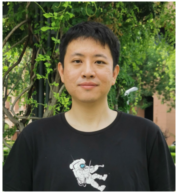
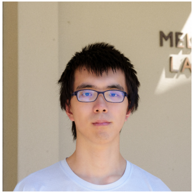
BPL $\subseteq$ L-AC^1



Kuan Cheng
(Peking University)



Yichuan Wang
(Tsinghua University)

2024-07-23

BPL $\subseteq$ L-AC^1

Definition (BPL, bounded-error probabilistic logspace)

M runs in space $O(\log|x|)$;

$$x \text{ (input)} \mapsto \begin{cases} \text{Yes} & \text{if } \Pr_r[M(x,r)=1] \geq 2/3 \\ \text{No} & \text{if } \Pr_r[M(x,r)=1] \leq 1/3 \\ \bot & \text{otherwise} \end{cases} \qquad (r\text{: random bits})$$

(Important) M **only has read-once access to random tape** r**;**
Halt on any random tape $\implies |r| \leq \text{poly}(|x|)$.

BPL $\subseteq$ L-AC^1

Definition (BPL, bounded-error p

Big Open Problem: L $\stackrel{?}{=}$ BPL

M runs in space $O(\log|x|)$;

$$x \text{ (input)} \mapsto \begin{cases} \text{Yes} & \text{if } \Pr_r[M(x,r)=1] \geq 2/3 \\ \text{No} & \text{if } \Pr_r[M(x,r)=1] \leq 1/3 \\ \perp & \text{otherwise} \end{cases} \qquad (r\text{: random bits})$$

(Important) M **only has read-once access to random tape** r**;**
Halt on any random tape $\implies |r| \leq \text{poly}(|x|)$.

BPL $\subseteq$ L-AC^1

Definition (BPL, bounded-error p

M runs in space $O(\log|x|)$;

Big Open Problem: L $\stackrel{?}{=}$ BPL

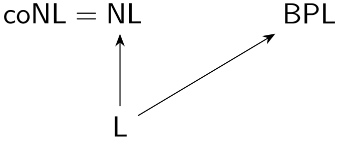
$$x \text{ (input)} \mapsto \begin{cases} \text{Yes} & \text{if } \Pr_r[M(x,r)=1] \geq 2/3 \\ \text{No} & \text{if } \Pr_r[M(x,r)=1] \leq 1/3 \\ \perp & \text{otherwise} \end{cases} \qquad (r\text{: random bits})$$

(Important) M **only has read-once access to random tape** r**;**
Halt on any random tape $\implies |r| \leq \text{poly}(|x|)$.

Definition (L-AC^1)

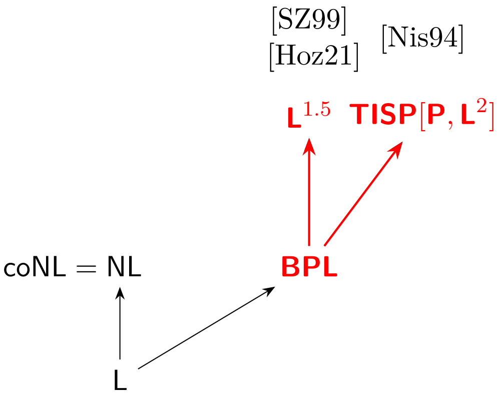
- **Unbounded fan-in AND/OR** gates and **free NOT** gates.
- poly(n)-size, $O(\log n)$-depth.
- Logspace uniform.

Relations Between Complexity Classes ($\mathsf{A} \to \mathsf{B}$ means $\mathsf{A} \subseteq \mathsf{B}$)



- **Wish to prove:** $\mathsf{L} = \mathsf{BPL}$.
- **Don't even know:** does $\mathsf{L} = \mathsf{NL}$ imply $\mathsf{L} = \mathsf{BPL}$?

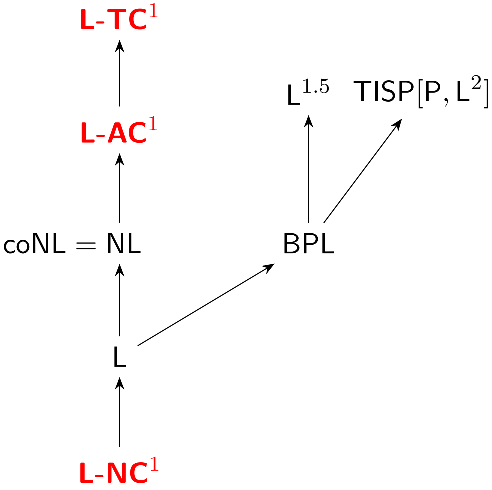
Relations Between Complexity Classes ($\mathsf{A} \to \mathsf{B}$ means $\mathsf{A} \subseteq \mathsf{B}$)



- **Progress towards $\mathsf{L} = \mathsf{BPL}$:**

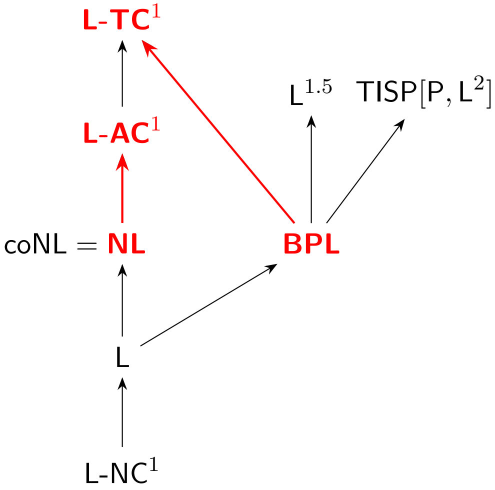
  [Sav70] [BCP83] [Nis92] [Nis94]
  [INW94] [NZ96] [SZ99] [KvM02]
  [CH20] [Hoz21] [Pyn23]
  ...
  Black-box PRG
  Richardson Iteration
  ...

Relations Between Complexity Classes ($\mathsf{A} \to \mathsf{B}$ means $\mathsf{A} \subseteq \mathsf{B}$)



- Hierachy of shallow circuit classes:
  $\mathsf{L\text{-}NC}^1$: logspace uniform NC^1;
  $\mathsf{L\text{-}AC}^1$: logspace uniform AC^1;
  $\mathsf{L\text{-}TC}^1$: logspace uniform TC^1.

Relations Between Complexity Classes ($\mathsf{A} \to \mathsf{B}$ means $\mathsf{A} \subseteq \mathsf{B}$)



- Hierachy of shallow circuit classes:
  L-NC^1: logspace uniform NC^1;
  L-AC^1: logspace uniform AC^1;
  L-TC^1: logspace uniform TC^1.
- $\mathsf{NL} \subseteq \mathsf{L\text{-}AC}^1$, $\mathsf{BPL} \subseteq \mathsf{L\text{-}TC}^1$.
  We will see why.

Preliminaries: Matrix Norm

Definition (L_1-norm)

- For vector, $\left\|(x_1, x_2, \cdots, x_n)^\top\right\| := |x_1| + |x_2| + \cdots + |x_n|$.
- For matrix, $\|\mathbf{A}\| := \max_{\mathbf{x}} \frac{\|\mathbf{A}\mathbf{x}\|}{\|\mathbf{x}\|} = \max_i \left\|(i\text{-th column of } \mathbf{A})\right\|$.

Proposition

- $\|\mathbf{A} + \mathbf{B}\| \le \|\mathbf{A}\| + \|\mathbf{B}\|$
- $\|\mathbf{A}\mathbf{B}\| \le \|\mathbf{A}\| \cdot \|\mathbf{B}\|$

Definition (Stochastic vector/matrix)

A vector/matrix $\mathbf{A}$ is stochastic if:

1. all entries of $\mathbf{A}$ are nonnegative; and
2. sum of entries in each column is $= 1$.

