Circuits

Gregory Wilsenach
Computer Laboratory
University of Cambridge

December 9, 2016

Contents

Chapter 1

October 25th: Rank Gates (not-functions)

This is my latest version of the support and circuit capturing result.

1.1 The Support Theorem

Definition 1. *Let A and B be sets, say that a function $f : \{0,1\}^{A\times B} \to \{0,1\}$ is matrix-symmetric if for any $\omega : A \times B \to \{0,1\}$ and $(\alpha, \beta) \in \boldsymbol{Sym}_A \times \boldsymbol{Sym}_B$ we have $f((\alpha,\beta)\cdot\omega) = f(\omega)$.*
A function $f : \{0,1\}^* \to \{0,1\}$ is matrix-symmetric iff for any two finite sets A, B we have that that $f|_{A\times B}$ is matrix-symmetric (where $f|_{A\times B}$ is the restriction of the function f to inputs sequences of size $|A \times B|$ and where elements of the input sequence are labelled by elements of $A \times B$).

Without a loss of generality we assume that A and B are always initial segments of the natural numbers.

Definition 2 (Circuits on Structures)**.** *For $\mathbb{B}_{\boldsymbol{Sym}}$ a basis of Boolean symmetric functions, $\mathbb{B}_{\boldsymbol{MatSym}}$ a basis of Boolean matrix-symmetric functions and τ a set of relation symbols, we define a $(\mathbb{B}_{\boldsymbol{Sym}}, \mathbb{B}_{\boldsymbol{Sym}}, \tau)$-circuit C_n computing a q-ary query Q is a structure $\langle G, W, \Omega, \Sigma, \Lambda, L\rangle$.*

- *G is called the set of gates of C_n and $|C_n| := |G|$.*
- *$W \subseteq G \times G$, where W is called the wires of the circuit. (G, W) must be a directed acyclic graph. For $g \in G$ we $H_g := \{h \in C_n : W(h, g)\}$ be the set of children of g.*
- *Ω is an injective function from $[n]^q$ to G. The gates in the image of Ω are called the output gates. When $q = 0$, Ω is a constant function mapping to a single output gate.*
- *Σ is a function from G to $\mathbb{B}_{\boldsymbol{Sym}} \uplus \mathbb{B}_{\boldsymbol{MatStab}} \uplus \tau \uplus \{0,1\}$ which maps input gates to $\tau \uplus \{0,1\}$ and where $\Sigma^{-1}(0) \leq 1$ $\Sigma^{-1}(1) \leq 1$ and the internal gates get mapped into $\mathbb{B}_{\boldsymbol{Sym}} \uplus \mathbb{B}_{\boldsymbol{MatStab}}$. Gates mapped to τ are called relational gates and gates mapped to 1 or 0 are called constant gates.*

- Λ *is a sequence of injective functions* $(\Lambda_R)_{R\in\tau}$ *where for each* $R \in \tau$, Λ_R *maps each relational gate* g *with* $R = \Sigma(g)$ *to the tuple* $\Lambda_R(g) \in [n]^r$, *where* r *is the arity of the symbol* R. *When no ambiguity arises we write* $\Lambda(g)$ *for* $\Lambda_R(g)$.

- L *is the set of labelings of* C_n. *Let* $g \in G$ *be an internal gate and* H *be the children of* g. *Then let* S *be a set where* $\{0,1\}^S = \boldsymbol{Dom}(\Sigma(g))$. *Then* L *is a function that assigns to* g *an onto function* $\omega_g : S \to H$. *We call this the matrix labelling of* g.

Definition 3. *Let* $\omega_1 : A \times B \to H$ *and* $\omega_2 : A' \times B' \to H'$ *define the equivalence relation* $\sim$ *as* $\omega_1 \sim \omega_2$ *iff* ω_1 *and* ω_2 *have the same domain and co-domain and there exists* $(\alpha, \beta) \in \boldsymbol{Sym}_A \times \boldsymbol{Sym}_B$ *and for all* $(a, b) \in A \times B$ *we have* $\omega_1 = \omega_2 \circ (\alpha, \beta)$.

Definition 4 (Automorphism)**.** *let* $C = \langle G, W, \Omega, \Sigma, \Lambda, L\rangle$ *be a* $(\mathbb{B}_{\boldsymbol{Sym}}, \mathbb{B}_{\boldsymbol{Sym}}, \tau)$*-circuit computing at* q*-ary query on structures of size* n. *Let* $\sigma \in \boldsymbol{Sym}_n$ *and* $\pi : G \to G$ *be a bijection such that*

- *for all gates* $g, h \in G$, $W(g, h)$ *iff* $W(\pi g, \pi h)$,

- *for all output tuples* $x \in [n]^q$, $\pi\Omega(x) = \Omega(\sigma x)$,

- *for all gates* $g \in G$, $\Sigma(g) = \Sigma(\pi g)$, *and*

- *for each relational gate* $g \in G$, $\sigma\Lambda(g) = \Lambda(\pi g)$

- *for each internal gate* g *if* $\Sigma(g) \in (B)_{\boldsymbol{MatSym}}$. *Then we have that* $L(\pi g) \sim \pi \circ L(g)$.

