IEEE International Symposium on Information Theory (ISIT) 2017, Aachen - Germany

Minimizing Latency for Secure Distributed Computing

Rawad Bitar

Illinois Institute of Technology

Joint work with

Parimal Parag and Salim El Rouayheb

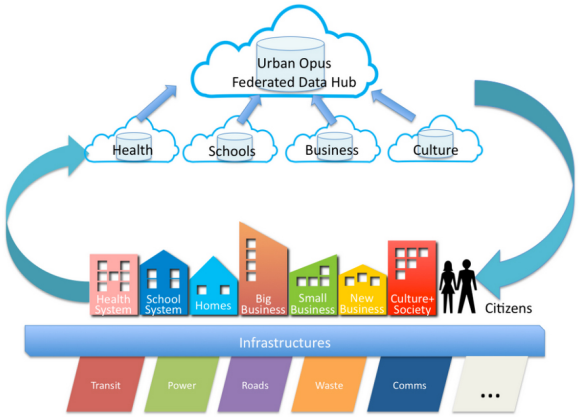
IISc – Bangalore IIT - Chicago

Distributed Computing and Applications

Distributing tasks within data centers (MapReduce)

Outsourcing computations to volunteers (crowdsourcing)

Outsourcing computation to companies

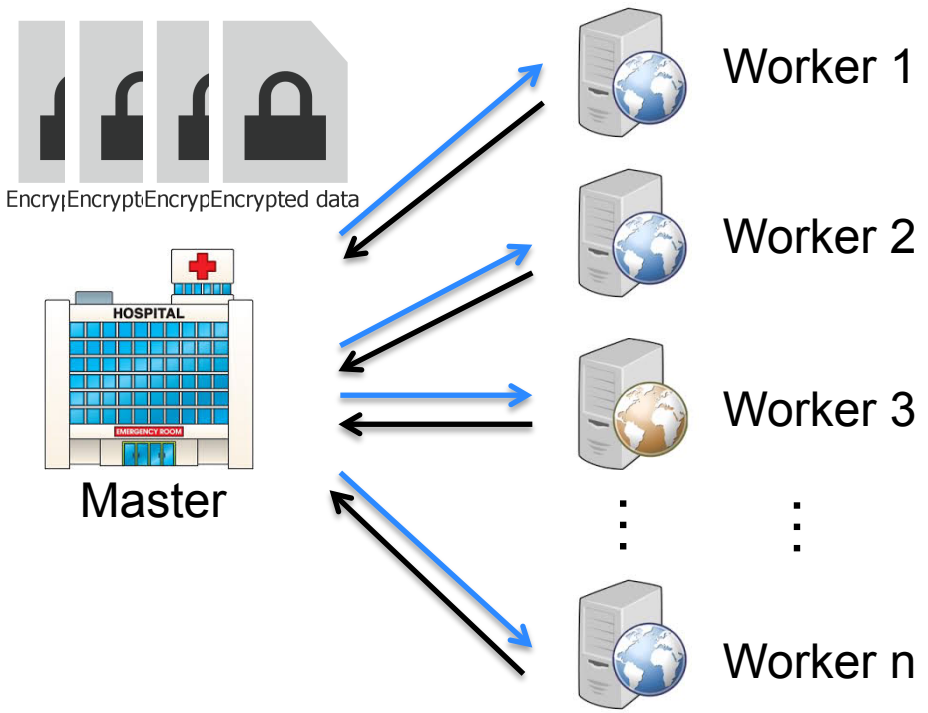


Federated learning

Medical research
Genome sequencing
DNA sequencing…

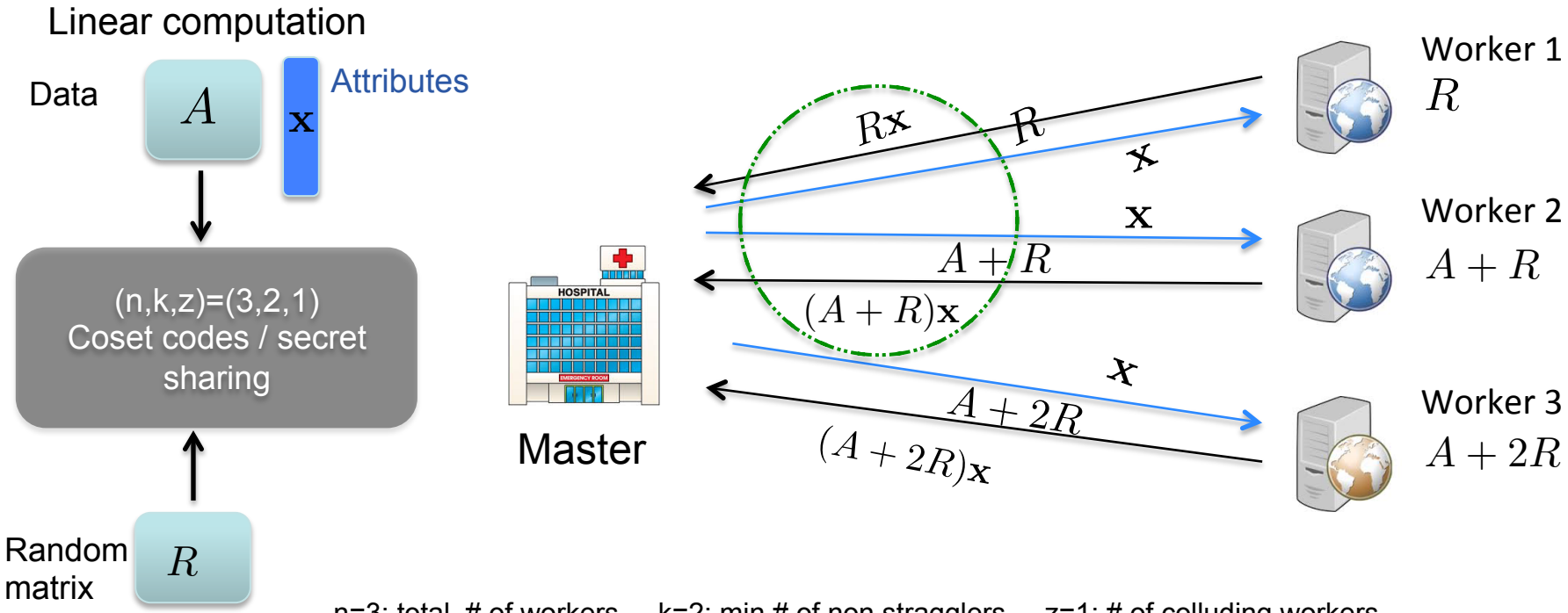
Main concern: Confidentiality

Secure Distributed Computing



- Data must remain confidential from the workers (passive eavesdropper)
- Linear computation over encrypted data
- Computational secrecy is achieved using homomorphic encryption
- Secure distributed computing with information theoretic guarantees

