

Simulations between PRAMs

Theorem 1

We can simulate a p -processor priority CRCW PRAM on a p -processor EREW PRAM with slowdown $\mathcal{O}(\log p)$.

Simulations between PRAMs

Theorem 2

We can simulate a p -processor priority CRCW PRAM on a $p \log p$ -processor common CRCW PRAM with slowdown $\mathcal{O}(1)$.

Simulations between PRAMs

Theorem 3

We can simulate a p -processor priority CRCW PRAM on a p -processor common CRCW PRAM with slowdown $\mathcal{O}\left(\frac{\log p}{\log \log p}\right)$.

Simulations between PRAMs

Theorem 4

We can simulate a p -processor priority CRCW PRAM on a p -processor arbitrary CRCW PRAM with slowdown $\mathcal{O}(\log \log p)$.

Lower Bounds for the CREW PRAM

Ideal PRAM:

- ▶ every processor has unbounded local memory
- ▶ in each step a processor reads a global variable
- ▶ then it does some (**unbounded**) computation on its local memory
- ▶ then it writes a global variable

Lower Bounds for the CREW PRAM

Definition 5

An input index i **affects a memory location M** at time t on some input I if the content of M at time t differs between inputs I and $I(i)$ (i -th bit flipped).

$$L(M, t, I) = \{i \mid i \text{ affects } M \text{ at time } t \text{ on input } I\}$$

Lower Bounds for the CREW PRAM

Definition 6

An input index i **affects a processor** P at time t on some input I if the state of P at time t differs between inputs I and $I(i)$ (i -th bit flipped).

$$K(P, t, I) = \{i \mid i \text{ affects } P \text{ at time } t \text{ on input } I\}$$

Lower Bounds for the CREW PRAM

Lemma 7

If $i \in K(P, t, I)$ with $t > 1$ then either

- ▶ $i \in K(P, t - 1, I)$, or*
- ▶ P reads a global memory location M on input I at time t , and $i \in L(M, t - 1, I)$.*

Lower Bounds for the CREW PRAM

Lemma 8

If $i \in L(M, t, I)$ with $t > 1$ then either

- ▶ A processor writes into M at time t on input I and $i \in K(P, t, I)$, or*
- ▶ No processor writes into M at time t on input I and
 - ▶ either $i \in L(M, t - 1, I)$*
 - ▶ or a processor P writes into M at time t on input $I(i)$.**

Let $k_0 = 0, \ell_0 = 1$ and define

$$k_{t+1} = k_t + \ell_t \text{ and } \ell_{t+1} = 3k_t + 4\ell_t$$

Lemma 9

$|K(P, t, I)| \leq k_t$ and $|L(M, t, I)| \leq \ell_t$ for any $t \geq 0$

base case ($t = 0$):

- ▶ No index can influence the local memory/state of a processor before the first step (hence $|K(P, 0, I)| = k_0 = 0$).
- ▶ Initially every index in the input affects exactly one memory location. Hence $|L(M, 0, I)| = 1 = \ell_0$.

induction step ($t \rightarrow t + 1$):

$K(P, t + 1, I) \subseteq K(P, t, I) \cup L(M, t, I)$, where M is the location read by P in step $t + 1$.

Hence,

$$\begin{aligned} |K(P, t + 1, I)| &\leq |K(P, t, I)| + |L(M, t, I)| \\ &\leq k_t + \ell_t \end{aligned}$$

induction step ($t \rightarrow t + 1$):

For the bound on $|L(M, t + 1, I)|$ we have two cases.

Case 1:

A processor P writes into location M at time $t + 1$ on input I .

Then,

$$\begin{aligned} |L(M, t + 1, I)| &\leq |K(P, t + 1, I)| \\ &\leq k_t + \ell_t \\ &\leq 3k_t + 4\ell_t = \ell_{t+1} \end{aligned}$$

Case 2:

No processor P writes into location M at time $t + 1$ on input I .

An index i affects M at time $t + 1$ iff i affects M at time t or some processor P writes into M at $t + 1$ on $I(i)$.

$$L(M, t + 1, I) \subseteq L(M, t, I) \cup Y(M, t + 1, I)$$

$Y(M, t + 1, I)$ is the set of indices u_j that cause some processor P_{w_j} to write into M at time $t + 1$ on input I .

$Y(M, t + 1, I)$ is the set of indices u_j that cause some processor P_{w_j} to write into M at time $t + 1$ on input I .

Fact:

For all pairs u_s, u_t with $P_{w_s} \neq P_{w_t}$ either $u_s \in K(P_{w_t}, t + 1, I(u_t))$ or $u_t \in K(P_{w_s}, t + 1, I(u_s))$.

Otherwise, P_{w_t} and P_{w_s} would both write into M at the same time on input $I(u_s)(u_t)$.