Remark 1. *I also need a new notion of robustness here in order to collapse gates that have the same children and equivalent labelings are properly regarded as being the same gate.*

Definition 5. *Let* ω *be a matrix labeling for* g. *Define the matrix stabilizer for* ω, *denoted by* $\boldsymbol{MatStab}(\omega)$, *to be the set of all* $\sigma \in \boldsymbol{Sym}_n$ *such that* $\sigma H = H$ *and there exists* $(\alpha, \beta) \in \boldsymbol{Sym}_A \times \boldsymbol{Sym}_B$ *such that for all* $(i, j) \in A \times B$ *we have that* $\omega(\alpha i, \beta j) = \sigma\omega(i, j)$.

Definition 6. *Let* g *be a gate with matrix labeling* ω, $h, h' \in H$ *and* $\sigma \in \boldsymbol{Sym}_n$. *We say that a pair* (h, h') *is compatible with* (σ, ω) *if* $\sigma h, \sigma h' \in H$ *and there exists* $(\gamma_1, \gamma_2), (\gamma_1', \gamma_2') \in \boldsymbol{Sym}_A \times \boldsymbol{Sym}_B$ *s.t.*

$$(\gamma_1, \gamma_2) \cdot \omega^{-1}(h) = \omega^{-1}(\sigma h), \text{ and}$$
$$(\gamma_1', \gamma_2') \cdot \omega^{-1}(h') = \omega^{-1}(\sigma h'),$$

and for all $(i, j) \in \omega^{-1}(h)$, $(i', j') \in \omega^{-1}(h')$ *we have that*

$$i = i' \implies \gamma_1(i) = \gamma_1'(i'), \text{ and}$$
$$j = j' \implies \gamma_2(j) = \gamma_2'(j').$$

Lemma 1. *Let* g *be a gate,* ω *be a matrix labeling of* g, $\sigma \in \boldsymbol{Sym}_n$. *Then* $\sigma \in \boldsymbol{MatStab}(\omega)$ *iff for all* $h, h' \in H$ *we have that* (h, h') *is compatible with* (σ, ω).

Proof. ' $\Rightarrow$': We have that $\sigma \in \mathbf{MatStab}(\omega)$ and so there exists $(\alpha, \beta) \in \mathbf{Sym}_A \times \mathbf{Sym}_B$ such that for all $(i, j) \in A \times B$ we have $\sigma\omega(i, j) = \omega(\alpha i, \beta j)$. From compatibility we have that $h, h' \in H$ Let $h, h' \in H$. Let $(\gamma_1, \gamma_2) := (\gamma_1', \gamma_2') := (\alpha, \beta)$. This assignment is sufficient to prove the direction.

' $\Leftarrow$': Suppose for all $h, h' \in H$ we have $(\gamma_1, \gamma_2), (\gamma_1, \gamma_2') \in \mathbf{Sym}_A \times \mathbf{Sym}_B$ satisfying the above requirements. Notice that for a given $i \in A$ and any $j, j' \in B$, let $h = \omega(i, j)$ and $h' = \omega(i, j')$, then we have that $\gamma_1(i) = \gamma_1'(i)$. It follows that we can define a $\alpha \in \mathbf{Sym}_A$ by $\alpha(i) = \gamma_1(i)$. Similarly we can define $\beta \in \mathbf{Sym}_B$ by $\beta(j) = \gamma_2(j)$.

□

Definition 7. *Let g be a gate with matrix labeling ω. Let $(\sigma, h, h') \in \boldsymbol{Sym}([n]) \times H^2$. Say that (σ, h, h') is useful if (h, h') is incompatible with (σ, ω).*

Say that two distinct pairs $(\sigma_1, h_1, h_1'), (\sigma_2, h_2, h_2') \in \boldsymbol{Sym}([n]) \times H^2$ are mutually independent if

- $\sigma_2 h_1 = h_1$,
- $\sigma_2\sigma_1 h_1 = \sigma_1 h_1$,
- $\sigma_2 h_1' = h_1'$,
- $\sigma_2\sigma_1 h_1' = \sigma h_1'$,

We say that a set $S \subseteq \boldsymbol{Sym}([n]) \times H^2$ is useful if each pair in it is useful. We say that S is independent if each pair of distinct pairs in S are mutually independent.

We denote the usual equivalence relation on $\mathbf{Sym}_n$ from the (right) cosets of $\mathbf{MatStab}(\omega)$ by $\sim_\omega$.

Lemma 2. *Let $\sigma_1, \sigma_1 \in \boldsymbol{Sym}_n$ and suppose $\sigma_1 \sim_\omega \sigma_2$ then for any $(h, h') \in H^2$ we have that (h, h') is compatible with (σ_1, ω) iff (h, h') is compatible with (σ_2, ω).*

Proof. Suppose we have that $\sigma_1 \sim_\omega \sigma_2$. And additionally supposed (h, h') compatible with (σ_1, ω).

