GASP Codes for Secure Distributed Matrix Multiplication

Rafael G.L. D'Oliveira
Salim El Rouayheb
David Karpuk

Rutgers University
and
Universidad de los Andes

Setting

- **User**
- **Servers**

Setting

- **User**
  - Has two matrices $A \in \mathbb{F}_q^{r \times s}$ and $B \in \mathbb{F}_q^{s \times t}$.
  - Wants the product $AB \in \mathbb{F}_q^{r \times t}$
- **Servers**

Setting

- **User**
  - Has two matrices $A \in \mathbb{F}_q^{r \times s}$ and $B \in \mathbb{F}_q^{s \times t}$.
  - Wants the product $AB \in \mathbb{F}_q^{r \times t}$
- **Servers**
  - We denote the number of servers by N.
  - Each receives two matrices and outputs their product.
  - Honest but curious.

Setting

- **User**
  - Has two matrices $A \in \mathbb{F}_q^{r \times s}$ and $B \in \mathbb{F}_q^{s \times t}$.
  - Wants the product $AB \in \mathbb{F}_q^{r \times t}$
- **Servers**
  - We denote the number of servers by N.
  - Each receives two matrices and outputs their product.
  - Honest but curious.
- **Goal:**
  - Use servers to compute AB.
  - Reveal no information about A or B to any server.
  - At most T servers collude.
  - Minimize communication costs.

Polynomial Codes: Simplest Example

- The user has two matrices $A \in \mathbb{F}_q^{r \times s}$ and $B \in \mathbb{F}_q^{s \times t}$.

Polynomial Codes: Simplest Example

- The user has two matrices $A \in \mathbb{F}_q^{r\times s}$ and $B \in \mathbb{F}_q^{s\times t}$.
  - Generate random matrices $R \in \mathbb{F}_q^{r\times s}$ and $S \in \mathbb{F}_q^{s\times t}$.
  - Generate polynomials

$$f(x) = A + Rx \quad \text{and} \quad g(x) = B + Sx.$$

Polynomial Codes: Simplest Example

- The user has two matrices $A \in \mathbb{F}_q^{r \times s}$ and $B \in \mathbb{F}_q^{s \times t}$.
  - Generate random matrices $R \in \mathbb{F}_q^{r \times s}$ and $S \in \mathbb{F}_q^{s \times t}$.
  - Generate polynomials

$$f(x) = A + Rx \quad \text{and} \quad g(x) = B + Sx.$$

  - Let $h(x) = f(x).g(x)$. Then,

$$h(x) = AB + (AS + RB)x + RSx^2.$$

Polynomial Codes: Simplest Example

- The user has two matrices $A \in \mathbb{F}_q^{r \times s}$ and $B \in \mathbb{F}_q^{s \times t}$.
  - Generate random matrices $R \in \mathbb{F}_q^{r \times s}$ and $S \in \mathbb{F}_q^{s \times t}$.
  - Generate polynomials

$$f(x) = A + Rx \quad \text{and} \quad g(x) = B + Sx.$$

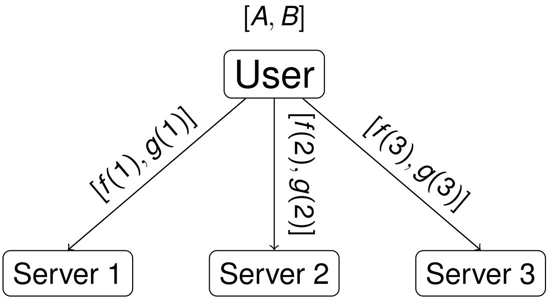
  - Let $h(x) = f(x).g(x)$. Then,

$$h(x) = AB + (AS + RB)x + RSx^2.$$

  - With 3 evaluations of h, we can reconstruct h and compute

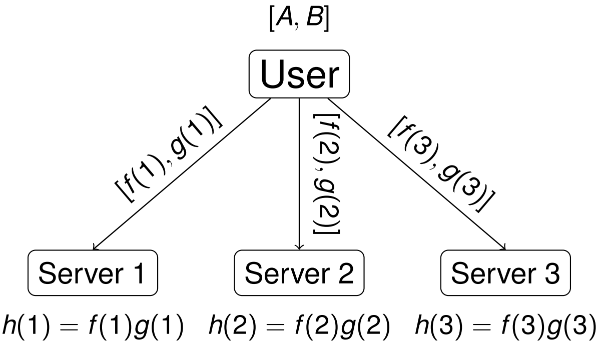
$$h(0) = AB.$$

Polynomial Codes: Simplest Example



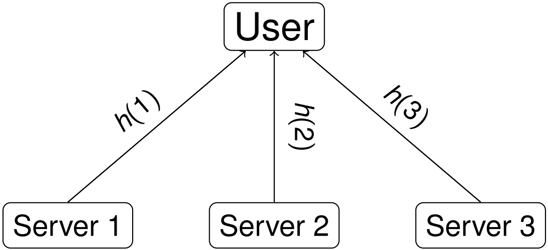
- $f(x) = A + Rx$.
- $g(x) = B + Sx$.
- $h(x) = f(x).g(x) = AB + (AS + RB)x + RSx^2$.