Preliminaries: Matrix Norm

Definition (L_1-norm)

- For vector, $\left\|(x_1, x_2, \cdots, x_n)^\top\right\| := |x_1| + |x_2| + \cdots + |x_n|$.
- For matrix, $\|\mathbf{A}\| := \max_{\mathbf{x}} \frac{\|\mathbf{A}\mathbf{x}\|}{\|\mathbf{x}\|} = \max_i \left\|(i\text{-th column of } \mathbf{A})\right\|$.

Proposition

- $\|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|$
- $\|\mathbf{A}\mathbf{B}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{B}\|$

Definition (Stochastic vector/matrix)

A vector/matrix $\mathbf{A}$ is stochastic if:

1. all entries of $\mathbf{A}$ are nonnegative; and
2. sum of entries in each column is $= 1$.

Stochastic Vector: represents a probability distribution.
Stochastic Matrix: maps stochastic vectors to stochastic vectors.

Preliminaries: Matrix Norm

Definition (L_1-norm)

- For vector, $\left\|(x_1, x_2, \cdots, x_n)^\top\right\| := |x_1| + |x_2| + \cdots + |x_n|$.
- For matrix, $\|\mathbf{A}\| := \max_{\mathbf{x}} \frac{\|\mathbf{Ax}\|}{\|\mathbf{x}\|} = \max_i \left\|(i\text{-th column of } \mathbf{A})\right\|$.

Proposition

- $\|\mathbf{A} + \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|$
- $\|\mathbf{AB}\| \leq \|\mathbf{A}\| \cdot \|\mathbf{B}\|$

Stochastic Vector: represents a probability distribution.
Stochastic Matrix: maps stochastic vectors to stochastic vectors.

Definition (Substochastic vector/matrix)

A vector/matrix $\mathbf{A}$ is substochastic if:

1. all entries of $\mathbf{A}$ are nonnegative; and
2. sum of entries in each column is ≤ 1. (i.e., $\|\mathbf{A}\| \leq 1$)

Matrix Multiplication View of BPL

- w: number of internal states of the logspace TM (with a fixed input).
  $w \leq \text{poly}(n)$

Matrix Multiplication View of **BPL**

- w: number of internal states of the logspace TM (with a fixed input). $w \le \text{poly}(n)$
- Define transition matrix $\mathbf{A} \in \mathbb{R}^{w \times w}$: for $i, j \in [w]$,

$$\mathbf{A}_{i,j} := \text{At state } i \text{, after one step, the probability to reach } j.$$

Matrix Multiplication View of **BPL**

- w: number of internal states of the logspace TM (with a fixed input). $w \leq \text{poly}(n)$
- Define transition matrix $\mathbf{A} \in \mathbb{R}^{w \times w}$: for $i, j \in [w]$,

$$\mathbf{A}_{i,j} := \text{At state } i\text{, after one step, the probability to reach } j.$$

- For any t,

$$\left(\mathbf{A}^t\right)_{i,j} = \text{At state } i\text{, after } t \text{ steps, the probability to reach } j.$$

Matrix Multiplication View of BPL

- w: number of internal states of the logspace TM (with a fixed input). $w \le \text{poly}(n)$
- Define transition matrix $\mathbf{A} \in \mathbb{R}^{w \times w}$: for $i, j \in [w]$,

$$\mathbf{A}_{i,j} := \text{At state } i\text{, after one step, the probability to reach } j.$$

- For any t,

$$\left(\mathbf{A}^t\right)_{i,j} = \text{At state } i\text{, after } t \text{ steps, the probability to reach } j.$$

- Estimating powers of $\mathbf{A}$ is BPL-complete!

Matrix Multiplication View of BPL

BPL-complete problem:
Given substochastic $\mathbf{A} \in \mathbb{R}^{n \times n}$,
estimate $\mathbf{A}^n$ with error $\leq 1/n$.

Matrix Multiplication View of BPL

BPL-complete problem:
Given substochastic $\mathbf{A} \in \mathbb{R}^{n \times n}$, estimate $\mathbf{A}^n$ with error $\leq 1/n$.

NL-complete problem:
Given $\mathbf{A} \in \{0,1\}^{n \times n}$, compute $\mathbf{A}^n$ under $(\mathrm{AND}, \mathrm{OR})$-matrix multiplication.

Matrix Multiplication View of BPL

BPL-complete problem:
Given substochastic $\mathbf{A} \in \mathbb{R}^{n \times n}$, estimate $\mathbf{A}^n$ with error $\leq 1/n$.

$$\mathbf{C}_{i,j} = \sum_{k \in [n]} (\mathbf{A}_{i,k} \cdot \mathbf{B}_{k,j})$$

NL-complete problem:
Given $\mathbf{A} \in \{0,1\}^{n \times n}$, compute $\mathbf{A}^n$ under $(\mathrm{AND}, \mathrm{OR})$-matrix multiplication.

$$\mathbf{C}_{i,j} = \bigvee_{k \in [n]} (\mathbf{A}_{i,k} \wedge \mathbf{B}_{k,j})$$

Matrix Multiplication View of **BPL**

BPL-complete problem:
Given substochastic $\mathbf{A} \in \mathbb{R}^{n \times n}$,
estimate $\mathbf{A}^n$ with error $\leq 1/n$.

$\mathbf{C}_{i,j} = \sum_{k \in [n]} (\mathbf{A}_{i,k} \cdot \mathbf{B}_{k,j})$

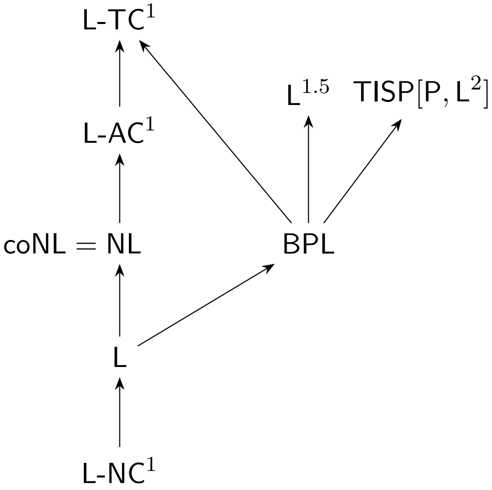
$1/\operatorname{poly}(n)$-error standard MM
can be computed in $\mathsf{L\text{-}TC}^0$
$\implies$ **BPL** $\subseteq$ $\mathbf{L\text{-}TC}^1$

NL-complete problem:
Given $\mathbf{A} \in \{0,1\}^{n \times n}$,
compute $\mathbf{A}^n$ under
$(\mathrm{AND}, \mathrm{OR})$-matrix multiplication.

