Simulating Mostly Classical Circuits

Arjun Aggarwal

August 2024

1 Simulating $\mathsf{U} \circ \mathsf{AC}^0$ Circuits

Definition 1 ($\mathsf{U} \circ \mathsf{AC}^0$ Circuits)**.** A circuit C is a $\mathsf{U} \circ \mathsf{AC}^0$ circuit if C contains a layer of arbitrary 1-qubit unitaries, followed by a standard basis measurement of each qubit, and an AC^0 circuit that operates on the result of the measurements.

The main result of this section is the following theorem

Theorem 2. *Let C be a $\mathsf{U} \circ \mathsf{AC}^0$ circuit of size s that acts on n input qubits and a ancillas and achieves $1/2+\epsilon$ correlation with a boolean function f. There exists a randomized AC^0 circuit C_R of size poly(s) that achieves $1/2 + \epsilon'$ correlation with f, where $|\epsilon - \epsilon'| \leq 2^{-n}$*

Any $\mathsf{U} \circ \mathsf{AC}^0$ circuit C can be written as $C = \left(\bigotimes_{j=1}^{m} U_j\right) C$, where U_j is a single-qubit unitary, $m := n+a$ and C is a AC^0 circuit. We adopt the convention that the first n qubits are reserved for the input and the rest a qubits are ancillas. Since we allow arbitrary single-qubit unitaries, we can assume without loss of generality that the ancilla qubits are initialized to $|0^a\rangle$.

Let $x \in \{0,1\}^n$ be the input to C and $y = (y_j)$ be the string obtained by measuring in the standard basis after applying the unitary layer. Call the distribution from which y is sampled $\mathcal{U}_x$. The following lemmas show that it is sufficient to pointwise approximate $\mathcal{U}_x$ to 2^{-2n} precision to prove Theorem 2.

Lemma 3. *Let $\mathcal{U}'_x$ be a distribution such that $|\mathcal{U}_x(y) - \mathcal{U}'_x(y)| \leq 2^{-2n}$ for any $y \in \{0,1\}^m$. Then,*

$$\mathbb{P}_{\substack{y \sim \mathcal{U}_x \\ y' \sim \mathcal{U}'_x}} \left[C(y) \neq C(y')\right] \leq 2^{-2n}$$

Lemma 4. *Let C_1 and C_2 be randomized circuits such that for any $x \in \{0,1\}^n$,*

$$\mathbb{P}\left[C_1(x) \neq C_2(x)\right] \leq 2^{-2n}$$

If C_1 achieves correlation ϵ_1 with some boolean function f, then C_2 achieves correlation ϵ_2 where $|\epsilon_1 - \epsilon_2| \leq 2^{-n}$.

Thus, via Lemma 3 and 4, we reduce the problem of simulating $\mathsf{U} \circ \mathsf{AC}^0$ circuits to analyzing $\mathcal{U}_x$ and sampling from a nearby distribution using a randomized AC^0 circuit.

1.1 Analyzing $\mathcal{U}_x$

Since each U_j in the unitary layer of C acts on a single qubit, it can be written as

$$U_j = \begin{pmatrix} \sqrt{1-p_j} & -\sqrt{p_j} \\ \sqrt{p_j} & \sqrt{1-p_j} \end{pmatrix}$$

where $p_j \in [0,1]$. Technically, there should be a complex relative phase between entries in adjacent matrix rows. However, since we can only perform standard basis measurements, the relative phase does not affect the probability outcomes. Thus, we can safely ignore the phase.

Based on the structure of U_j, we note that if $x_j = 0$, y_j is sampled from $\text{Bern}\left(p_j\right)$ and if $x_j = 1$ from $\text{Bern}\left(1-p_j\right)$. Let

$$q_j := p_j(1-x_j) + (1-p_j)x_j$$

We can write the distribution of y_j more succinctly as

$$y_j \sim \begin{cases} \text{Bern}\left(q_j\right) & \text{if } 1 \leq j \leq n \\ \text{Bern}\left(p_j\right) & \text{otherwise} \end{cases}$$

1.2 Sampling from $\mathcal{U}_x$ Using Randomized AC^0 Circuits

A key step in sampling from a distribution close to $\mathcal{U}_x$ is to sample from $\text{Bern}\left(p\right)$ for any $p \in [0,1]$ up to an error of 2^{-k}, where $k \in \mathbb{N}$ is a given parameter, given $\text{poly}(k)$ uniformly random bits. Another way to think about this problem is to implement a coin flip with bias p given access to polynomially many unbiased coin flips.

If we can solve the above problem, it is straightforward to sample from $\text{Bern}\left(p_j\right)$ up to error 2^{-3n}. To sample from $\text{Bern}\left(q_j\right)$ (up to 2^{-3n} error), we first sample $b \sim \text{Bern}\left(p_j\right)$ and return $b \oplus x_j$. It can be easily verified that the distribution of $b \oplus x_j$ is indeed $\text{Bern}\left(q_j\right)$. Thus, we can sample y' from a distribution $\mathcal{U}'_x$ over $\{0,1\}^m$ such that

$$y'_j \sim \begin{cases} \text{Bern}\left(q'_j\right) & \text{if } 1 \leq j \leq n \\ \text{Bern}\left(p'_j\right) & \text{otherwise} \end{cases}$$

where $|q_j - q'_j| \leq 2^{-3n}$ and $|p_j - p'_j| \leq 2^{-3n}$. We can verify that this distribution pointwise approximates $\mathcal{U}_x$ to the required degree of precision.

