma(u_1), \ldots, \sigma(u_k)) = \lambda(\pi(i))$; and
- takes $g \in G$ to $\pi(g) \in G$ with
  1. $\xi(g) = \xi(\pi(g))$, and
  2. if $v \in G \uplus I$ has $(v, g) \in E$, then $(\pi(v), \pi(g)) \in E$.

If π exists, σ induces an automorphism of C. (wlog., π is unique.)

Symmetric CDAGs



$\sigma = (u_1)(u_2 u_3)$

Let $C = (G, I, E, \xi, \lambda)$ be a CDAG over U.

Let $\sigma \in \mathrm{Sym}_U$ be a permutation.

Consider a bijection π on the nodes of C that

- fixes the root r,
- takes $i \in I$ to $\pi(i) \in I$ with
  1. $\xi(i) = \xi(\pi(i))$, and
  2. $\pi(\lambda(i)) = \pi(u_1, \ldots, u_k) := (\sigma(u_1), \ldots, \sigma(u_k)) = \lambda(\pi(i))$; and
- takes $g \in G$ to $\pi(g) \in G$ with
  1. $\xi(g) = \xi(\pi(g))$, and
  2. if $v \in G \uplus I$ has $(v, g) \in E$, then $(\pi(v), \pi(g)) \in E$.

If π exists, σ induces an automorphism of C. (wlog., π is unique.)

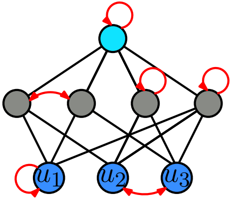
Symmetric CDAGs

Let $C = (G, I, E, \xi, \lambda)$ be a CDAG over U.

Let $\sigma \in \mathrm{Sym}_U$ be a permutation.

Consider a bijection π on the nodes of C that

- fixes the root r,
- takes $i \in I$ to $\pi(i) \in I$ with
  1. $\xi(i) = \xi(\pi(i))$, and
  2. $\pi(\lambda(i)) = \pi(u_1, \ldots, u_k) := (\sigma(u_1), \ldots, \sigma(u_k)) = \lambda(\pi(i))$; and
- takes $g \in G$ to $\pi(g) \in G$ with
  1. $\xi(g) = \xi(\pi(g))$, and
  2. if $v \in G \uplus I$ has $(v, g) \in E$, then $(\pi(v), \pi(g)) \in E$.



$\sigma = (u_1)(u_2u_3)$

If π exists, σ induces an automorphism of C. (wlog., π is unique.)

Call C symmetric if $\forall\sigma \in \mathrm{Sym}_U$, σ induces an automorphism of C.

Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U supports a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U supports a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

- If $\mathcal{S}_1$ and $\mathcal{S}_2$ support v, then so does their transitive closure.

Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U supports a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

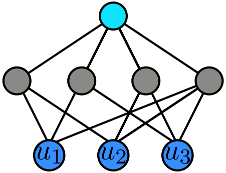
- If $\mathcal{S}_1$ and $\mathcal{S}_2$ support v, then so does their transitive closure.
  $\Rightarrow$ There is unique coarsest partition $\mathrm{Supp}(v)$ supporting v.

Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U supports a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

- If $\mathcal{S}_1$ and $\mathcal{S}_2$ support v, then so does their transitive closure.
  $\Rightarrow$ There is unique coarsest partition $\mathrm{Supp}(v)$ supporting v.

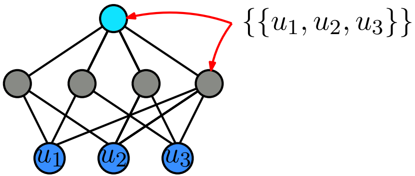


Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U supports a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

- If $\mathcal{S}_1$ and $\mathcal{S}_2$ support v, then so does their transitive closure.
  $\Rightarrow$ There is unique coarsest partition $\mathrm{Supp}(v)$ supporting v.

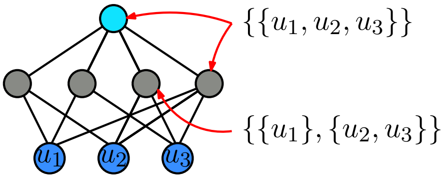


Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U **supports** a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

- If $\mathcal{S}_1$ and $\mathcal{S}_2$ support v, then so does their transitive closure.
  $\Rightarrow$ There is unique coarsest partition $\mathrm{Supp}(v)$ supporting v.

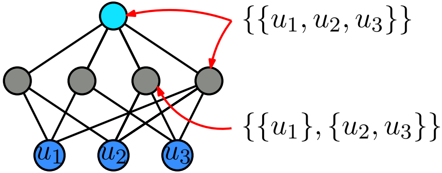


Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U **supports** a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

- If $\mathcal{S}_1$ and $\mathcal{S}_2$ support v, then so does their transitive closure.
  $\Rightarrow$ There is unique coarsest partition $\mathrm{Supp}(v)$ supporting v.