$\mathbf{C}_{i,j} = \bigvee_{k \in [n]} (\mathbf{A}_{i,k} \wedge \mathbf{B}_{k,j})$

$(\mathrm{AND}, \mathrm{OR})$-MM
can be computed in $\mathsf{L\text{-}AC}^0$
$\implies$ **NL** $\subseteq$ $\mathbf{L\text{-}AC}^1$

Relations Between Complexity Classes ($\mathsf{A} \rightarrow \mathsf{B}$ means $\mathsf{A} \subseteq \mathsf{B}$)

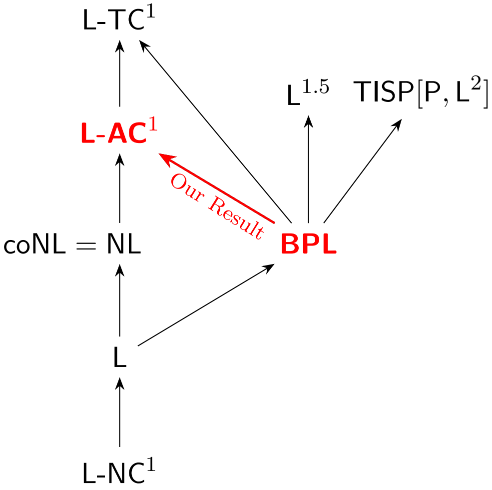


- Known Relations:

$$\mathsf{L\text{-}NC}^1 \subseteq \mathsf{L} \subseteq \mathsf{NL} \subseteq \mathsf{L\text{-}AC}^1 \subseteq \mathsf{L\text{-}TC}^1$$

$$\mathsf{L} \subseteq \mathsf{BPL} \subseteq \mathsf{L\text{-}TC}^1$$

Relations Between Complexity Classes (A $\to$ B means A $\subseteq$ B)



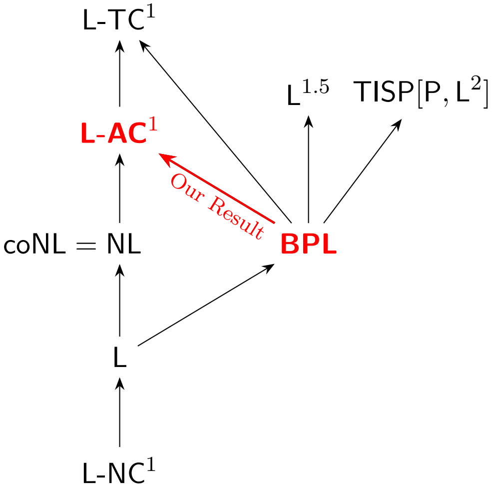
- Known Relations:

  $$\mathsf{L\text{-}NC}^1 \subseteq \mathsf{L} \subseteq \mathsf{NL} \subseteq \mathsf{L\text{-}AC}^1 \subseteq \mathsf{L\text{-}TC}^1$$

  $$\mathsf{L} \subseteq \mathsf{BPL} \subseteq \mathsf{L\text{-}TC}^1$$

- **Our result: $\mathsf{BPL} \subseteq \mathsf{L\text{-}AC}^1$.**

Relations Between Complexity Classes (A → B means A ⊆ B)



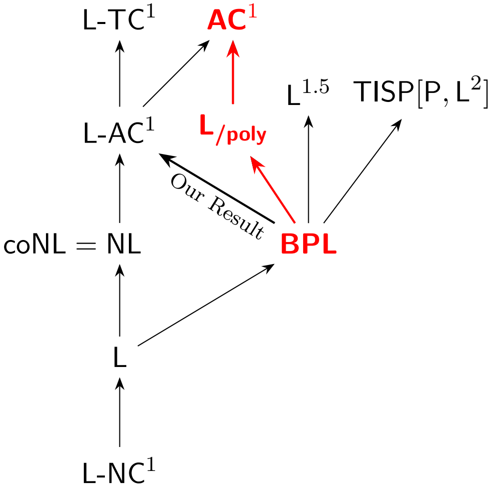
- Known Relations:

$$\text{L-NC}^1 \subseteq \text{L} \subseteq \text{NL} \subseteq \text{L-AC}^1 \subseteq \text{L-TC}^1$$

$$\text{L} \subseteq \text{BPL} \subseteq \text{L-TC}^1$$

- **Our result: BPL ⊆ L-AC^1.**
- Partial progress towards BPL ⊆ NL.

Relations Between Complexity Classes ($A \to B$ means $A \subseteq B$)



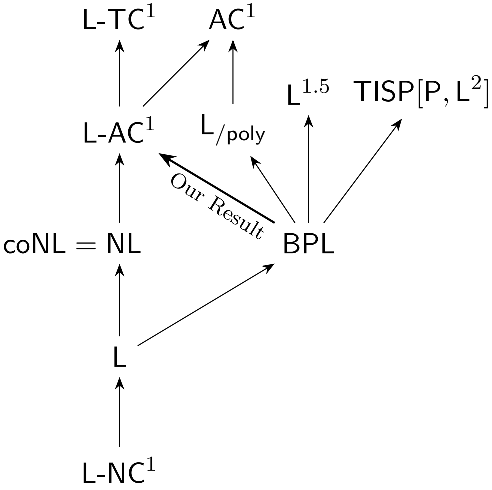
- Known Relations:

$$\text{L-NC}^1 \subseteq \text{L} \subseteq \text{NL} \subseteq \text{L-AC}^1 \subseteq \text{L-TC}^1$$

$$\text{L} \subseteq \text{BPL} \subseteq \text{L-TC}^1$$

- **Our result: BPL $\subseteq$ L-AC^1.**
- Partial progress towards BPL $\subseteq$ NL.
- Remark: BPL $\subseteq$ AC^1 (non-uniform AC^1) is trivial.

Relations Between Complexity Classes ($\mathsf{A} \to \mathsf{B}$ means $\mathsf{A} \subseteq \mathsf{B}$)



- Known Relations:

$$\mathsf{L\text{-}NC}^1 \subseteq \mathsf{L} \subseteq \mathsf{NL} \subseteq \mathsf{L\text{-}AC}^1 \subseteq \mathsf{L\text{-}TC}^1$$

$$\mathsf{L} \subseteq \mathsf{BPL} \subseteq \mathsf{L\text{-}TC}^1$$

- **Our result: $\mathsf{BPL} \subseteq \mathsf{L\text{-}AC}^1$.**
- Partial progress towards $\mathsf{BPL} \subseteq \mathsf{NL}$.
- Remark: $\mathsf{BPL} \subseteq \mathsf{AC}^1$ (non-uniform AC^1) is trivial.

Overview of our Algorithm $\mathsf{BPL} \subseteq \mathsf{L\text{-}AC}^1$

- **Part1.** Non-trivial L-AC algorithms
  - Approximate counting in L-AC.
  - Low-precision matrix operations in L-AC.
    (e.g. Multiply $n \times n$ matrices with $1/3$ error in $\mathsf{L\text{-}AC}^0$)
- **Part2.** Use low-precision operations to compute matrix powering
  - Recall: estimate $\mathbf{A} \mapsto \mathbf{A}^n$ with $1/n$-error is BPL-complete.
  - Numerical analysis techniques! (Popular in recent years)

Overview of our Algorithm $\mathsf{BPL} \subseteq \mathsf{L\text{-}AC}^1$

- **Part1. Non-trivial L-AC algorithms**
  - **Approximate counting in L-AC.**
  - **Low-precision matrix operations in L-AC.**
    (e.g. Multiply $n \times n$ matrices with $1/3$ error in $\mathsf{L\text{-}AC}^0$)
- **Part2.** Use low-precision operations to compute matrix powering
  - Recall: estimate $\mathbf{A} \mapsto \mathbf{A}^n$ with $1/n$-error is BPL-complete.
  - Numerical analysis techniques! (Popular in recent years)

A Nontrivial L-AC Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0**. [Ajtai'90]
  (Reminder: exact majority cannot be computed in AC^0.)

$$(x_1, x_2, \cdots, x_n) \mapsto \begin{cases} \text{YES} & \text{if input contains} \geq \frac{2n}{3} \text{ 1's} \\ \text{NO} & \text{if input contains} \leq \frac{n}{3} \text{ 1's} \\ \perp & \text{otherwise} \end{cases}$$

A Nontrivial **L-AC** Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0**. [Ajtai'90]
  (Reminder: exact majority cannot be computed in AC^0.)

$$(x_1, x_2, \cdots, x_n) \mapsto \begin{cases} \text{YES} & \text{if input contains} \geq \frac{2n}{3} \text{ 1's} \\ \text{NO} & \text{if input contains} \leq \frac{n}{3} \text{ 1's} \\ \perp & \text{otherwise} \end{cases}$$

- Later generalized in [Viola'07] [Viola'11] [Cook'20] etc. for different purposes.