Now, we show how to sample from $\text{Bern}\left(p\right)$ up to 2^{-k} precision using an AC^0 with access to k uniformly random bits. Let t be the largest integer such that $t \cdot 2^{-k} \leq p < (t+1) \cdot 2^{-k}$. Given the requirements, it is sufficient to sample from $\text{Bern}\left(t \cdot 2^{-k}\right)$. To do so, we implement a threshold function $\tau_t : \{0,1\}^k \mapsto \{0,1\}$ with the following behavior:

$$\tau_t(z) = \begin{cases} 1 & \text{if } \text{bin}(z) \leq t \\ 0 & \text{otherwise} \end{cases}$$

where $\text{bin}(z)$ is the binary integer represented by the string z. To implement τ_t, we subtract t from $\text{bin}(z)$ (in two's complement representation). We return the sign bit of the resulting string (sign bit equals 1 indicates that the number is negative). Since binary subtraction can be implemented in AC^0, τ_t can also be implemented in AC^0. A simple calculation shows that the distribution of $\tau_t(r)$ is $\text{Bern}\left(t \cdot 2^{-k}\right)$, where r is drawn uniformly at random from $\{0,1\}^k$. This concludes the proof of Theorem 2

2 Simulating Mostly Classical Circuits

Definition 5 (Mostly Classical Circuits [Ros20])**.** A circuit C is mostly classical if C can be written as a layer $L = \bigotimes R_{|\theta\rangle}$, where each $R_{|\theta\rangle} = I - 2\,|\theta\rangle\langle\theta|$ is a reflection operator and $|\theta\rangle = \bigotimes_j |\theta_j\rangle$ is a product state, followed by standard basis measurement of each qubit, and an AC^0 circuit that operates on the outcome of the measurement.

Definition 6 (Niceness Property of Mostly Classical Circuits)**.** A mostly classical circuit is nice if every reflection gate $R_{|\theta\rangle} \in L$ satisfies $|\,\langle x|\theta\rangle\,|^2 \leq 1/4, \forall x \in \{0,1\}^k$, where k is the number of qubits R_θ acts on.

Just like $\mathsf{U} \circ \mathsf{AC}^0$ circuits, we will show that mostly classical circuits can be simulated in AC^0. More specifically, we prove the following theorem:

Theorem 7. *Let C be a mostly classical circuit of size s that acts on n input and a ancillas and achieves $1/2{+}\epsilon$ correlation with a boolean function f. There exists a randomized AC^0 circuit C_R of size poly(s) that achieves $1/2+\epsilon'$ correlation with f, where $|\epsilon - \epsilon'| \leq c \cdot 2^{-n}$ for some universal constant c.*

We first prove Theorem 7 mostly classical nice circuits. We proceed in the same way as before: we analyze the distribution of the strings obtained by measuring after applying the reflection layer, and then show that this distribution can be generated using a randomized AC^0 circuit.

Since the output distribution of $L = \bigotimes R_{|\theta\rangle}$ can be simulated by simulating the output distribution of each $R_{|\theta\rangle}$ individually in parallel, we restrict our attention to a single reflection gate.

Let $x \in \{0,1\}^k$ be the input to the gate and $y \in \{0,1\}^k$ be the string obtained by measuring the output of the gate. Ignoring a relative phase, each $|\theta_j\rangle$ can be written as $|\theta_j\rangle = \sqrt{1-p_j}\,|0\rangle + \sqrt{p_j}\,|1\rangle$ for some $p_j \in [0,1]$. For $p \in [0,1]$ and $b \in \{0,1\}$, we define $\text{Eq}(p,b)$ to be the probability that a random variable $r \sim \text{Bern}\,(p)$ has value b i.e.

$$\text{Eq}(p,b) = (1-p)(1-b) + pb$$

Let $\mathcal{U}_x$ be the distribution of y. By a Lemma 4.5 and Corollary 4.6 from [Ros20], $\mathcal{U}_x$ can be thought of as the following convex combination:

$$\mathcal{U}_x = \left(1 - 4\prod_j \text{Eq}(p_j, x_j)\right)\mathcal{U}_1 + \left(4\prod_j \text{Eq}(p_j, x_j)\right)\mathcal{U}_2$$

where,

$$\mathcal{U}_1(z) = \delta_x(z) = \begin{cases} 1 & \text{if } z = x \\ 0 & \text{otherwise} \end{cases}$$
$$\mathcal{U}_2(z) = \prod_j \mathrm{Eq}(p_j, z_j)$$

The following process can generate this convex combination: we first sample $b \sim \mathrm{Bern}\left(4\prod_j \mathrm{Eq}(p_j, x_j)\right)$. If b is 0, we return x. Otherwise, we sample $y_j \sim \mathrm{Bern}\,(p_j)$ and return $y = (y_j)_j$. A useful subcircuit in sampling from the above distribution is the 2-to-1 multiplexer. The multiplexer receives two input bits i_0 and i_1, a control bit, and produces a single output. If the control bit is 0, the output is i_0; else, the output is i_1. Clearly, a 2-to-1 multiplexer can be implemented in constant depth.

We sample $b \sim \mathrm{Bern}\left(4\prod_j \mathrm{Eq}(p_j, x_j)\right)$ and $y_j \sim \mathrm{Bern}\,(p_j)$. For each index j, we multiplex y_j and x_j, using b as the control.