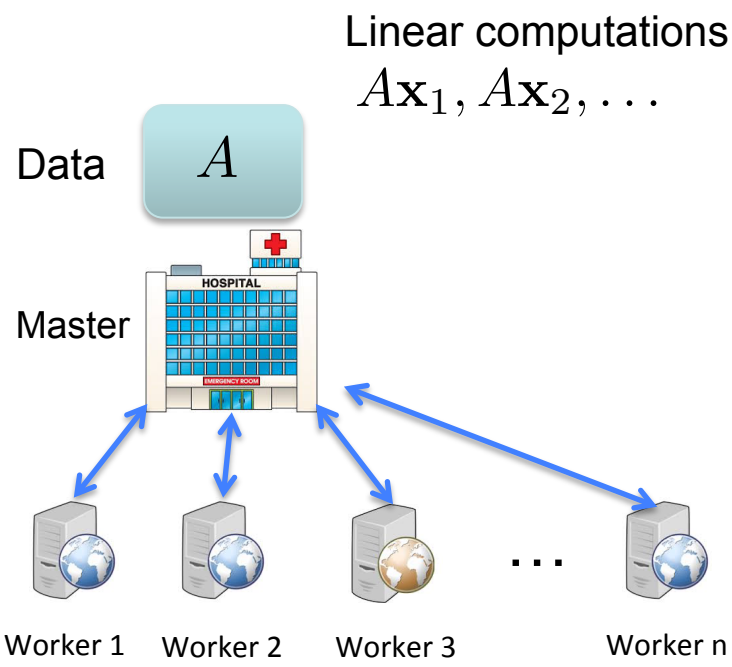
ARE "CLASSICAL CODES" EFFICIENT?



n=3: total # of workers, k=2: min # of non stragglers, z=1: # of colluding workers

- Master decodes $A\mathbf{x} = (A+R)\mathbf{x} - R\mathbf{x}$
- Master has to decode $R\mathbf{x}$

Model of Secure Distributed Computing



- n = # of workers
- n-k = max # of stragglers
- z = max # of colluding workers (passive adversary)

- Information theoretic secrecy
  $H(A|\text{observation of any } z \text{ workers}) = H(A)$
- Iterative algorithm
- T_w = upload $\mathbf{x}$ + computation + download result
- T_w's are iid shifted exponential [Liang and Kozat '14]

$$F_{T_w}(t) = \begin{cases} 0 & \text{if } t < c/(k-z), \\ 1 - e^{-\lambda(k-z)(t-\frac{c}{k-z})} & \text{otherwise.} \end{cases}$$

- T_M : Master waiting time

Goal: Design codes that *minimize* T_M and guarantee information theoretic secrecy

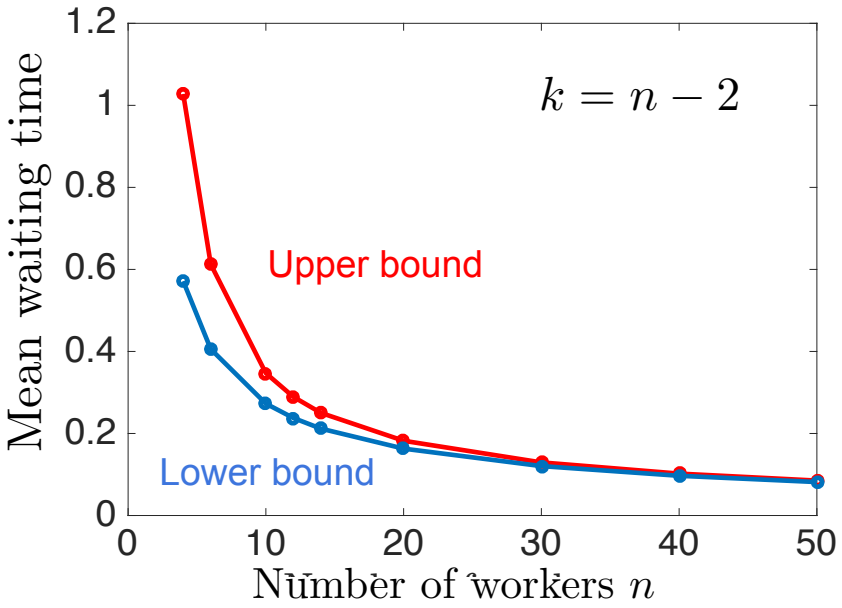
Our Results: 1) Bounds on $\mathbb{E}[T_{\text{MSC}}]$