A Nontrivial $\mathsf{L\text{-}AC}$ Algorithm

- A non-trivial algorithm: **Approximate counting in $\mathsf{L\text{-}AC}^0$**. [Ajtai'90]
(Reminder: exact majority cannot be computed in AC^0.)

$$(x_1, x_2, \cdots, x_n) \mapsto \begin{cases} \text{YES} & \text{if input contains} \geq \frac{2n}{3} \text{ 1's} \\ \text{NO} & \text{if input contains} \leq \frac{n}{3} \text{ 1's} \\ \perp & \text{otherwise} \end{cases}$$

- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$	$\mathrm{poly}(n)$

(If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S, (1+\varepsilon)S]$.)

A Nontrivial L-AC Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0**. [Ajtai'90]
(Reminder: exact majority cannot be computed in AC^0.)

$$(x_1, x_2, \cdots, x_n) \mapsto \begin{cases} \text{YES} & \text{if input contains} \geq \frac{2n}{3} \text{ 1's} \\ \text{NO} & \text{if input contains} \leq \frac{n}{3} \text{ 1's} \\ \perp & \text{otherwise} \end{cases}$$

- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$	$\text{poly}(n)$

(If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S, (1+\varepsilon)S]$.)

Proof Sketch: Use *Samplers/Universal Hash Functions* to reduce to the standard approximate counting.

A Nontrivial L-AC Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0**. [Ajtai'90]
  (Reminder: exact majority cannot be computed in AC^0.)

$$(x_1, x_2, \cdots, x_n) \mapsto \begin{cases} \text{YES} & \text{if input contains} \geq \frac{2n}{3} \text{ 1's} \\ \text{NO} & \text{if input contains} \leq \frac{n}{3} \text{ 1's} \\ \perp & \text{otherwise} \end{cases}$$

- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$	$\text{poly}(n)$

  (If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S, (1+\varepsilon)S]$.)

- **Application:** (add poly(n) | multiply two) positive real numbers with small multiplicative error. (ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$-depth)

A Nontrivial **L-AC** Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0** [Ajtai'90]
  (I

Example:

- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n}+1\right)$	$\mathrm{poly}(n)$

  (If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S, (1+\varepsilon)S]$.)

- **Application:** (add poly(n) | multiply two) positive real numbers with small multiplicative error. (ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n}+1\right)$-depth)

A Nontrivial **L-AC** Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0** [Ajtai'90]
  (I

Example:

$a = 0.12$

$b = 0.39$

- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$	$\mathrm{poly}(n)$

(If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S, (1+\varepsilon)S]$.)

- **Application:** (add poly(n) | multiply two) positive real numbers with small multiplicative error. (ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$-depth)

A Nontrivial L-AC Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0** [Ajtai'90]
  (I

Example:

$a = 0.12 \longrightarrow 1000000000$

$b = 0.39 \longrightarrow 1111000000$

- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$	$\text{poly}(n)$

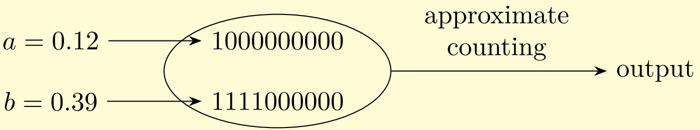
(If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S, (1+\varepsilon)S]$.)

- **Application:** (add poly(n) | multiply two) positive real numbers with small multiplicative error. (ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$-depth)

A Nontrivial **L-AC** Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0** [Ajtai'90]

Example:



- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n}+1\right)$	$\mathrm{poly}(n)$

(If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S,(1+\varepsilon)S]$.)

- **Application:** (add $\mathrm{poly}(n)$ | multiply two) positive real numbers with small multiplicative error. (ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n}+1\right)$-depth)

A Nontrivial $\mathsf{L\text{-}AC}$ Algorithm

- A non-trivial algorithm: **Approximate counting in $\mathsf{L\text{-}AC}^0$**. [Ajtai'90]
(Reminder: exact majority cannot be computed in AC^0.)

$$(x_1, x_2, \cdots, x_n) \mapsto \begin{cases} \text{YES} & \text{if input contains} \geq \frac{2n}{3} \text{ 1's} \\ \text{NO} & \text{if input contains} \leq \frac{n}{3} \text{ 1's} \\ \perp & \text{otherwise} \end{cases}$$

- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$	$\mathrm{poly}(n)$

(If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S, (1+\varepsilon)S]$.)

- **Application:** (add $\mathrm{poly}(n)$ | multiply two) positive real numbers with small multiplicative error. (ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$-depth)
- **Further Application:** Matrix operations with ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$-depth!

A Nontrivial L-AC Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0.** [Ajtai'90]
  (Reminder: exact majority cannot be computed in AC^0.)

$$(x_1, x_2, \ldots, x_n) \mapsto \begin{cases} \text{YES} & \text{if input contains} \geq \frac{2n}{3} \text{ 1's} \\ \text{NO} & \text{if input contains} \leq \frac{n}{3} \text{ 1's} \end{cases}$$

Just compute each entry respectively.

- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$	$\text{poly}(n)$

(If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S, (1+\varepsilon)S]$.)

- **Application:** (add poly(n) | multiply two) positive real numbers with small multiplicative error. (ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$-depth)
- **Further Application:** Matrix operations with ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$-depth!

A Nontrivial L-AC Algorithm

- A non-trivial algorithm: **Approximate counting in L-AC^0**. [Ajtai'90]
  (Reminder: exact majority cannot be computed in AC^0.)

$$(x_1, x_2, \cdots, x_n) \mapsto \begin{cases} \text{YES} & \text{if input contains} \geq \frac{2n}{3} \text{ 1's} \\ \text{NO} & \text{if input contains} \leq \frac{n}{3} \text{ 1's} \\ \perp & \text{otherwise} \end{cases}$$

- **Our Generalization:** estimate the number of 1's with **small multiplicative error**

error	depth	size
ε	$O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$	$\text{poly}(n)$

  (If there are S 1's, we need to output an estimate in $[(1-\varepsilon)S, (1+\varepsilon)S]$.)

- **Application:** (add poly(n) | multiply two) positive real numbers with small multiplicative error. (ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$-depth)
- **Further Application:** Matrix operations with ε-error in $O\left(\frac{\log(1/\varepsilon)}{\log\log n} + 1\right)$-depth!

Overview of our Algorithm BPL $\subseteq$ L-AC^1

- **Part1.** Non-trivial L-AC algorithms
  - Approximate counting in L-AC.
  - Low-precision matrix operations in L-AC.
    (e.g. Multiply $n \times n$ matrices with $1/3$ error in L-AC^0)
- **Part2. Use low-precision operations to compute matrix powering**
  - **Recall: estimate $\mathbf{A} \mapsto \mathbf{A}^n$ with $1/n$-error is BPL-complete.**
  - **Numerical analysis techniques! (Popular in recent years)**

Overview of our Algorithm $\mathsf{BPL} \subseteq \mathsf{L\text{-}AC}^1$

- **Part1.** Non-trivial L-AC algorithms
  - Approximate counting in L-AC.
  - Low-precision matrix operations in L-AC.
    (e.g. Multiply $n \times n$ matrices with $1/3$ err

A General Framework:
1. Compute some weak approximations.
2. "Boost" to high precision.