Let $U = \{u_1, \dots, u_r\}$ denote all indices that cause some processor to write into M .

Let $V = \{(I(u_1), P_{w_1}), \dots\}$.

We set up a bipartite graph between U and V , such that $(u_i, (I(u_j), P_{w_j})) \in E$ if u_i affects P_{w_j} at time $t + 1$ on input $I(u_j)$.

Each vertex $(I(u_j), P_{w_j})$ has degree at most k_{t+1} as this is an upper bound on indices that can influence a processor P_{w_j} .

Hence, $|E| \leq r \cdot k_{t+1}$.

For an index u_j there can be at most k_{t+1} indices u_i with $P_{w_i} = P_{w_j}$.

Hence, there must be at least $\frac{1}{2}r(r - k_{t+1})$ pairs u_i, u_j with $P_{w_i} \neq P_{w_j}$.

Each pair introduces at least one edge.

Hence,

$$|E| \geq \frac{1}{2}r(r - k_{t+1})$$

This gives $r \leq 3k_{t+1} \leq 3k_t + 3\ell_t$

Recall that $L(M, t + 1, i) \subseteq L(M, t, i) \cup Y(M, t + 1, I)$

$$|L(M, t + 1, i)| \leq 3k_t + 4\ell_t$$

$$\begin{pmatrix} k_{t+1} \\ \ell_{t+1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 3 & 4 \end{pmatrix} \begin{pmatrix} k_t \\ \ell_t \end{pmatrix} \quad \begin{pmatrix} k_0 \\ \ell_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Eigenvalues:

$$\lambda_1 = \frac{1}{2}(5 + \sqrt{21}) \text{ and } \lambda_2 = \frac{1}{2}(5 - \sqrt{21})$$

Eigenvectors:

$$v_1 = \begin{pmatrix} 1 \\ -(1 - \lambda_1) \end{pmatrix} \text{ and } v_2 = \begin{pmatrix} 1 \\ -(1 - \lambda_2) \end{pmatrix}$$

$$v_1 = \begin{pmatrix} 1 \\ \frac{3}{2} + \frac{1}{2}\sqrt{21} \end{pmatrix} \text{ and } v_2 = \begin{pmatrix} 1 \\ \frac{3}{2} - \frac{1}{2}\sqrt{21} \end{pmatrix}$$

$$\mathbf{v}_1 = \left(\frac{3}{2} + \frac{1}{2}\sqrt{21} \right) \text{ and } \mathbf{v}_2 = \left(\frac{3}{2} - \frac{1}{2}\sqrt{21} \right)$$

$$\begin{pmatrix} k_0 \\ \ell_0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{21}}(\mathbf{v}_1 - \mathbf{v}_2)$$

$$\begin{pmatrix} k_t \\ \ell_t \end{pmatrix} = \frac{1}{\sqrt{21}}(\lambda_1^t \mathbf{v}_1 - \lambda_2^t \mathbf{v}_2)$$

Solving the recurrence gives

$$k_t = \frac{\lambda_1^t}{\sqrt{21}} - \frac{\lambda_2^t}{\sqrt{21}}$$

$$\ell_t = \frac{3 + \sqrt{21}}{2\sqrt{21}} \lambda_1^t + \frac{-3 + \sqrt{21}}{2\sqrt{21}} \lambda_2^t$$

with $\lambda_1 = \frac{1}{2}(5 + \sqrt{21})$ and $\lambda_2 = \frac{1}{2}(5 - \sqrt{21})$.

Theorem 10

The following problems require logarithmic time on a CREW PRAM.

- ▶ *Sorting a sequence of x_1, \dots, x_n with $x_i \in \{0, 1\}$*
- ▶ *Computing the maximum of n inputs*
- ▶ *Computing the sum $x_1 + \dots + x_n$ with $x_i \in \{0, 1\}$*

A Lower Bound for the EREW PRAM

Definition 11 (Zero Counting Problem)

Given a monotone binary sequence x_1, x_2, \dots, x_n determine the index i such that $x_i = 0$ and $x_{i+1} = 1$.

We show that this problem requires $\Omega(\log n - \log p)$ steps on a p -processor EREW PRAM.

Let I_i be the input with i zeros folled by $n - i$ ones.

Index i affects processor P at time t if the state in step t is differs between I_{i-1} and I_i .

Index i affects location M at time t if the content of M after step t differs between inputs I_{i-1} and I_i .

Lemma 12

If $i \in K(P, t)$ then either

- ▶ *$i \in K(P, t - 1)$, or*
- ▶ *P reads some location M on input I_i (and, hence, also on I_{i-1}) at step t and $i \in L(M, t - 1)$*

Lemma 13

If $i \in L(M, t)$ then either

- ▶ $i \in L(M, t - 1)$, or
- ▶ Some processor P writes M at step t on input I_i and $i \in K(P, t)$.
- ▶ Some processor P writes M at step t on input I_{i-1} and $i \in K(P, t)$.