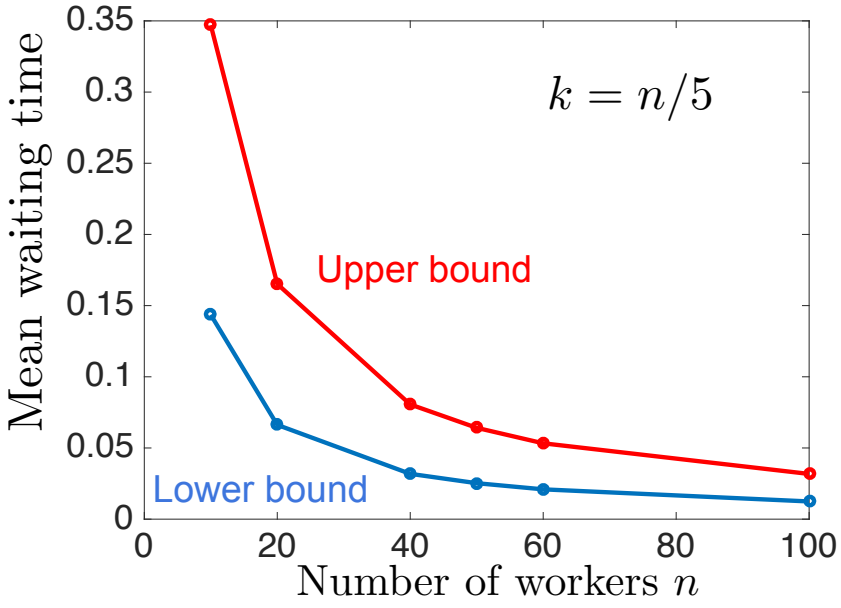
Theorem 1: [Bounds on mean waiting time] The mean waiting time $\mathbb{E}[T_{\text{MSC}}]$ of an (n, k, z) system using Staircase codes is upper bounded by

$$\mathbb{E}[T_{\text{MSC}}] \leq \min_{d \in \{k,\ldots,n\}} \left(\frac{H_n - H_{n-d}}{\lambda(d-z)} + \frac{c}{d-z} \right), \qquad (1)$$

where H_n is the n^{th} harmonic sum defined as $H_n \triangleq \sum_{i=1}^{n} \frac{1}{i}$, and $H_0 \triangleq 0$. The mean waiting time is lower bounded by

$$\mathbb{E}[T_{\text{MSC}}] \geq \frac{c}{n-z} + \max_{d \in \{k,\ldots,n\}} \sum_{i=0}^{k-1} \binom{n}{i} \sum_{j=0}^{i} \binom{i}{j} \frac{2(-1)^j}{\lambda \left(2(n-i+j)(d-z) + (n-d)(n-d+1)\right)}. \qquad (2)$$