Example: Richardson Iteration [AKM+20] [CDRST21] [PV21] [CDST22] [PP22] [CHLTW23]$\cdots$

- **Part2. Use low-precision operations to compute matrix powering**
  - **Recall: estimate $\mathbf{A} \mapsto \mathbf{A}^n$ with $1/n$-error is BPL-complete.**
  - **Numerical analysis techniques! (Popular in recent years)**

Framework

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Framework

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k,t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k,t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \leq i \leq \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

Framework

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \leq i \leq \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

- **Proof Sketch of Lemma**
- **Lemma** $\implies$ $\textsf{BPL} \subseteq \textsf{L-AC}^1$

Boost weaker approximation to stronger approximation

- **Example:** (For simplicity, assume matrix multiplication commutes)
- **Weak Approximations:**
  $\left\|\widetilde{\mathbf{A}^{16}}-\mathbf{A}^{16}\right\|\le\varepsilon_4$, $\left\|\widetilde{\mathbf{A}^{8}}-\mathbf{A}^{8}\right\|\le\varepsilon_3$, $\left\|\widetilde{\mathbf{A}^{4}}-\mathbf{A}^{4}\right\|\le\varepsilon_2$, $\left\|\widetilde{\mathbf{A}^{2}}-\mathbf{A}^{2}\right\|\le\varepsilon_1$
- **Approximate $\mathbf{A}^{32}$:**

Boost weaker approximation to stronger approximation

- **Example:** (For simplicity, assume matrix multiplication commutes)

- **Weak Approximations:**
  $\left\|\widetilde{\mathbf{A}^{16}} - \mathbf{A}^{16}\right\| \le \varepsilon_4$, $\left\|\widetilde{\mathbf{A}^{8}} - \mathbf{A}^{8}\right\| \le \varepsilon_3$, $\left\|\widetilde{\mathbf{A}^{4}} - \mathbf{A}^{4}\right\| \le \varepsilon_2$, $\left\|\widetilde{\mathbf{A}^{2}} - \mathbf{A}^{2}\right\| \le \varepsilon_1$

$$\left\|\widetilde{\mathbf{M}} - \mathbf{M}\right\| \le \varepsilon \implies \left\|\left(\widetilde{\mathbf{M}} - \mathbf{M}\right)^2\right\| \le \varepsilon^2$$

$$\implies \mathbf{M}^2 \overset{\varepsilon^2}{\approx} 2\mathbf{M}\widetilde{\mathbf{M}} - \widetilde{\mathbf{M}}^2$$

- **Approximate $\mathbf{A}^{32}$:**

Boost weaker approximation to stronger approximation

- **Example:** (For simplicity, assume matrix multiplication commutes)
- **Weak Approximations:**
  $\left\|\widetilde{\mathbf{A}^{16}} - \mathbf{A}^{16}\right\| \leq \varepsilon_4, \left\|\widetilde{\mathbf{A}^{8}} - \mathbf{A}^{8}\right\| \leq \varepsilon_3, \left\|\widetilde{\mathbf{A}^{4}} - \mathbf{A}^{4}\right\| \leq \varepsilon_2, \left\|\widetilde{\mathbf{A}^{2}} - \mathbf{A}^{2}\right\| \leq \varepsilon_1$

$$\left\|\widetilde{\mathbf{M}} - \mathbf{M}\right\| \leq \varepsilon \implies \left\|\left(\widetilde{\mathbf{M}} - \mathbf{M}\right)^2\right\| \leq \varepsilon^2$$

$$\implies \mathbf{M}^2 \overset{\varepsilon^2}{\approx} 2\mathbf{M}\widetilde{\mathbf{M}} - \widetilde{\mathbf{M}}^2$$

- **Approximate $\mathbf{A}^{32}$:**

$\mathbf{A}^{32}$

Boost weaker approximation to stronger approximation

- **Example:** (For simplicity, assume matrix multiplication commutes)

- **Weak Approximations:**
$\left\|\widetilde{\mathbf{A}^{16}} - \mathbf{A}^{16}\right\| \leq \varepsilon_4$, $\left\|\widetilde{\mathbf{A}^{8}} - \mathbf{A}^{8}\right\| \leq \varepsilon_3$, $\left\|\widetilde{\mathbf{A}^{4}} - \mathbf{A}^{4}\right\| \leq \varepsilon_2$, $\left\|\widetilde{\mathbf{A}^{2}} - \mathbf{A}^{2}\right\| \leq \varepsilon_1$

$$\left\|\widetilde{\mathbf{M}} - \mathbf{M}\right\| \leq \varepsilon \implies \left\|\left(\widetilde{\mathbf{M}} - \mathbf{M}\right)^2\right\| \leq \varepsilon^2$$

$$\implies \mathbf{M}^2 \overset{\varepsilon^2}{\approx} 2\mathbf{M}\widetilde{\mathbf{M}} - \widetilde{\mathbf{M}}^2$$

- **Approximate $\mathbf{A}^{32}$:**

$$\mathbf{A}^{32} \overset{\varepsilon_4^2}{\approx} 2\mathbf{A}^{16}\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$

Boost weaker approximation to stronger approximation

- **Example:** (For simplicity, assume matrix multiplication commutes)

- **Weak Approximations:**
$\left\|\widetilde{\mathbf{A}^{16}} - \mathbf{A}^{16}\right\| \leq \varepsilon_4, \left\|\widetilde{\mathbf{A}^{8}} - \mathbf{A}^{8}\right\| \leq \varepsilon_3, \left\|\widetilde{\mathbf{A}^{4}} - \mathbf{A}^{4}\right\| \leq \varepsilon_2, \left\|\widetilde{\mathbf{A}^{2}} - \mathbf{A}^{2}\right\| \leq \varepsilon_1$

$$\left\|\widetilde{\mathbf{M}} - \mathbf{M}\right\| \leq \varepsilon \implies \left\|\left(\widetilde{\mathbf{M}} - \mathbf{M}\right)^2\right\| \leq \varepsilon^2$$

$$\implies \mathbf{M}^2 \overset{\varepsilon^2}{\approx} 2\mathbf{M}\widetilde{\mathbf{M}} - \widetilde{\mathbf{M}}^2$$

- **Approximate $\mathbf{A}^{32}$:**

$$\mathbf{A}^{32} \overset{\varepsilon_4^2}{\approx} 2\mathbf{A}^{16}\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$

$$\overset{2\varepsilon_3^2}{\approx} 4\mathbf{A}^{8}\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 2\left(\widetilde{\mathbf{A}^{8}}\right)^2\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$

Boost weaker approximation to stronger approximation

- **Example:** (For simplicity, assume matrix multiplication commutes)

- **Weak Approximations:**
$\left\| \widetilde{\mathbf{A}^{16}} - \mathbf{A}^{16} \right\| \leq \varepsilon_4$, $\left\| \widetilde{\mathbf{A}^{8}} - \mathbf{A}^{8} \right\| \leq \varepsilon_3$, $\left\| \widetilde{\mathbf{A}^{4}} - \mathbf{A}^{4} \right\| \leq \varepsilon_2$, $\left\| \widetilde{\mathbf{A}^{2}} - \mathbf{A}^{2} \right\| \leq \varepsilon_1$

$$\left\| \widetilde{\mathbf{M}} - \mathbf{M} \right\| \leq \varepsilon \implies \left\| \left(\widetilde{\mathbf{M}} - \mathbf{M} \right)^2 \right\| \leq \varepsilon^2$$
$$\implies \mathbf{M}^2 \overset{\varepsilon^2}{\approx} 2\mathbf{M}\widetilde{\mathbf{M}} - \widetilde{\mathbf{M}}^2$$