It follows there exists $(\alpha, \beta) \in \mathbf{Sym}_A \times \mathbf{Sym}_B$ such that $\sigma_1\omega(i, j) = \sigma_2\omega(\alpha i, \beta j)$, and we have $(\gamma_1, \gamma_2), (\gamma_1', \gamma_2') \in \mathbf{Sym}_A \times \mathbf{Sym}_B$ satisfying the requirements of compatibility. We note that $(\alpha, \beta) \cdot \omega^{-1}(\sigma_1 h) = \omega^{-1}(\sigma_2 h)$. Thus by composition we have that $(\alpha\gamma_1, \beta\gamma_2) \cdot \omega^{-1}(h) = \omega^{-1}(\sigma_2 h)$ and $(\alpha\gamma_1', \beta\gamma_2') \cdot \omega^{-1}(h') = \omega^{-1}(\sigma_2 h')$, and clearly the remaining requirement for compatibility follows from the bijectivity of α and β. The result follows. The other direction of the implication follows from symmetry. □

Claim 1. *Let g be a rank gate with labeling ω and child set H. Let S be a useful and independent. We then have that $|\boldsymbol{Sym}_n : \boldsymbol{MatStab}(\omega)| \leq 2^{|S|}$.*

Proof. Let $R \subseteq S$ and define $\sigma_R = \Pi_{(\sigma,h,h') \in R}\sigma$ (with some arbitrary order assumed on S). Let R and Q be distinct subsets of S and WLOG let $|R| \geq |Q|$. We want to show that we don't have $\sigma_R\omega \sim_\omega \sigma_Q\omega$. Pick any $(\sigma, h, h') \in R/Q \neq \emptyset$. Given that $\sigma_R h = \sigma h$ and $\sigma_R h' = \sigma h'$ it is easy to see that the usefulness of (σ, h, h') implies the incompatibility of (h, h') with with (ω, σ_R). Moreover, the fact that $\sigma_Q h = h$ and $\sigma_Q h' = h'$ makes it easy to see (h, h') is compatible with (ω, σ_Q). From the above lemma we may conclude that that we do not have $\sigma_R \sim_\omega \sigma_Q$, and the result follows. □

The following two lemmas proved by Anderson and Dawar [] are both of use in proving the following theorem.

Lemma 3. *For any ϵ and n such that $0 < \epsilon < 1$ and $\log n \geq \frac{4}{\epsilon}$, if $\mathcal{P}$ is a partition of $[n]$ with k parts, $s = [\boldsymbol{Sym}_n : \boldsymbol{SetStab}(\mathcal{P})]$ and $n \leq sleq2^{n^{1-\epsilon}}$, then $\min\{k, n-k\} \leq \frac{8}{\epsilon}\frac{\log s}{\log n}$.*

Lemma 4. *For any ϵ and n such that $0 < \epsilon < 1$ and $\log n \geq \frac{8}{\epsilon^2}$, if $\mathcal{P}$ is a partition of $[n]$ with $|\mathcal{P}| \leq \frac{n}{2}$, $s := [\boldsymbol{Sym}_n : \boldsymbol{SetStab}(\mathcal{P})]$ and $n \leq s \leq 2^{n^{1-\epsilon}}$, then $\mathcal{P}$ contains a part P with at least $n - \frac{33}{\epsilon} \cdot \frac{\log s}{\log n}$.*

If g is a symmetric gate (i.e. the usual gates in a circuit) we note that $\mathbf{Orb}(g) = [\mathbf{Sym}_n : \mathbf{Stab}(g)]$ by the orbit-stabilizer theorem.

If g is a matrix-symmetric gate then $\mathbf{Orb}(g) = [\mathbf{Sym}_n : \mathbf{MatStab}(\omega)]$, where ω is the matrix labelling associated with g.

Theorem 5. *For any ϵ and n such that $\frac{2}{3} \leq \epsilon \leq 1$ and $n \geq \frac{128}{\epsilon^2}$, if C is a symmetric, rigid circuit on structures of size n and $s := \max_{g \in C} |\boldsymbol{Orb}(g)| \leq 2^{n^{1-\epsilon}}$, then, $SP(C) \leq \frac{33}{\epsilon}\frac{logs}{logn}$.*

Proof. It is easy to see that if g is a gate in C then $\mathbf{Stab}(g) \subseteq \mathbf{SetStab}(\mathbf{SP}(g))$, and so $s \geq \mathbf{Orb}(g) = [\mathbf{Sym}_n : \mathbf{Stab}(g)] \geq [\mathbf{Sym}_n : setstab(\mathbf{SP}(g))]$. Thus if $|\mathbf{SP}(g)| \leq fracn2$, then from Lemma 4, we have $\|SP(g)\| \leq \frac{33}{\epsilon} \cdot \frac{\log s}{\log n}$. The result thus follows from showing that for each g in C we have that $|\mathbf{SP}(g)| \leq \frac{n}{2}$.

The cases where g is a constant or relational gate are easy to handle.