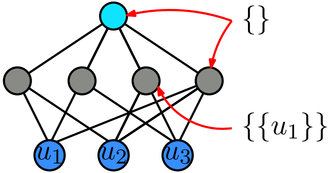
Supp induces a labelling of C.

Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U supports a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

- If $\mathcal{S}_1$ and $\mathcal{S}_2$ support v, then so does their transitive closure.
  $\Rightarrow$ There is unique coarsest partition $\mathrm{Supp}(v)$ supporting v.



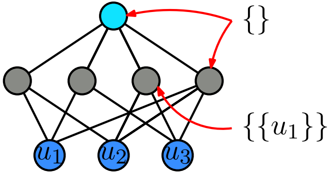
Supp induces a labelling of C.

Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U supports a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

- If $\mathcal{S}_1$ and $\mathcal{S}_2$ support v, then so does their transitive closure.
  $\Rightarrow$ There is unique coarsest partition $\mathrm{Supp}(v)$ supporting v.



Supp induces a labelling of C.

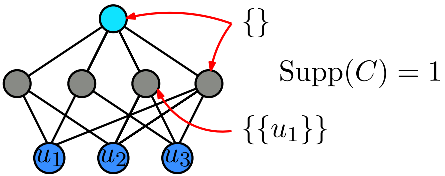
- Permutations of U act directly on this labelling.
- Define $\mathrm{Supp}(C)$ to be the maximum over all nodes v of the number of elements in all but the largest part of $\mathrm{Supp}(v)$.

Support

Let C be a symmetric CDAG over U.

A partition $\mathcal{S}$ of U supports a node $v \in C$ if every permutation of U that fixes the parts of S fix v under the induced automorphism.

- If $\mathcal{S}_1$ and $\mathcal{S}_2$ support v, then so does their transitive closure. $\Rightarrow$ There is unique coarsest partition $\mathrm{Supp}(v)$ supporting v.



Supp induces a labelling of C.

- Permutations of U act directly on this labelling.
- Define $\mathrm{Supp}(C)$ to be the maximum over all nodes v of the number of elements in all but the largest part of $\mathrm{Supp}(v)$.

Support Theorem

$\mathrm{Supp}(C)$ is tightly constrained by the number of nodes in C.

Support Theorem

$\mathrm{Supp}(C)$ is tightly constrained by the number of nodes in C.

Support Theorem

For any $1 > \epsilon \geq \frac{2}{3}$, let C be a symmetric s-node CDAG over U with $\log |U| \geq \frac{56}{\epsilon^2}$, and $s \leq 2^{|U|^{1-\epsilon}}$. Then

$$\mathrm{Supp}(C) \leq \frac{33}{\epsilon} \frac{\log s}{\log |U|}.$$

Support Theorem

$\mathrm{Supp}(C)$ is tightly constrained by the number of nodes in C.

Support Theorem

For any $1 > \epsilon \geq \frac{2}{3}$, let C be a symmetric s-node CDAG over U with $\log|U| \geq \frac{56}{\epsilon^2}$, and $s \leq 2^{|U|^{1-\epsilon}}$. Then

$$\mathrm{Supp}(C) \leq \frac{33}{\epsilon} \frac{\log s}{\log|U|}.$$

Corollary

Poly-size symmetric CDAGs have constant support.

Application: symmetric threshold circuits = FPC

The corollary leads to a characterisation of FPC.

Theorem

P-*uniform poly-size symmetric threshold circuits* $=$ FPC.

Application: symmetric threshold circuits = FPC

The corollary leads to a characterisation of FPC.

Theorem

P-*uniform poly-size symmetric threshold circuits* = FPC.

Proof Idea

1. Generate the P-uniform circuit over the number sort, using the Immerman-Vardi theorem.
2. Label gates with their support partition.
3. Transform labels into tuples by duplicating gates.
4. Determine equality test indicating edges.
5. Evaluate circuit w.r.t. unordered universe using equality test.

Bonus Application: Circuit Lower Bounds

Consider arithmetic circuits whose inputs are matrices $X \in \mathbb{F}^{U \times U}$:

- Constants 0, and 1.
- Basis $+, -,$ and $\times$.
- Variables $X = \{x_{u,v}\}_{u,v \in U}$.

Bonus Application: Circuit Lower Bounds

Consider arithmetic circuits whose inputs are matrices $X \in \mathbb{F}^{U \times U}$:

- Constants 0, and 1.
- Basis $+, -$, and $\times$.
- Variables $X = \{x_{u,v}\}_{u,v \in U}$.

The permanent $\mathrm{Per}(X)$ is the invariant polynomial:

$$\mathrm{Per}(X) := \sum_{\sigma \in \mathrm{Sym}_U} \prod_{u \in U} x_{u,\sigma(u)}$$

Bonus Application: Circuit Lower Bounds

Consider arithmetic circuits whose inputs are matrices $X \in \mathbb{F}^{U \times U}$:

- Constants 0, and 1.
- Basis $+, -$, and $\times$.
- Variables $X = \{x_{u,v}\}_{u,v \in U}$.