- **Approximate $\mathbf{A}^{32}$:**

$$\mathbf{A}^{32} \overset{\varepsilon_4^2}{\approx} 2\mathbf{A}^{16}\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$
$$\overset{2\varepsilon_3^2}{\approx} 4\mathbf{A}^{8}\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 2\left(\widetilde{\mathbf{A}^{8}}\right)^2\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$
$$\overset{4\varepsilon_2^2}{\approx} 8\mathbf{A}^{4}\left(\widetilde{\mathbf{A}^{4}}\right)\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 4\left(\widetilde{\mathbf{A}^{4}}\right)^2\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 2\left(\widetilde{\mathbf{A}^{8}}\right)^2\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$

Boost weaker approximation to stronger approximation

- **Example:** (For simplicity, assume matrix multiplication commutes)
- **Weak Approximations:**
  $\left\|\widetilde{\mathbf{A}^{16}} - \mathbf{A}^{16}\right\| \le \varepsilon_4$, $\left\|\widetilde{\mathbf{A}^{8}} - \mathbf{A}^{8}\right\| \le \varepsilon_3$, $\left\|\widetilde{\mathbf{A}^{4}} - \mathbf{A}^{4}\right\| \le \varepsilon_2$, $\left\|\widetilde{\mathbf{A}^{2}} - \mathbf{A}^{2}\right\| \le \varepsilon_1$

$$\left\|\widetilde{\mathbf{M}} - \mathbf{M}\right\| \le \varepsilon \implies \left\|\left(\widetilde{\mathbf{M}} - \mathbf{M}\right)^2\right\| \le \varepsilon^2$$

$$\implies \mathbf{M}^2 \overset{\varepsilon^2}{\approx} 2\mathbf{M}\widetilde{\mathbf{M}} - \widetilde{\mathbf{M}}^2$$

- **Approximate $\mathbf{A}^{32}$:**

$$\mathbf{A}^{32} \overset{\varepsilon_4^2}{\approx} 2\mathbf{A}^{16}\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$

$$\overset{2\varepsilon_3^2}{\approx} 4\mathbf{A}^{8}\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 2\left(\widetilde{\mathbf{A}^{8}}\right)^2\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$

$$\overset{4\varepsilon_2^2}{\approx} 8\mathbf{A}^{4}\left(\widetilde{\mathbf{A}^{4}}\right)\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 4\left(\widetilde{\mathbf{A}^{4}}\right)^2\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 2\left(\widetilde{\mathbf{A}^{8}}\right)^2\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$

$$\overset{16\varepsilon_1}{\approx} 8\left(\widetilde{\mathbf{A}^{2}}\right)^2\left(\widetilde{\mathbf{A}^{4}}\right)\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 4\left(\widetilde{\mathbf{A}^{4}}\right)^2\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 2\left(\widetilde{\mathbf{A}^{8}}\right)^2\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2$$

Boost weaker approximation to stronger approximation

- **Example:** (For simplicity, assume matrix multiplication commutes)

- **Weak Approximations:**
$\left\|\widetilde{\mathbf{A}^{16}} - \mathbf{A}^{16}\right\| \leq \varepsilon_4$, $\left\|\widetilde{\mathbf{A}^{8}} - \mathbf{A}^{8}\right\| \leq \varepsilon_3$, $\left\|\widetilde{\mathbf{A}^{4}} - \mathbf{A}^{4}\right\| \leq \varepsilon_2$, $\left\|\widetilde{\mathbf{A}^{2}} - \mathbf{A}^{2}\right\| \leq \varepsilon_1$

- **Approximate $\mathbf{A}^{32}$: (error:** $\boxed{\varepsilon_4^2 + 2\varepsilon_3^2 + 4\varepsilon_2^2 + 16\varepsilon_1}$**)**

$$
\begin{aligned}
\mathbf{A}^{32} &\overset{\varepsilon_4^2}{\approx} 2\mathbf{A}^{16}\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2 \\
&\overset{2\varepsilon_3^2}{\approx} 4\mathbf{A}^{8}\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 2\left(\widetilde{\mathbf{A}^{8}}\right)^2\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2 \\
&\overset{4\varepsilon_2^2}{\approx} 8\mathbf{A}^{4}\left(\widetilde{\mathbf{A}^{4}}\right)\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 4\left(\widetilde{\mathbf{A}^{4}}\right)^2\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 2\left(\widetilde{\mathbf{A}^{8}}\right)^2\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2 \\
&\overset{16\varepsilon_1}{\approx} 8\left(\widetilde{\mathbf{A}^{2}}\right)^2\left(\widetilde{\mathbf{A}^{4}}\right)\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 4\left(\widetilde{\mathbf{A}^{4}}\right)^2\left(\widetilde{\mathbf{A}^{8}}\right)\left(\widetilde{\mathbf{A}^{16}}\right) - 2\left(\widetilde{\mathbf{A}^{8}}\right)^2\left(\widetilde{\mathbf{A}^{16}}\right) - \left(\widetilde{\mathbf{A}^{16}}\right)^2
\end{aligned}
$$

The Iteration Formula

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k,t \le O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \le i \le \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

The Iteration Formula

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \le O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \le i \le \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

Lemma (Iteration Formula)

For any positive integers $k \ge t$, suppose $\mathbf{B}_i$ is an approximation of $\mathbf{A}^{2^i}$ such that $\left\|\mathbf{B}_i - \mathbf{A}^{2^i}\right\| \le \varepsilon_i$ *(for $i = 1, 2, \cdots, k-1$). Define*

$$\mathbf{C} := -\sum_{i=1}^{t-1} \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-i+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-i}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q} + \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-t+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-t}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q}.$$

Then $\left\|\mathbf{C} - \mathbf{A}^{2^k}\right\| \le \varepsilon_{k-1}^2 + 2\varepsilon_{k-2}^2 + 4\varepsilon_{k-3}^2 + \cdots + 2^{t-2}\varepsilon_{k-t+1}^2 + 2^t \varepsilon_{k-t}.$

The Iteration Formula

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k,t \le O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k,t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \le i \le \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

Lemma (Iteration Formula)

$$\varepsilon_{k-i} \le 1/2^{\lfloor t/2 \rfloor + 2i} \implies \left\| \mathbf{C} - \mathbf{A}^{2^k} \right\| \le 0.99/2^t$$

For any positive integers $k \ge t$, suppose $\left\| \mathbf{B}_i - \mathbf{A}^{2^i} \right\| \le \varepsilon_i$ *(for $i = 1, 2, \cdots, k-1$). Define*

$$\mathbf{C} := -\sum_{i=1}^{t-1} \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-i+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-i}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q} + \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-t+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-t}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q}.$$

Then $\left\| \mathbf{C} - \mathbf{A}^{2^k} \right\| \le \varepsilon_{k-1}^2 + 2\varepsilon_{k-2}^2 + 4\varepsilon_{k-3}^2 + \cdots + 2^{t-2}\varepsilon_{k-t+1}^2 + 2^t \varepsilon_{k-t}.$

The Iteration Formula

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k,t \le O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k,t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \le i \le \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ **in $O(t)$-depth**.

Lemma (Iteration Formula)

For any positive integers $k \ge t$, suppose $\mathbf{B}_i$ is an approximation of $\mathbf{A}^{2^i}$ such that $\left\| \mathbf{B}_i - \mathbf{A}^{2^i} \right\| \le \varepsilon_i$ *(for $i = 1, 2, \cdots, k-1$). Define*

$$\mathbf{C} := -\sum_{i=1}^{t-1} \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-i+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-i}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q} + \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-t+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-t}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q}.$$

Then $\left\| \mathbf{C} - \mathbf{A}^{2^k} \right\| \le \varepsilon_{k-1}^2 + 2\varepsilon_{k-2}^2 + 4\varepsilon_{k-3}^2 + \cdots + 2^{t-2}\varepsilon_{k-t+1}^2 + 2^t \varepsilon_{k-t}.$

The Iteration Formula

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \le O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \le i \le \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ **in $O(t)$-depth**.