We now consider the case for internal gates. Let g be the topologically first internal gate with $|\mathbf{SP}(g)| > \frac{n}{2}$. If g is not a matrix-symmetric gate then the result follows from the argument presented by Anderson and Dawar []. Suppose g is a matrix-symmetric gate, and suppose g has a labelling ω. We now argue that this leads to a contradiction.

Let $k' := \lceil \frac{8 \log s}{\epsilon \log n} \rceil$. From the assumptions on s, n and ϵ we have that $k' \leq \frac{1}{4} n^{1-\epsilon} < \frac{n}{2}$. Lemma 3 implies that $n - |\mathbf{SP}(g)| \leq k'$

From Claim 1 it remains to show that we can construct a sufficiently large useful and independent set of gate-automorphism pairs S. Divide $[n]$ into $\lfloor \frac{n}{k'+2} \rfloor$ disjoint sets S_i of size $k'+2$ and ignore the elements left over. It follows that for each i there is a permutation σ_i which fixes $[n]/S_i$ pointwise but moves g. Suppose there was no such σ_i it follows that every every permutation that fixes $[n]/S_i$ pointwise fixes g. Thus the partition of all the singletons in $[n]/S_i$ and S_i is a supporting partition. As $\mathbf{SP}(g)$ is the coarsest partition it follows that $|\mathbf{SP}(g)| \leq n - (k'+2) + 1 = n - k' - 1$, which contradicts the inequality $n - |\mathbf{SP}(g)| \leq k'$.

Since g is moved by each σ_i and C is rigid it follows that we don't have $\sigma_i \notin \mathbf{MatStab}(\omega)$. Thus there exists (h_i, h_i') that is inconsistent with (σ_i, ω), and so the triple (σ_i, h_i, h_i') is useful.

Let $\mathbf{SP}(h)^*$ be the union of all parts of $\mathbf{SP}(h)$ except for the largest part. Let $Q_i = \mathbf{SP}(h_i)^* \cup \mathbf{SP}(\sigma_i h_i)^* \cup \mathbf{SP}(h_i')^* \cup \mathbf{SP}(\sigma_i h_i')^*$. Then note that if σ_j fixes Q_i then by construction, we have that $\sigma_j \in \mathbf{Stab}_n(\mathbf{SP}(h_i)) \cap \mathbf{Stab}_n(\mathbf{SP}(\sigma_i h_i)) \cap \mathbf{Stab}_n(\mathbf{SP}(h_i')) \cap \mathbf{Stab}_n(\mathbf{SP}(\sigma_i h_i'))$

Define a graph K with vertices given by the sets S_i and an edge from S_i to S_j (with $i \neq j$) if $Q_i \cap S_j \neq \emptyset$. It follows then that if there is no edge between S_i and S_j then (σ_i, h_i, h_i') and (σ_j, h_j, h_j') are mutually independent. It remains to argue that K has a large independent set. This is possible as the out-degree of S_i in K is bounded by

$$|Q_i| \leq \|SP(h_i)\| + \|SP(\sigma_i h_i)\| + \|SP(h_i')\| + \|SP(\sigma_i h_i')\| \leq 4 \cdot \frac{33 \log s}{\epsilon \log n}$$

.

This follows as the sets S_i are disjoint and we may apply Lemma 4 to each of the child gates. It follows that the average total degree (in + out degree) of K is at most $2 \cdot |Q_i| \leq 34 \cdot k'$. Now greedily select a maximal independent set in K by repeatedly selecting S_i with the lowest total degree and eliminating it and its neighbours. This action does not affect the bound on the average total degree of K and hence determines an independent set I in K of size at least

$$\frac{\lfloor \frac{n}{k'+2} \rfloor}{34k'+1} \geq \frac{n-(k'+2)}{34k'+1k'+2} \geq \frac{n\frac{7}{16}}{34k'^2+69k'+2} \geq \frac{n}{(16k')^2}.$$

Take $S = \{(\sigma, h, h') : S_i \in I\}$. Then from the above argument we have that S is useful and independent.

Moreover, from Claim 1, we have that $s \geq |\mathbf{Orb}(g)| \geq 2^{|S|} \geq 2^{\frac{n}{(16k')^2}}$ then $n^{1-\epsilon} \geq \log s \geq n \cdot (\frac{128}{\epsilon} \frac{\log s}{\log n})^{-2} > n \cdot (n^{1-\epsilon})^{-2} = n^{2\epsilon-1} \geq n^{1-\epsilon}$. This is a contradiction. □

1.2 Rigid Circuits

Definition 8. *We say that a circuit C has bijective labels if for each gate g in C, $L(g)$ is a bijection.*

Lemma 6. *There is an algorithm that runs in polynomial time that takes in a circuit C and outputs a circuit with unique gates C'. Moreover, if C was symmetric then C' is symmetric. If C is rigid then C' is rigid.*

Proof. Let $S = 2 * |C|$. Recurse through the gates of C topologically and let h be the next gate topologically. If for all $g \in W(h, \cdot)$ we have that $L|(g)^{-1}(h)| = 1$ continue on to the next gate. If not add in a tower of S gates such that $h \to^h_1 \to \ldots \to^h_S$ (i.e. we have a tower of gates with h as input to $^h_1$ and the output of each ^h_i connected to the input of each $^h_{i+1}$ for each $1 \leq i < S$). Now for each $g \in W(h, \cdot)$, if $L(g)^{-1}(h) = \{s_0, \ldots, s_r\}$, for each $1 \leq i \leq r$ add in the wires $W(^h_i, g)$ and set $L(g)(s_i) =^h_i$. Now continue on to the next gate topologically and run the above algorithm.