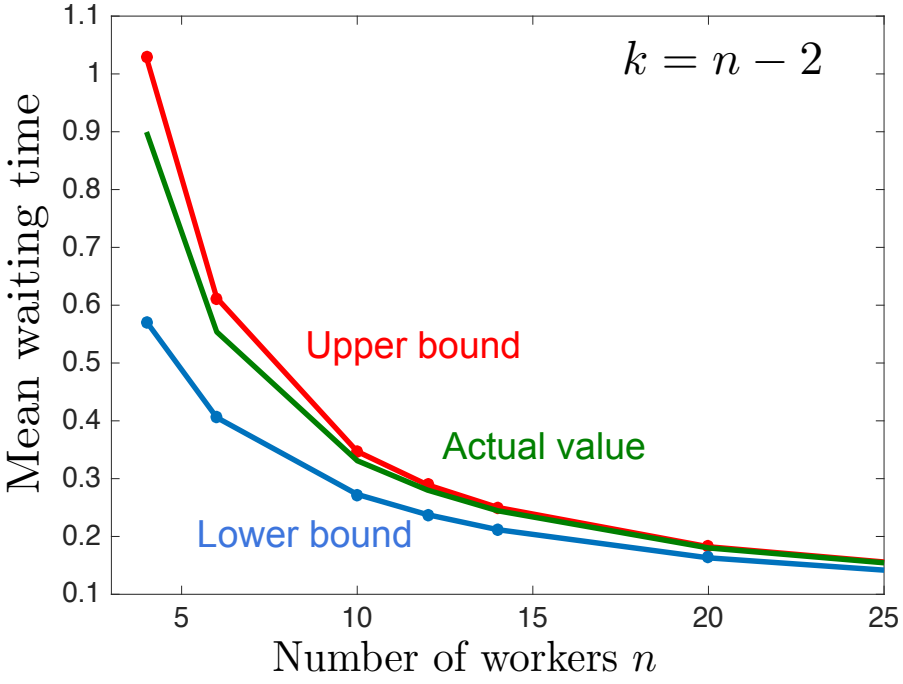


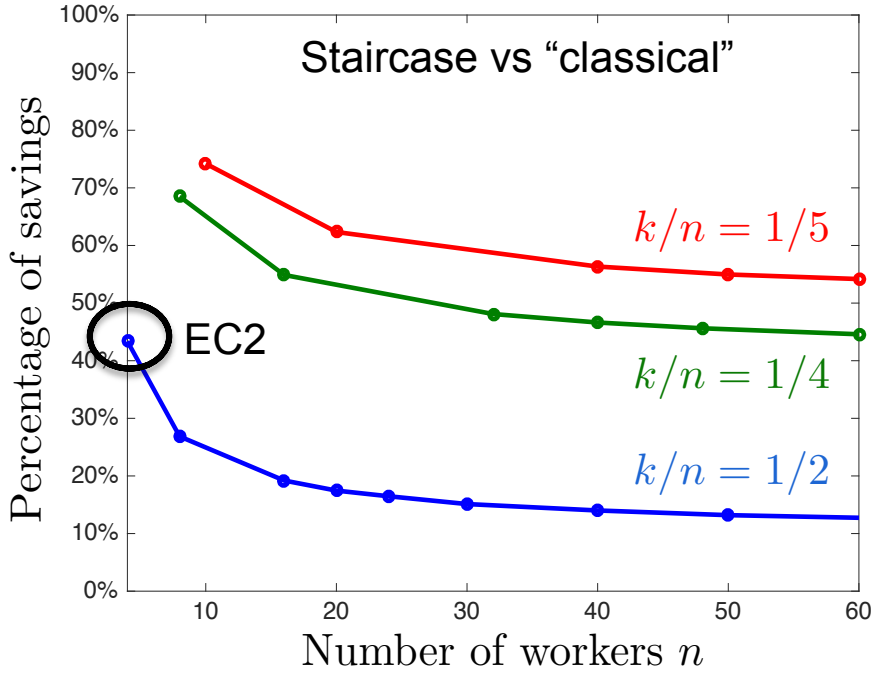


Our Results: 2) Distribution of T_{MSC}

Theorem 2: [Exact mean waiting time] The mean waiting time $\mathbb{E}[T_{\text{MSC}}]$ for a $(k+1, k, z)$ systems using Staircase codes is given by

$$\mathbb{E}\left[T_{\text{MSC}}\right] = \frac{c}{k-z+1} + \frac{1}{\lambda}\sum_{i=1}^{k+1}(-1)^i \binom{k+1}{i}\left[\frac{i\exp\left(\frac{-\lambda c}{k-z}\right)}{(k-z)i+1} - \frac{1}{(k-z+1)i}\right]$$





Related Work on Straggler Mitigation

Use of codes to reduce delays, e.g.,
MDS codes [Huang, Pawar, Zhang, and Ramchandran '12]
Task replication [Wang, Joshi, Wornell '14]
Availability codes [Kadhe, Soljanin, and Sprintson '15]

Accounting for workers workload, e.g.,
Adaptive encoding [Liang and Kozat '14]

Existence of a tradeoff, e.g.,
Between storage cost and content download time [Joshi, Liu and Soljanin '14]

Latency analysis, e.g.,
[Parag, Bura and Chamberland '17], [Shah, Bouillard, Baccelli '17]

Use of codes for specific application, e.g.,
Matrix multiplication [Lee, Lam, Pedarsani, Papailiopoulos and Ramchandran '16]
Gradient descent [Tandon, Lei, Dimakis, Karampatziakis '16]
Inverse linear problems [Yang, Grover and Kar '17]
Convolution of two long vectors [Dutta, Cadambe and Grover '17]

Accounting for workers workload, e.g.,
Heterogeneous clusters [Reisizadehmobarakeh, Prakash, Pedarsani, Avestimehr '17]

Tradeoff between computation and download, e.g.,
[Li, Maddah Ali and Avestimehr '16]

Accounting for computation at the worker, e.g.,
[Dutta, Cadambe, and Grover '16], [Yu, Maddah-Ali and Avestimehr '17]
[Halbawi, Azizan-Ruhi, Salehi and Hassibi '17]

Plus tutorial and lots of good talks at this ISIT… but how about secrecy?

Related Work on Secrecy

Information theoretic secrecy, e.g.,

Secure matrix multiplication using Shamir secret sharing [Atallah and Frikken '10]

Homomorphic encryption, e.g.,

Linear regression on encrypted data based on homomorphic encryption and Yao garbled circuit [Nikolaenko, Weinsberg, Ioannidis, Joye, Boneh and Taft '13]

Privacy preserving multi-party deep neural network based on homomorphic encryption [Takabi, Hesamifard, and Ghasemi '16]

Survey on privacy and Genome sequencing [Naveed, Ayday, Clayton, Fellay, Gunter, Hubaux, Malin, and Wang '15]

Ingredients of Our Main Results

Minimum latency secure distributed computing

Comparison to “classical” codes
Implementation on EC2

Coding theory

Staircase codes
[**B.** and El Rouayheb T-IT ‘17]