Lemma (Iteration For

For any positive integers k ≥

$\left\| \mathbf{B}_i - \mathbf{A}^{2^i} \right\| \le \varepsilon_i$ *(for $i = 1$*

1. Compute all matrix operations with **error** $1/2^{O(t)}$.
   Depth: $O\left(\frac{t}{\log\log n} + 1\right) \le O\left(\frac{t}{\log t}\right)$.
2. Multiplication of $O(t)$ matrices in each term.

$$\mathbf{C} := -\sum_{i=1}^{t-1} \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-i+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-i}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q} + \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-t+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-t}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q}.$$

Then $\left\| \mathbf{C} - \mathbf{A}^{2^k} \right\| \le \varepsilon_{k-1}^2 + 2\varepsilon_{k-2}^2 + 4\varepsilon_{k-3}^2 + \cdots + 2^{t-2}\varepsilon_{k-t+1}^2 + 2^t \varepsilon_{k-t}.$

The Iteration Formula

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \le O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \le i \le \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

Lemma (Iteration Formula)

For any positive integers $k \ge t$, suppose $\mathbf{B}_i$ is an approximation of $\mathbf{A}^{2^i}$ such that $\left\|\mathbf{B}_i - \mathbf{A}^{2^i}\right\| \le \varepsilon_i$ *(for $i = 1, 2, \cdots, k-1$). Define*

$$\mathbf{C} := -\sum_{i=1}^{t-1} \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-i+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-i}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q} + \sum_{\substack{\{j_1 < \cdots < j_p\} \uplus \{j'_1 < \cdots < j'_q\} \\ = \{k-1, k-2, \cdots, k-t+1\}}} \mathbf{B}_{j_p} \cdots \mathbf{B}_{j_1} \mathbf{B}_{k-t}^2 \mathbf{B}_{j'_1} \cdots \mathbf{B}_{j'_q}.$$

Then $\left\|\mathbf{C} - \mathbf{A}^{2^k}\right\| \le \varepsilon_{k-1}^2 + 2\varepsilon_{k-2}^2 + 4\varepsilon_{k-3}^2 + \cdots + 2^{t-2}\varepsilon_{k-t+1}^2 + 2^t \varepsilon_{k-t}.$

Subroutine → Complete Algorithm

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k,t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k,t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \leq i \leq \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

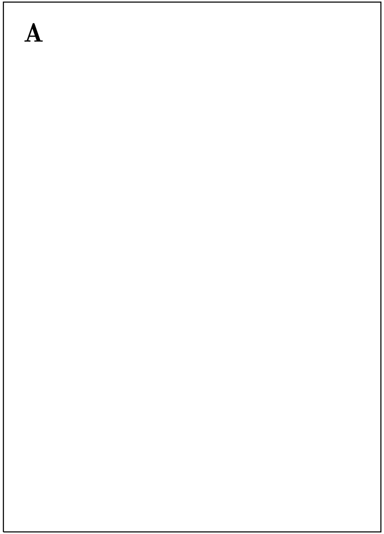
Subroutine → Complete Algorithm

- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \leq i \leq \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

Subroutine → Complete Algorithm

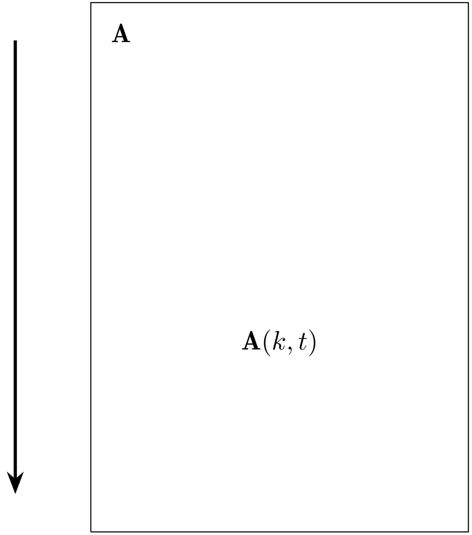


- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \leq i \leq \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

Subroutine → Complete Algorithm

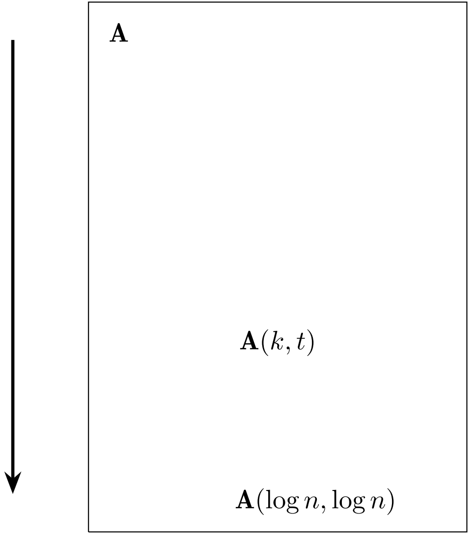


- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \le O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s $(1 \le i \le \min\{k, O(t)\})$, we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

Subroutine → Complete Algorithm

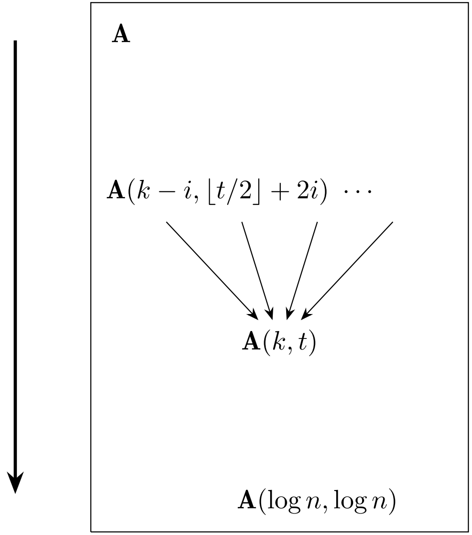


- Intermediate matrices $\mathbf{A}(k, t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k - i, \lfloor t/2 \rfloor + 2i)$'s ($1 \leq i \leq \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k, t)$ in $O(t)$-depth.

Subroutine → Complete Algorithm



- Intermediate matrices $\mathbf{A}(k, t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

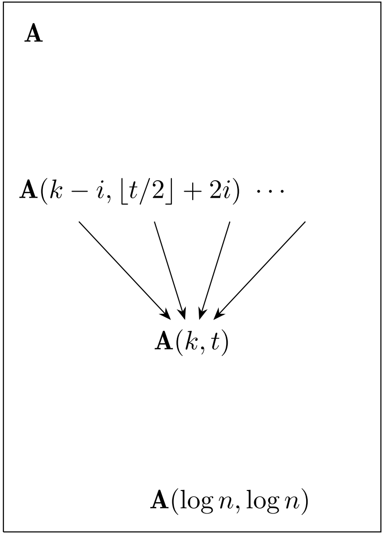
For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \leq i \leq \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k, t)$ in $O(t)$-depth.

Subroutine → Complete Algorithm

Depth

0

$C \cdot (2k + t)$



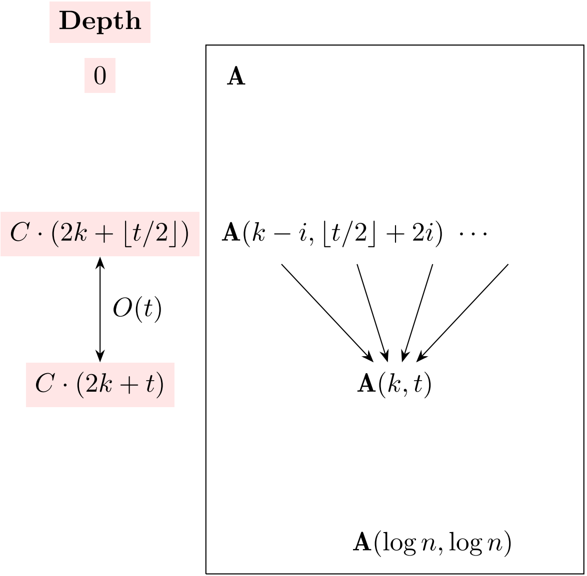
- Intermediate matrices $\mathbf{A}(k, t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \le O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k - i, \lfloor t/2 \rfloor + 2i)$'s ($1 \le i \le \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k, t)$ in $O(t)$-depth.