We call this updated circuit C'.

Firstly, note that after running the above algorithm for each g the labelling of g will be a bijection. Moreover, it's easy to see that the output of each gate remains unchanged, and as such the output of the circuit is unchanged.

Secondly, notice that the size of C' is at most $2 * |C|^2$, and note that the above algorithm runs in polynomial time.

Now suppose that C is symmetric. Let σ be a permutation on the input universe and π_C the induced automorphism on C. We now define π, the induced automorphism on C'. For each gate g in C', if g is in C then set $\pi(g) := pi_C(g)$. If g is not in C then g must be some ^h_i, for some h in C. Then set $\pi(^h_i) :=^{\pi_C(h)}_i$. It is easy to see that π is an automorphism, and π extends σ.

Suppose that C is rigid. It is easy to see that C' will be rigid as well.

□

Lemma 7. *There is an algorithm that runs in polynomial time that takes in a circuit C and outputs a circuit C' such that C' is rigid and has bijective labels. Moreover, if C was symmetric it follows that C' will be symmetric.*

Proof. First run the algorithm from Lemma 6 on C, and call the output circuit C.

Recurse through the gates of C topologically. For each internal gate g, for all g' in C such that $g \neq g'$, $W(\cdot, g) = W(\cdot, g')$, $W(g, \cdot) = W(g', \cdot)$, $\Sigma(g) = \Sigma(g')$, $L(g) \sim L(g')$ and $\Omega^{-1}(g) = \Omega^{-1}(g')$, delete g' and for all $s \in L^{-1}(g')$ set $L(s) := g$.

Now re-run the algorithm from Lemma 6 on C and output the result. □

Lemma 8. *Let $C = \langle G, W, \Omega, \Sigma, \Lambda, L\rangle$ be a $(\mathbb{B}_{\boldsymbol{Sym}}, \mathbb{B}_{\boldsymbol{MatSym}}, \tau)$ − circuit on structures of size n. There is a deterministic algorithm which runs in Poly($|C|$) and outputs a rigid $(\mathbb{B}_{\boldsymbol{Sym}}, \mathbb{B}_{\boldsymbol{MatSym}}, \tau)$circuit C' such that $G' = G$ and for any $g \in G$, and any input τ-structure $\mathcal{A}$ and any bijection γ from A to $[n]$, $C[\gamma\mathcal{A}](g) = C'[\gamma\mathcal{A}](g)$ and if C is symmetric then so is C'.*

Proof. □

1.3 Computing Supports

Lemma 9. *Let C be a rigid $(\mathbb{B}_{\boldsymbol{Sym}}, MB, \tau)$-circuit on structures of size n and $\sigma \in \boldsymbol{Sym}_n$. There is a deterministic algorithm which runs in time Poly($|C|$) and outputs for each gate g its image under the automorphism π induced by σ, if it exists.*

Proof. The proof proceeds by recursively going through the circuit and building the mapping π induced by σ.

Suppose g is a constant gate, then $\pi g := g$. Suppose g is a relational gate, then there is at most one gate g' such that $\Sigma(g) = \Sigma(g')$ and $\sigma\Lambda(g') = \Lambda(g)$. If such a g' exists assign $\pi g := g'$, else terminate with failure.

If g is an symmetric internal gate then (from rigidity) there is at most one gate g' such that $\Sigma(g) = \Sigma(g')$ and $W_{g'} = \pi W_g$. Assign $\pi g := g'$ if such a gate exists, or else terminate with failure.

If g is a matrix-symmetric internal gate then consider the set of gates g' such that g' has children πW_g and $\Sigma(g) = \Sigma(g')$, and let $A \times B = \mathbf{Dom}(Sigma(g))$. If no such gate g' exists, terminate with failure. Define $\sigma_{\pi,g'} : A \times B \to A \times B$ by $\sigma_{\pi,g'} = \omega_{g'}^{-1}\pi\omega_g$. Then clearly $\omega_{g'}\sigma_{\pi,g'} = \pi\omega_g$, and it's easy to show that $\pi\omega_g \sim \omega_{g'}$ iff $\sigma_{\pi,g'} \in \mathbf{Sym}_A \times \mathbf{Sym}_B$. But this just involves checking that σ acts as a bijection on A and B separately and, given that $|A|$ and $|B|$ are both bounded by $|C|$, the algorithm which just iterates through A and B is sufficient. If for every g' it is found that $\pi_{\sigma,g'}$ is not in $\mathbf{Sym}_A \times \mathbf{Sym}_B$ then terminate with failure. If there is a g' for which $\pi_{\sigma,g'} \in \mathbf{Sym}_A \times \mathbf{Sym}_B$ then it is unique by rigidity and so set $\pi g := g'$.