Queuing theory

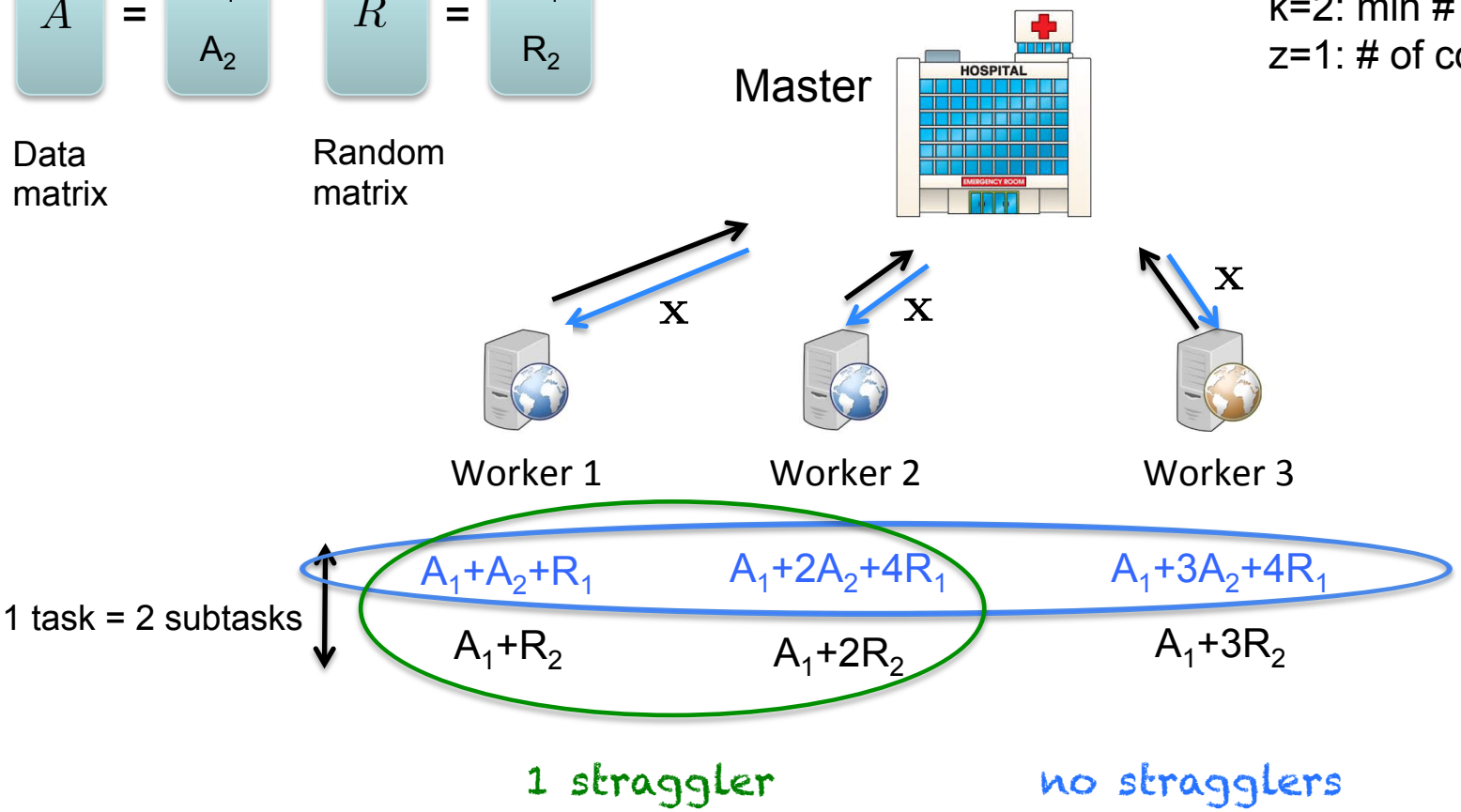
Service time
Order Statistics
Concentration bounds

Example of Staircase Codes

A = A_1 / A_2 — Data matrix

R = R_1 / R_2 — Random matrix

n=3: total # of workers
k=2: min # of non stragglers
z=1: # of colluding workers



- ✓ Master has two options for decoding Ax (choose the faster)
- ✓ Master does not have to decode Rx (except when n-k workers are stragglers)

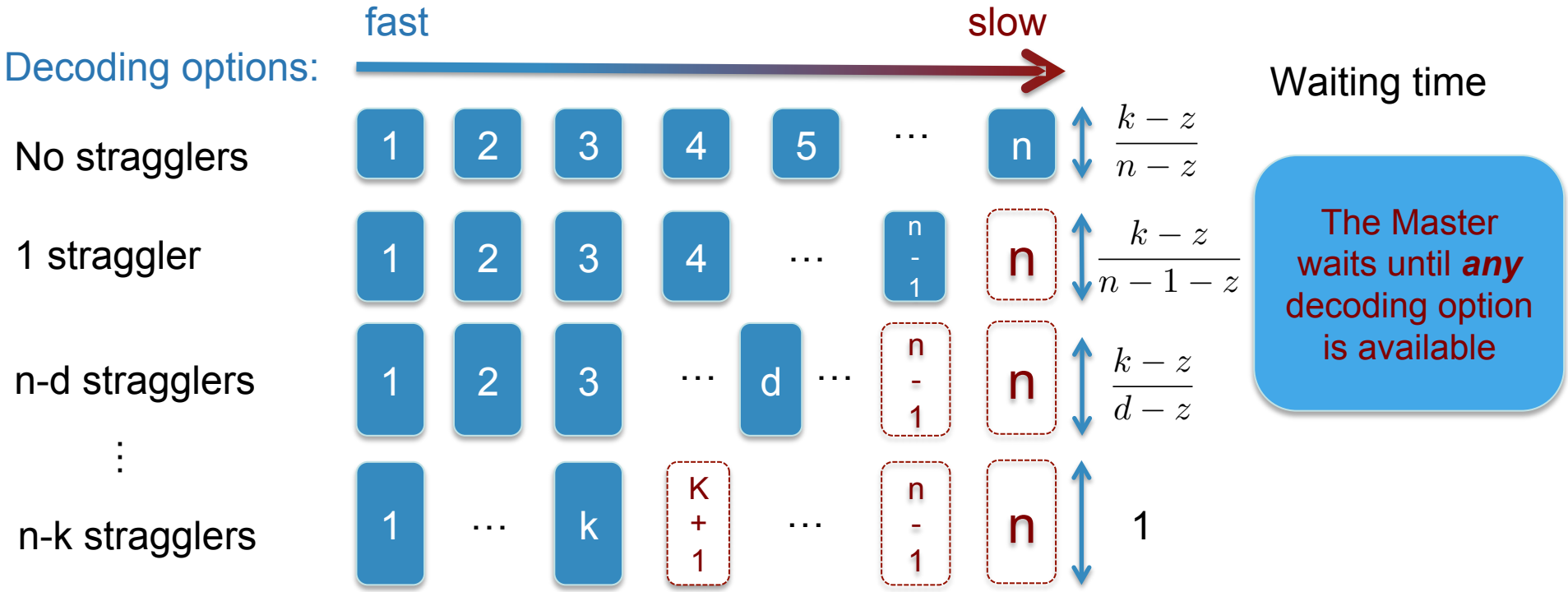
Abstraction of Staircase Codes

- (n, k, z) Universal staircase codes

$$\begin{bmatrix} \text{Share } 1 \\ \vdots \\ \text{Share } n \end{bmatrix} = n \left\{ \text{Vandermonde} \right. \overbrace{\quad}^{n} \begin{bmatrix} \mathcal{A} & \mathcal{D}_1 & \mathcal{D}_2 & \cdots & \mathcal{D}_{n-k} \\ & \mathcal{R}_2 & \mathcal{R}_3 & \cdots & \mathcal{R}_{n-k+1} \\ \mathcal{R}_1 & \bar{\mathbf{0}} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}$$