Define

$$C(t) = \sum_P |K(P, t)| + \sum_M \max\{0, |L(M, t)| - 1\}$$

$$C(T) \geq n, C(0) = 0$$

Claim:

$$C(t) \leq 6C(t-1) + 3|P|$$

This gives $C(T) \leq \frac{6^T - 1}{5} 3|P|$ and hence $T = \Omega(\log n - \log |P|)$.

For an index i to **newly** appear in $L(M, t)$ some processor must write into M on either input I_i or I_{i-1} .

Hence, any index in $K(P, t)$ can at most generate two **new** indices in $L(M, t)$.

This means that the number of new indices in any set $L(M, t)$ (over all M) is at most

$$2 \sum_P |K(P, t)|$$

Hence,

$$\sum_M |L(M, t)| \leq \sum_M |L(M, t - 1)| + 2 \sum_P |K(P, t)|$$

We can assume wlog. that $L(M, t - 1) \subseteq L(M, t)$. Then

$$\sum_M \max\{0, |L(M, t)| - 1\} \leq \sum_M \max\{0, |L(M, t - 1)| - 1\} + 2 \sum_P |K(P, t)|$$

For an index i to **newly** appear in $K(P, t)$, P must read a memory location M with $i \in L(M, t)$ on input I_i (and also on input I_{i-1}).

Since we are in the EREW model at most one processor can do so in every step.

Let $J(i, t)$ be memory locations read in step t on input I_i , and let $J_t = \bigcup_i J(i, t)$.

$$\sum_P |K(P, t)| \leq \sum_P |K(P, t-1)| + \sum_{M \in J_t} |L(M, t-1)|$$

Over all inputs I_i a processor can read at most $|K(P, t-1)| + 1$ different memory locations (why?).

Hence,

$$\begin{aligned} \sum_P |K(P, t)| &\leq \sum_P |K(P, t - 1)| + \sum_{M \in J_t} |L(M, t - 1)| \\ &\leq \sum_P |K(P, t - 1)| + \sum_{M \in J_t} (|L(M, t - 1)| - 1) + J_t \\ &\leq 2 \sum_P |K(P, t - 1)| + \sum_{M \in J_t} (|L(M, t - 1)| - 1) + |P| \\ &\leq 2 \sum_P |K(P, t - 1)| + \sum_M \max\{0, |L(M, t - 1)| - 1\} + |P| \end{aligned}$$

Recall

$$\sum_M \max\{0, |L(M, t)| - 1\} \leq \sum_M \max\{0, |L(M, t - 1)| - 1\} + 2 \sum_P |K(P, t)|$$

This gives

$$\begin{aligned} & \sum_P K(P, t) + \sum_M \max\{0, |L(M, t)| - 1\} \\ & \leq 4 \sum_M \max\{0, |L(M, t - 1)| - 1\} + 6 \sum_P |K(P, t - 1)| + 3|P| \end{aligned}$$

Hence,

$$C(t) \leq 6C(t - 1) + 3|P|$$

Lower Bounds for CRCW PRAMS

Theorem 14

Let $f : \{0, 1\}^n \rightarrow \{0, 1\}$ be an arbitrary Boolean function. f can be computed in $\mathcal{O}(1)$ time on a common CRCW PRAM with $\leq n2^n$ processors.

Can we obtain non-constant lower bounds if we restrict the number of processors to be polynomial?

Boolean Circuits

- ▶ nodes are either **AND**, **OR**, or **NOT** gates or are special **INPUT/OUTPUT** nodes
- ▶ **AND** and **OR** gates have unbounded fan-in (indegree) and unbounded fan-out (outdegree)
- ▶ **NOT** gates have unbounded fan-out
- ▶ **INPUT** nodes have indegree zero; **OUTPUT** nodes have outdegree zero
- ▶ **size** is the number of edges
- ▶ **depth** is the longest path from an input to an output

Theorem 15

Let $f : \{0, 1\}^n \rightarrow \{0, 1\}^m$ be a function with n inputs and $m \leq n$ outputs, and circuit C computes f with depth $D(n)$ and size $S(n)$. Then f can be computed by a common CRCW PRAM in $\mathcal{O}(D(n))$ time using $S(n)$ processors.

Given a family $\{C_n\}$ of circuits we may not be able to compute the corresponding family of functions on a CRCW PRAM.

Definition 16

A family $\{C_n\}$ of circuits is **logspace uniform** if there exists a deterministic Turing machine M s.t

- ▶ M runs in logarithmic space.
- ▶ For all n , M outputs C_n on input 1^n .