If g is an output gate, then check that for all tuples in $[n]^q$ we have that $\pi\Omega(x) = \Omega(\sigma(x))$, and terminate with failure if the condition is not met.

If the algorithm has not terminated with failure, output the automorphism.

The algorithm clearly runs in Poly($|C|$) □

1.4 Evaluating Symmetric Circuits

1.5 Translating to Formulas of FPC

Let $\mathcal{C} = (C_n)_{n\in\mathbb{N}}$ be a P-uniform family of polynomial-size symmetric $(\mathbb{B}_{\mathbf{Sym}}, \mathbb{B}_{\mathbf{MatSym}}, \tau)$ circuits, and where $\mathbb{B}_{\mathbf{MatSym}}$ is the rank basis. It remains to show that there is a formula Q in the vocabulary $\tau \uplus \{\leq\}$ such that for any n and τ-structure $\mathcal{A}$ with a universe U of carnality n, the q-ary query defined by C_n on the input $\mathcal{A}$ is defined by the formula Q when interpreted in the structure $\mathcal{A}^{\leq} = \mathcal{A} \uplus \langle [n], \leq \rangle$.

Since $\mathcal{C}$ is P-uniform, and from Lemmas **??** and **??** and the Immerman-Vardi theorem, we have an FP interpretation defining a rigid symmetric circuit with bijective labels equivalent to C_n (which we also call C_n) over the the number sort of $\mathcal{A}^{\leq}$, where $\Phi = (\phi_G, \phi_W, \phi_\Omega, (\phi_s)_{s\in\mathbb{B}_{\mathbf{Sym}}\uplus\{rank\}\uplus\tau\uplus\{0,1\}}, (\phi_{_R})_{R\in\tau}, \phi_L)$. We note that Phi is a t-width interpretation over the universe $[n]$. So if $\mu, \nu, \nu_1, \ldots \in [n]^t$, the formulas are defined such that:

- $\phi_G(\mu)$ holds iff μ encodes a gate
- $\phi_W(\nu, \mu)$ holds iff ν and μ encode gates and if g_ν and g_μ are these two encoded gates respectively then $W(g_\nu, g_\mu)$.
- $\phi_\Omega(\nu_1, \ldots, \nu_q, \mu)$ holds iff $\nu_1, \ldots, nu_q$ each encode numbers less than or equal to $[n]$ and $\Omega(\nu_1, \ldots, \nu_q) = g_\mu$.
- For all $s \in \mathbb{B}_{\mathbf{Sym}}\uplus\{rank\}\uplus\tau\uplus\{0,1\}$ we have that $\phi_s(\mu)$ holds iff μ encodes a gate such that $\Sigma(g_\mu) = s$ (or, in the case that $s = \mathbf{rk}$, we require that $\Sigma(g_\mu)$ is a rank symbol.)
- For all $R \in \tau$ with arity r we have that $\phi_{_R}(\nu_1, \ldots, \nu_r, \mu)$ holds iff $\nu_1, \ldots, \nu_r$ encodes numbers less than or equal to $[n]$, μ encodes a gate such that $\Sigma(g_\mu) = R$ and $\Lambda_R(g_\mu) = (\nu_1, \ldots, \nu_r)$.
- $\phi_L(\nu, \mu, \nu_1, nu_2)$ holds iff ν and μ encode gates and ν_1, ν_2 encode numbers, and $L(g_\mu)(\nu_1, \nu_2) = g_\nu$. If $\mathbf{Dom}(L(g_\mu)) = [a] \times [b]$, then the encoding of $[a]$ and $[b]$ are initial segments under the lithographically induced order. Moreover, the encodings of $[a]$ and $[b]$ preserve the order relation.

From now on we use μ and ν to stand for t-length sequences encoding gates, and use $\nu_1, \ldots$ for sequences that encode numbers. We use g_μ and g_ν for the gates encoded by these sequence, while simply identifying the sequence ν_i with the number it encodes.

Using the Immerman-Vardi Theorem, we can define an FP($\leq$) formula CHAR such that $\langle [n], \leq \rangle \models \text{CHAR}[g, p, u]$ iff $\langle [n], \leq \rangle \models \phi_G[g] \wedge \phi_{rank}[g]$ and the rank gate g (as per the interpretation) computes the rank over characteristic p and has threshold u.

We can define FP($\leq$) formulas $\max_A$ and $\max_B$ such that $\langle [n], \leq \rangle \models \max_A[g, \nu_1]$ iff ν_1 encodes a, where $\mathbf{Dom}(L(g)) = [a] \times [b]$. Similarly, $\langle [n], \leq \rangle \models \max_B[g, \nu_1]$ iff ν_1 encodes b where $\mathbf{Dom}(L(g)) = [a] \times [b]$.