- **Potential function** $\phi(k, t) := 2k + t$

Subroutine → Complete Algorithm



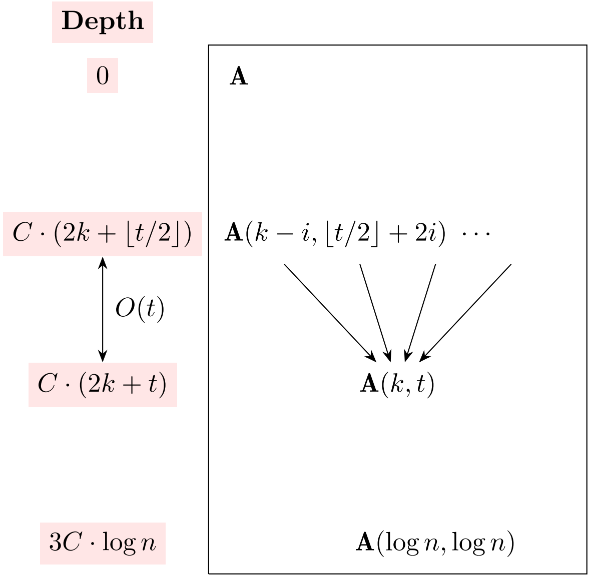
- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k,t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k,t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s $(1 \leq i \leq \min\{k, O(t)\})$, we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

- **Potential function** $\phi(k,t) := 2k + t$

Subroutine → Complete Algorithm



- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \le O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k-i, \lfloor t/2 \rfloor + 2i)$'s ($1 \le i \le \min\{k, O(t)\}$), we can compute a valid $\mathbf{A}(k,t)$ in $O(t)$-depth.

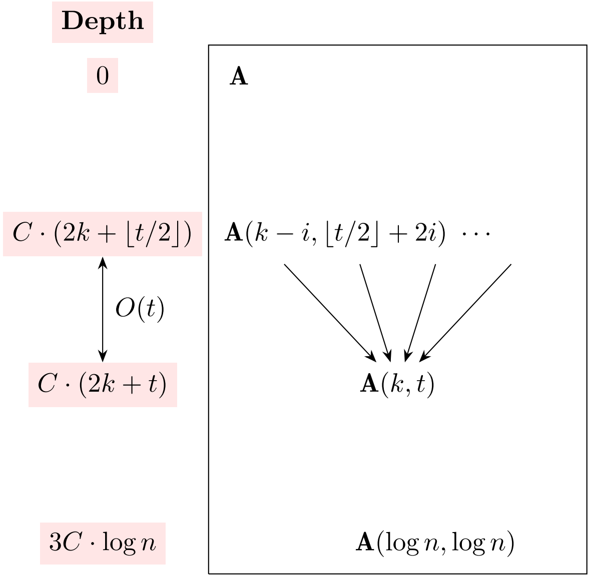
- **Potential function** $\phi(k,t) := 2k + t$

Summary

- **Part1.** Non-trivial L-AC algorithms
  - Approximate counting in L-AC.
  - Low-precision matrix operations in L-AC.
    (e.g. Multiply $n \times n$ matrices with $1/3$ error in $\mathsf{L\text{-}AC}^0$)

- **Part2.** Use low-precision operations to compute matrix powering
  - Recall: estimate $\mathbf{A} \mapsto \mathbf{A}^n$ with $1/n$-error is BPL-complete.
  - Numerical analysis techniques! (Popular in recent years)

Subroutine → Complete Algorithm



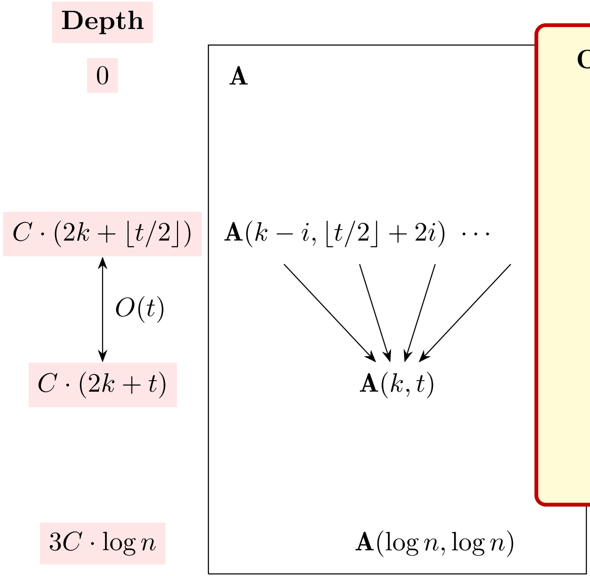
- Intermediate matrices $\mathbf{A}(k,t)$: a $1/2^t$-approximation of $\mathbf{A}^{2^k}$ (for $k, t \leq O(\log n)$)
  (Goal: given $\mathbf{A}$, compute a valid $\mathbf{A}(\log n, \log n)$)

Lemma (Subroutine)

For any k, t, given all $\mathbf{A}(k - i, \lfloor t/2 \rfloor + 2i)$'s $(1 \leq i \leq \min\{k, O(t)\})$, we can compute a valid $\mathbf{A}(k, t)$ in $O(t)$-depth.

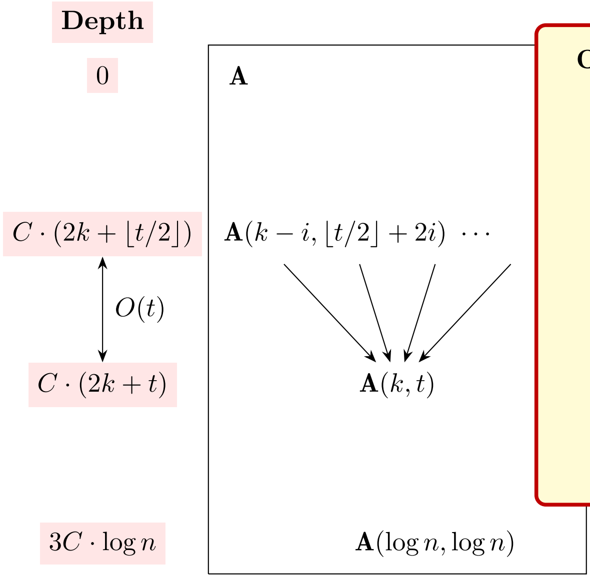
- **Potential function** $\phi(k,t) := 2k + t$

Subroutine → Complete Algorithm



Our Iteration Formula v.s. Richardson

Subroutine → Complete Algorithm



Our Iteration Formula v.s. Richardson

- **Richardson**
  - Saks-Zhou argument: $\mathbf{A} \mapsto \mathbf{A}^{2^{\sqrt{\log n}}}$ for $\sqrt{\log n}$ rounds.
  - Weaker result: $\mathsf{BPL} \subseteq \mathsf{L}\text{-}\mathsf{AC}^{1.5}$.
  - Each round depends on the previous round, not efficiently parallized!
- **Our Iteration Formula**
  - More efficiently parallized!

Thank you!

- Q&A

Thank you!

- Q&A
- Thank you for listening!