The permanent $\mathrm{Per}(X)$ is the invariant polynomial:

$$\mathrm{Per}(X) := \sum_{\sigma \in \mathrm{Sym}_U} \prod_{u \in U} x_{u,\sigma(u)}$$

One of the most efficient, i.e., size $2^{O(|U|)}$, ways of computing $\mathrm{Per}(X)$ known is as a symmetric multilinear formula [Ryser '57].

Bonus Application: Circuit Lower Bounds

Consider arithmetic circuits whose inputs are matrices $X \in \mathbb{F}^{U \times U}$:

- Constants 0, and 1.
- Basis $+, -$, and $\times$.
- Variables $X = \{x_{u,v}\}_{u,v \in U}$.

The permanent $\mathrm{Per}(X)$ is the invariant polynomial:

$$\mathrm{Per}(X) := \sum_{\sigma \in \mathrm{Sym}_U} \prod_{u \in U} x_{u,\sigma(u)} = \sum_{S \subseteq U} (-1)^{|U \setminus S|} \prod_{u \in U} \sum_{v \in S} x_{u,v}.$$

One of the most efficient, i.e., size $2^{O(|U|)}$, ways of computing $\mathrm{Per}(X)$ known is as a symmetric multilinear formula [Ryser '57].

Bonus Application: Circuit Lower Bounds

Consider arithmetic circuits whose inputs are matrices $X \in \mathbb{F}^{U \times U}$:

- Constants 0, and 1.
- Basis $+$, $-$, and $\times$.
- Variables $X = \{x_{u,v}\}_{u,v \in U}$.

The permanent $\mathrm{Per}(X)$ is the invariant polynomial:

$$\mathrm{Per}(X) := \sum_{\sigma \in \mathrm{Sym}_U} \prod_{u \in U} x_{u,\sigma(u)} = \sum_{S \subseteq U} (-1)^{|U \setminus S|} \prod_{u \in U} \sum_{v \in S} x_{u,v}.$$

One of the most efficient, i.e., size $2^{O(|U|)}$, ways of computing $\mathrm{Per}(X)$ known is as a symmetric multilinear formula [Ryser '57].

Theorem

Symmetric multilinear circuits for $\mathrm{Per}(X)$ *have size* $2^{|U|^{\Omega(1)}}$.

Bonus Application: Circuit Lower Bounds

Consider arithmetic circuits whose inputs are matrices $X \in \mathbb{F}^{U \times U}$:

- Constants 0, and 1.
- Basis $+, -$, and $\times$.
- Variables $X = \{x_{u,v}\}_{u,v \in U}$.

The permanent $\mathrm{Per}(X)$ is the invariant polynomial:

$$\mathrm{Per}(X) := \sum_{\sigma \in \mathrm{Sym}_U} \prod_{u \in U} x_{u,\sigma(u)} = \sum_{S \subseteq U} (-1)^{|U \setminus S|} \prod_{u \in U} \sum_{v \in S} x_{u,v}.$$

One of the most efficient, i.e., size $2^{O(|U|)}$, ways of computing $\mathrm{Per}(X)$ known is as a symmetric multilinear formula [Ryser '57].

Theorem

Symmetric multilinear circuits for $\mathrm{Per}(X)$ have size $2^{|U|^{\Omega(1)}}$.

Context: $2^{\Omega(\log^2 |U|)}$ size for multilinear formulas [Raz '08].

Summary

Support Theorem

For any $1 > \epsilon \geq \frac{2}{3}$, let C be a symmetric s-node CDAG over U with $\log|U| \geq \frac{48}{\epsilon}$, and $s \leq 2^{|U|^{1-\epsilon}}$. Then

$$\mathrm{Supp}(C) \leq \frac{24}{\epsilon} \frac{\log s}{\log|U|}.$$

Summary

Support Theorem

For any $1 > \epsilon \geq \frac{2}{3}$, let C be a symmetric s-node CDAG over U with $\log|U| \geq \frac{48}{\epsilon}$, and $s \leq 2^{|U|^{1-\epsilon}}$. Then

$$\mathrm{Supp}(C) \leq \frac{24}{\epsilon}\frac{\log s}{\log|U|}.$$

Applications:

Theorem

P-*uniform poly-size symmetric threshold circuits* $=$ FPC.

Theorem

Symmetric multilinear circuits for $\mathrm{Per}(X)$ *have size* $2^{|U|^{\Omega(1)}}$.

Open Questions

- Can the notion of support be generalised:
  - to multi-sorted domains,
  - to subgroups of Sym_U, or
  - to larger ranges of ϵ?
- Are there other applications in logic or circuit complexity?
- Is there a similar circuit characterisation of $\tilde{\mathrm{C}}\mathrm{PT(Card)}$?

Thanks!