Again using Lemma **??** and the Immerman-Vardi theorem, we can construct a formula SUPP such $\langle [n], \leq \rangle \models \text{SUPP}[g, u]$ iff $\langle [n], \leq \rangle \models \phi_G[g]$ and u is in $sp(g)$. This formula can

be used as in [] to inductively define $\mathrm{SUPP}_i(g, u)$ for each $i \in [n]$ which holds iff u is the ith element of the support of g.

Define the AGREE and θ_s formulas for all $s \in \mathbb{B}_{\mathbf{Sym}} \uplus \tau \uplus \{0, 1\}$ as in [].

Now we define the formula

$$\theta_{\mathbf{rk}} := (\mu, \bar{x}) := \bigwedge_{1 \leq i < j \leq k} x_i \neq x_j \wedge \forall \bar{y}([\mathbf{rk}(x \leq \phi_{mr}, y \leq \phi_{mc}, \pi \leq r).\theta^{'}]),$$

$$\theta^{'}(a, b) := \exists \nu(W(\nu, \mu) \wedge \mathrm{AGREE}(\mu, \nu, \bar{x}, \bar{y}) \wedge \phi_L(\mu, \nu, a, b) \wedge V(\nu, y))$$

Chapter 2

October 27th: Counting Gates Simulate Symmetric Gates

In this chapter we prove that symmetric circuits with majority gates are at least as powerful as symmetric circuits over any other Boolean basis consisting of symmetric functions.

Recall the following that $\mathbb{B} = \{\neg, \wedge, \vee\}$ and $\mathbb{B}_{\mathbf{Maj}} = \{\mathbf{Maj}\} \cup \mathbb{B}$.

Let $F : \{0,1\}^* \to \{0,1\}$ be a symmetric Boolean function. Since F is symmetric we note that for a fixed size input the output of F is entirely determined by the number of 1's in its input. Then let $c_F : \mathbb{N} \to 2^{\mathbb{N}}$ define a function where $c_F(n)$ is the set of all $m \leq n$ such that for all $\vec{x} \in \{0,1\}^n$ with m 1's we have $F(\vec{x}) = 1$. Clearly a symmetric Boolean function F is entirely determined by c_F.

Proposition 10. *There is a polynomial $p(k)$ such that for any symmetric function F and a given $k \in \mathbb{N}$ there is a circuit C_k on k inputs over the basis $\mathbb{B}_{\boldsymbol{Maj}}$ which is symmetric, constant depth and with width bounded by $p(k)$.*

Proof. We define the circuit C_k for inputs $\vec{x} = (x_1, \ldots, x_k)$.

For $a \in \mathbb{N}$ we define a get count_a by

$$\text{count}_a = \mathbf{Maj}(x_1, \ldots, x_k, \underbrace{0, \ldots, 0}_{2a-k}) \wedge \neg\mathbf{Maj}(x_1, \ldots, x_k, \underbrace{0, \ldots, 0}_{2a-k+2}) \text{ if } a \geq \frac{k}{2},$$

$$\text{count}_a = \mathbf{Maj}(x_1, \ldots, x_k, \underbrace{1, \ldots, 1}_{x-2a}) \wedge \neg\mathbf{Maj}(x_1, \ldots, x_k, \underbrace{1, \ldots, 1}_{k-2a-2}) \text{ if } a < \frac{k}{2}.$$

Then let $g = \bigvee_{a \in c_{F_i}(k)} \text{count}_a$ and let C_k be the circuit with input gates labeled by $\vec{x}$ and output gate g.

t is easy to see that C_k is constant depth and it's width is a polynomial in k. We have that in each layer of $C_{g'}$ each gate is connected to all gates in the previous layer, and as such the circuit is symmetric. □

The above proposition has a straight forward application to circuit characterizations.

Proposition 11. *Let $F = \{F_i : i \in I\}$ be a family of symmetric Boolean functions where $F_i : \{0,1\}^* \to \{0,1\}$.*

Let $(C_n)_{n\in\mathbb{N}}$ be family of symmetric circuits over the Boolean basis $\mathbb{B}\cup F$, where C_n is a circuit on structures of size n, and the size of each circuit in the family is bounded by some function $f(n)$. Then there exists a polynomial $q(n)$ and a family of symmetric circuits $(C'_n)_{n\in\mathbb{N}}$ over $\mathbb{B}_{\boldsymbol{Maj}}$, where C'_n is a circuit on structures of size n and $|C'_n| \leq q(f(n))$.

Proof. From C_n we construct C'_n in the obvious way. For each gate $g \in C_n$ of type F_i we have a symmetric circuit C_g from Proposition **??** that computes the same function as g. Then let C'_n be C_n but with each gate g replaced by C_g. It is easy to see that C'_n is symmetric. We also have that each gate g must have at most $f(n)$ inputs, and the size of C_g is bounded by $p(f(n))$. Thus the size of C'_n is bounded by $f(n)p(f(n))$. □

Chapter 3

October 29th: Symmetric Circuits and CPT

We already have that that the set of queries decidable by symmetric circuits (respectively with counting) is exactly the set of queries expressible in Fixed-Point Logic. We know that the CFI query is not expressible in FPC, and so there is no symmetric circuit that can compute the query.