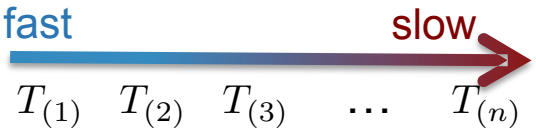
Polynomial number of subtasks

Decoding options:



Taste of the Proofs

Notation:



T_i : time spent by worker i
$T_i's$: are iid
$T_{(d)}$: d[th] order statistic of the $T_i's$
T_{MSC} : Master's waiting time

Waiting time: $$T_{\text{MSC}} = \min_{d\in\{k,\ldots,n\}} \left\{ \frac{k-z}{d-z} T_{(d)} \right\}$$

[Renyi '53]

Upper bound: $$\mathbb{E}[T_{\text{SC}}] = \mathbb{E}\left[\min_{d\in\{k,\ldots,n\}} \left\{ \frac{k-z}{d-z} T_{(d)} \right\} \right] \leq \min_{d\in\{k,\ldots,n\}} \left\{ \frac{H_n - H_{n-d}}{\lambda(d-z)} + \frac{c}{d-z} \right\}$$

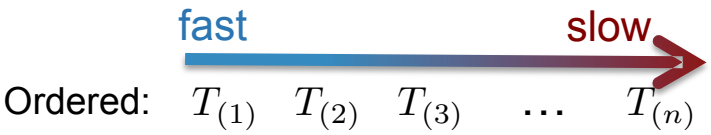
Jensen's inequality

Lower bound: $$\Pr\left(T_{MSC} > t\right) \geq \max_{d\in\{k,\ldots,n\}} \Pr\left(\left\{ T_{(k)} > \frac{t(d-z)}{k-z} \right\} \bigcap_{j=d+1}^{n} \left\{ T_{(j)} - T_{(j-1)} > \frac{t}{k-z} \right\} \right)$$

Properties of exponential random variables

Taste of the Proofs (Cont'd)

Notation:



T_i : time spent by worker i
$T_i's$: are iid
$T_{(d)}$: d[th] order statistic of the $T_i's$
T_{MSC} : Master's waiting time

Waiting time: $$T_{\text{MSC}} = \min_{d\in\{k,\ldots,n\}} \left\{ \frac{k-z}{d-z} T_{(d)} \right\}$$

Distribution:

$$\Pr\{T_{\text{MSC}} > t\} = \Pr \bigcap_{d=k}^{n} \left\{ T_{(d)} > \frac{t(d-z)}{k-z} \right\}$$