In our attempts to find a circuit characterisation of CPT, we might try and weaken the requirement of symmetry in our notion of symmetric circuits in order to express the CFI query (and perhaps the whole of CPT). One approach would be to weaken the requirement of having small orbits, to rather only having small orbits with respect to our particular input structure. We formalise this notion below.

Definition 9. *Let C be a circuit on structures of size n. Let g be a gate in C and let X be a set of input strings closed under the action of the symmetric group. The $inv_X = \{\sigma \in \boldsymbol{Sym}_n : \forall x \in X g \circ H(x) = g \circ H(\sigma x)$.*

Definition 10. *Let $(C_n)_{n\in\mathbb{N}}$ be a family of circuits. For any finite structure $\mathcal{A}$ let $X_\mathcal{A}$ be the set of all string encodings of $\mathcal{A}$. We say a circuit family has small isomorphism orbits if there is a polynomial $p(n)$ such that for any sufficiently large n, finite structure $\mathcal{A}$ of cardinality n and any $g \in C_n$ we have $|S_n : \boldsymbol{inv}_{X_\mathcal{A}}(g)| \leq p(n)$.*

Theorem 12. *Let $n \in \mathbb{N}$, $n \geq 5$. Let $r \in \mathbb{N}$ such that $r \leq n/2$. Suppose that G is a subgroup of $\boldsymbol{Sym}(n)$, then there is a constant $k \in mathbbN$ such that for all $n \geq k$ if $|\boldsymbol{Sym}(n) : G| \leq \binom{n}{r}$ then there exists $X \subseteq [n]$ with $|X| \leq r$ and such that $\boldsymbol{Alt}([n] \setminus X) \leq G$.*

The following result shows that even this weakened definition fails to capture CPT.

Theorem 13. *There is no family of circuits with small isomorphism orbits that expresses the CFI query.*

Proof. Let $(G_n)_{n\in\mathbb{N}}$ be a family of ordered graphs such that each G_n has odd and even CFI graphs $\mathfrak{G}$ and $\tilde{\mathfrak{G}}$ respectively, each of cardinality n. Furthermore, suppose the CFI query for (G_n) is not expressible in FPC (the existence of such a family is guaranteed by []). Suppose there exists a family of circuits $(C_n)_{n\in\mathbb{N}}$ with small isomorphism orbits that decides the CFI query for (G_n). Then we have that $C_n(\mathfrak{G}) = 1$ and $C_n(\tilde{\mathfrak{G}}) = 0$, and for any $g \in C_n$

$$|S_n : \mathbf{inv}_{X_{\mathcal{G}}}(g)| \leq p(n),$$
$$|S_n : \mathbf{inv}_{X_{\tilde{\mathcal{G}}}}(g)| \leq p(n)$$

. Let k be such that $p(n) \leq n^k$ for all sufficiently large n. Then we have that $p(n) \leq n^k \leq \binom{n}{k+1}$ for all sufficiently large n, and (again, taking n large enough) we have that $k \leq n/2$. Taking n larger than the constant in Theorem 12 it follows that there exists an $X_1, X_2 \subseteq [n]$ such that $|X_1|, |X_2| \leq k$ and $\mathbf{Alt}([n] \setminus X_1) \subseteq \mathbf{inv}_{X_{\mathfrak{G}}}(g)$, $\mathbf{Alt}([n] \setminus X_2) \subseteq \mathbf{inv}_{X_{\tilde{\mathfrak{G}}}}(g)$. It follows that

$$\begin{aligned}\mathbf{Alt}([n] \setminus (X_1 \cup X_2)) &\subseteq \mathbf{Alt}([n] \setminus X_1) \cap \mathbf{Alt}([n] \setminus X_2) \\ &\subseteq \mathbf{inv}_{X_{\mathfrak{G}}}(g) \cap \mathbf{inv}_{X_{\tilde{\mathfrak{G}}}}(g) \\ &\subseteq \mathbf{inv}_{X_{\mathfrak{G} \cup \tilde{\mathfrak{G}}}}(g) = \mathbf{inv}_{CFI(G)}(g),\end{aligned}$$

where $CFI(G)$ is set of all encodings of all CFI graphs of G_n.

We then have that

$$|S_n : \mathbf{inv}_{CFI(G)}(g)| \leq |S_n : \mathbf{Alt}([n] \setminus (X_1 \cup X_2)) \leq \frac{n!}{(n-2k)!/2} \leq n^{2k+1},$$

with the last inequality following for large enough n. We thus have from Theorem **??** that there is a symmetric circuit D_n that decides the CFI problem for G_n, and hence a family (D_n) that decides the CFI problem for the family of graphs (G_n). From the theorem of Dawar et al. [], it follows that there is as formula of FPC that decides the CFI query for (G_n) contradicting our assumption. □