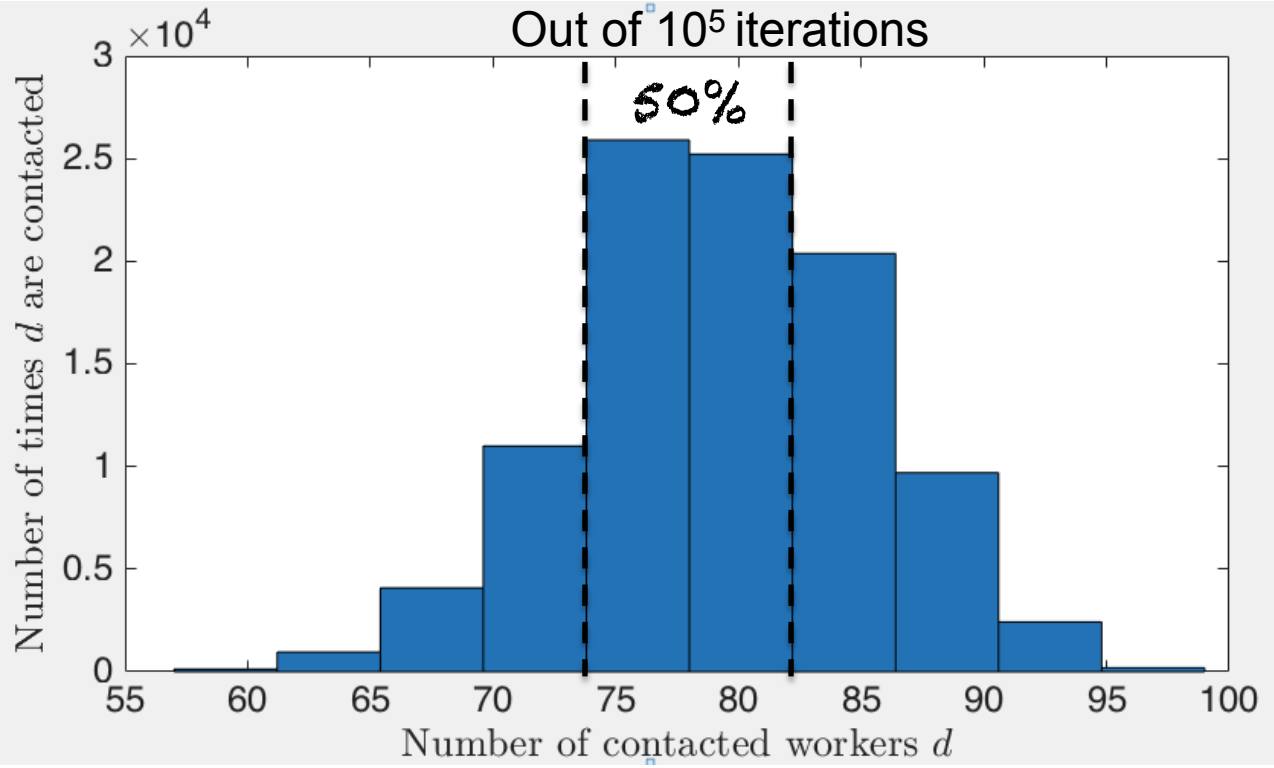
induction

$$= n! \int_{\frac{t(n-z)}{k-z}}^{\infty} \cdots \int_{t}^{y_{k+1}} \prod_{i=k}^{n} dF_{T_i}(y_i) \left(\int_{\frac{c}{k-z}}^{y_k} \cdots \int_{\frac{c}{k-z}}^{y_2} \prod_{i=1}^{k-1} dF_{T_i}(y_i) \right)$$

$$= \frac{F_{T_i}(y_k)^{k-1}}{(k-1)!} n! \int_{\frac{t(n-z)}{k-z}}^{\infty} \cdots \int_{t}^{y_{k+1}} \prod_{i=k}^{n} dF_{T_i}(y_i)$$

Can We Reduce the Number of Subtasks?

Yes, by using non-Universal Staircase codes!

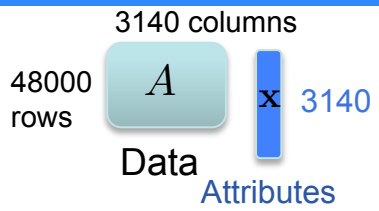


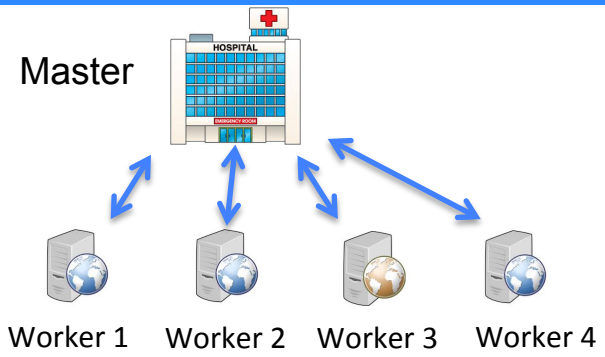
(n,k,z)=(100,50,1)

n: total # of workers
k: min # of non stragglers
z=1: # of colluding workers

[R. B., P. Parag and S. El Rouayheb, "Minimizing Latency for Secure Distributed Computing"] (extended version to be on arxiv)

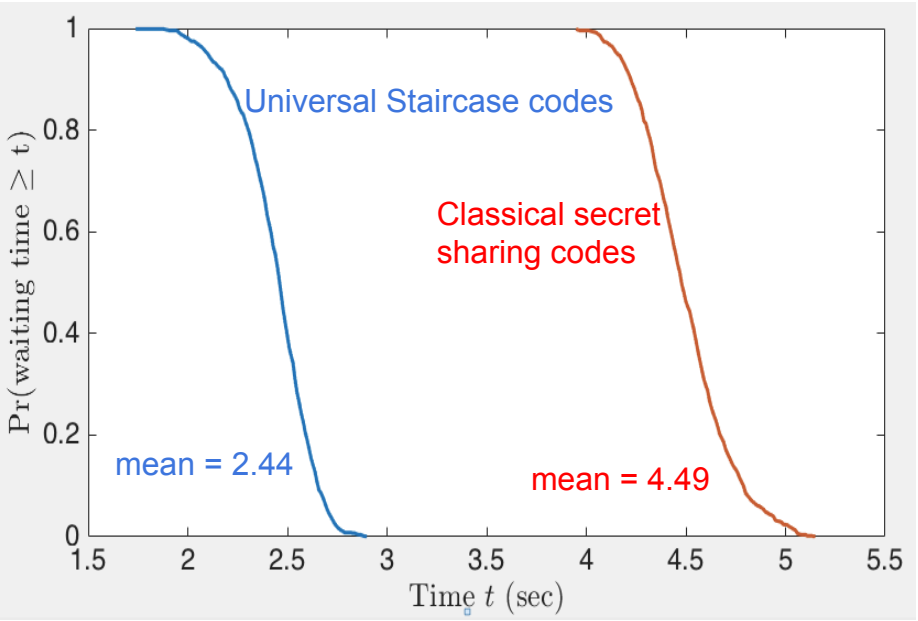
Implementation on Amazon EC2



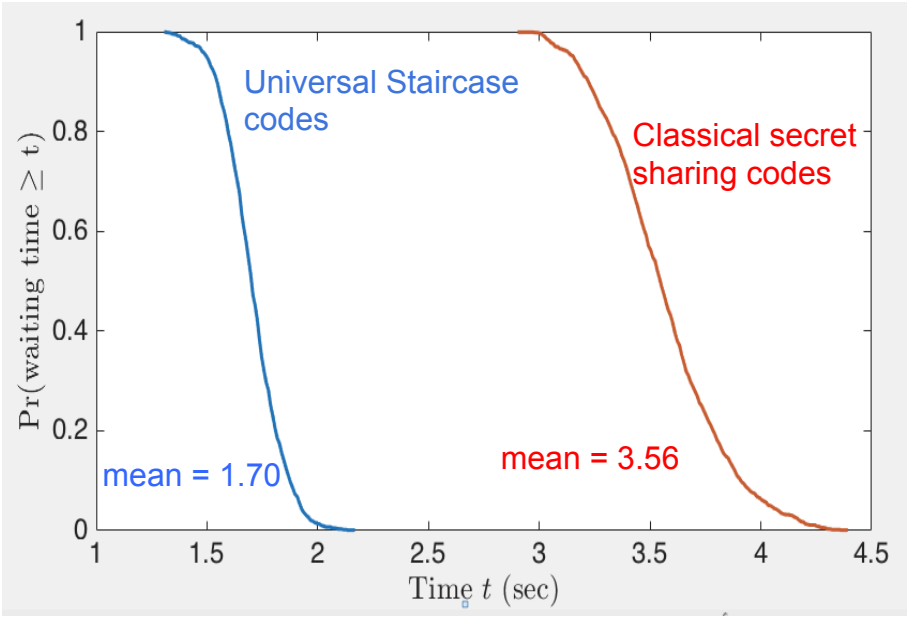


n=4: total # of workers
k=2: min # of non stragglers
z=1: # of colluding workers

Master in Oregon (west) and workers in Ohio (east)



Master and workers in Ohio (east)



[R. B., P. Parag and S. El Rouayheb, "Minimizing Latency for Secure Distributed Computing"] (extended version to be on arxiv)

Summary

- **Problem of interest:** straggler mitigation with secrecy constrains
- In the paper
  - Delay modeled by **exponential** random variables (shift = 0)
  - **No colluding workers (z = 1)**
  - **Upper and lower bound** on the Master's waiting time
  - **Expression** to derive the **exact PDF** of the Master's waiting time
  - **Exact PDF** for systems with **1 or 2 parities**

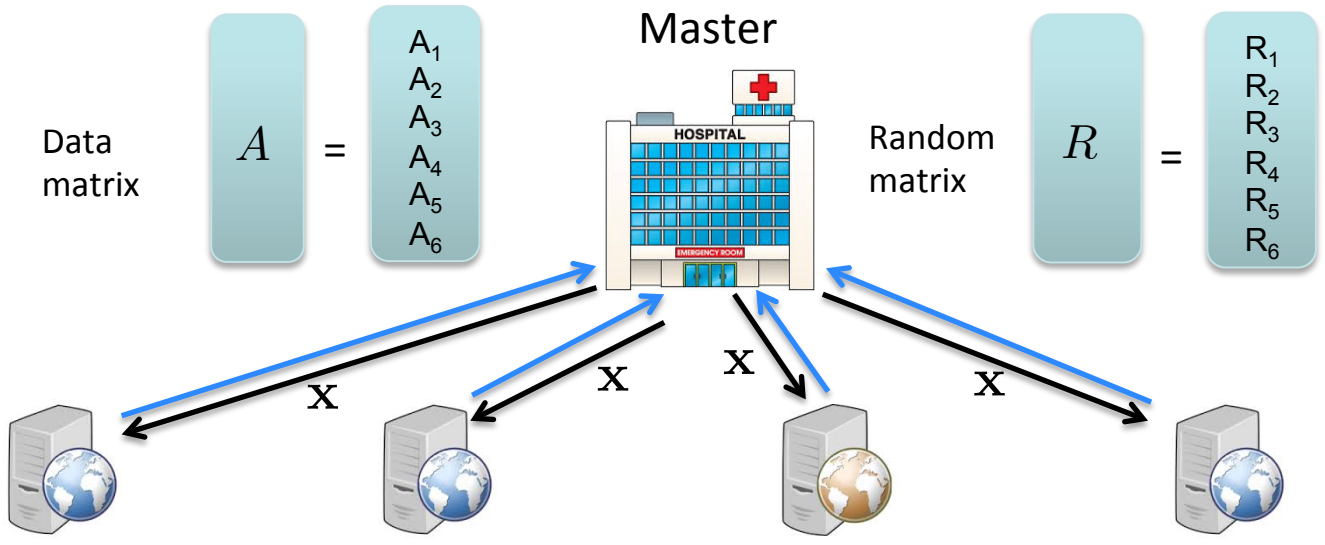
- In the extended version
  - Generalization to **shifted exponential** model and **z colluding workers,** z<k
  - **Concentration bound** on the number of responses needed
  - Hiding the **attributes**

- Future directions
  - Secure distributed computing for **non-linear** computation
  - Closing the **gap** between theory and practice
  - **Heterogeneous systems**: load balancing based on prior knowledge
  - Construction of **secure rateless** codes
  - Presence of **malicious** workers: codes and bounds on system latency


THANK YOU
Merci
Obrigado
Takk
Tack
Grazie
Danke
Kiitos
Gracias
Efharisto
Tak

Questions?

Example on Staircase Codes



Worker 1	Worker 2	Worker 3	Worker 4
$A_1+A_2+A_3+R_1$	$A_1+2A_2+4A_3+3R_1$	$A_1+3A_2+4A_3+2R_1$	$A_1+4A_2+A_3+4R_1$
$A_4+A_5+A_6+R_2$	$A_4+2A_5+4A_6+3R_2$	$A_4+3A_5+4A_6+2R_2$	$A_4+4A_5+A_6+4R_2$
$R_1+R_2+R_3$	$R_1+2R_2+4R_3$	$R_1+3R_2+4R_3$	$R_1+4R_2+R_3$
A_3+R_4	A_3+2R_4	A_3+3R_4	A_3+4R_4
A_6+R_5	A_6+2R_5	A_6+3R_5	A_6+4R_5
R_3+R_6	R_3+2R_6	R_3+3R_6	R_3+4R_6