Polynomial Codes: Simplest Example



- $f(x) = A + Rx$.
- $g(x) = B + Sx$.
- $h(x) = f(x).g(x) = AB + (AS + RB)x + RSx^2$.

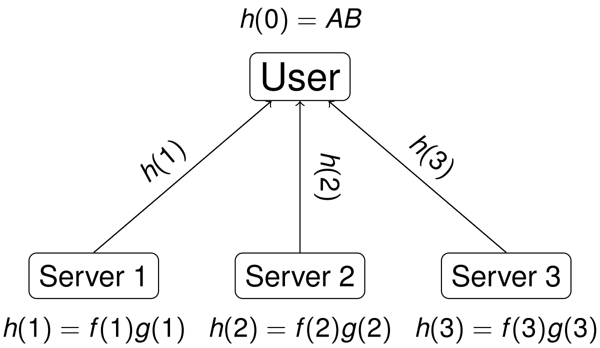
Polynomial Codes: Simplest Example



$h(1) = f(1)g(1)$ $h(2) = f(2)g(2)$ $h(3) = f(3)g(3)$

- $f(x) = A + Rx$.
- $g(x) = B + Sx$.
- $h(x) = f(x).g(x) = AB + (AS + RB)x + RSx^2$.

Polynomial Codes: Simplest Example



- $f(x) = A + Rx$.
- $g(x) = B + Sx$.
- $h(x) = f(x).g(x) = AB + (AS + RB)x + RSx^2$.

Partitioning the Matrices

- If $A \in \mathbb{F}_q^{r \times s}$ and $B \in \mathbb{F}_q^{s \times t}$, computing AB takes $O(rst)$ operations.

Partitioning the Matrices

- If $A \in \mathbb{F}_q^{r \times s}$ and $B \in \mathbb{F}_q^{s \times t}$, computing AB takes $O(rst)$ operations.
- Consider the following matrix partitioning.

$$A = \begin{bmatrix} A_1 \\ \vdots \\ A_K \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} B_1 & \cdots & B_L \end{bmatrix}.$$

Partitioning the Matrices

- If $A \in \mathbb{F}_q^{r \times s}$ and $B \in \mathbb{F}_q^{s \times t}$, computing AB takes $O(rst)$ operations.
- Consider the following matrix partitioning.

$$A = \begin{bmatrix} A_1 \\ \vdots \\ A_K \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} B_1 & \cdots & B_L \end{bmatrix}.$$

- The product AB is given by

$$AB = \begin{bmatrix} A_1 B_1 & \cdots & A_1 B_L \\ \vdots & \ddots & \vdots \\ A_K B_1 & \cdots & A_K B_L \end{bmatrix}$$

Partitioning the Matrices

- If $A \in \mathbb{F}_q^{r\times s}$ and $B \in \mathbb{F}_q^{s\times t}$, computing AB takes $O(rst)$ operations.
- Consider the following matrix partitioning.

$$A = \begin{bmatrix} A_1 \\ \vdots \\ A_K \end{bmatrix} \quad \text{and} \quad B = \begin{bmatrix} B_1 & \cdots & B_L \end{bmatrix}.$$

- The product AB is given by

$$AB = \begin{bmatrix} A_1 B_1 & \cdots & A_1 B_L \\ \vdots & \ddots & \vdots \\ A_K B_1 & \cdots & A_K B_L \end{bmatrix}$$

- Computing $A_i B_i$ takes $O(\frac{rst}{KL})$ operations.

Previous Work: Polynomial Codes for Stragglers

- Originally introduced in [Yu, Maddah-Ali, Avestimehr, ’17].
- Different Setting: mitigating stragglers
- Other Work: [Yu, Maddah-Ali, Avestimehr, ’18] ,
  [Dutta, Fahim, Haddadpour, Jeong, Cadambe, Grove, ’18],
  [Sheth, Dutta, Chaudhari, Jeong, Yang, Kohonen, Roos, Grove, ’18],
  [Li, Maddah-Ali, Yu, Avestimehr, ’18],
  etc.

Previous Work: Polynomial Codes for Security

- Three works consider secure distributed multiplication from the information theoretic point of view.

Previous Work: Polynomial Codes for Security

- Three works consider secure distributed multiplication from the information theoretic point of view.
- [Chang, Tandon, '18]: presents a scheme for $K = L$ with download rate

$$\mathcal{R} = \frac{K^2}{(K+T)^2}$$

Previous Work: Polynomial Codes for Security

- Three works consider secure distributed multiplication from the information theoretic point of view.
- [Chang, Tandon, '18]: presents a scheme for $K = L$ with download rate

$$\mathcal{R} = \frac{K^2}{(K+T)^2}$$

- [Kakar, Ebadifar, Sezgin, '18]: presents a scheme with download rate

$$\mathcal{R} = \frac{KL}{(K+T)(L+1)-1}$$

Previous Work: Polynomial Codes for Security

- Three works consider secure distributed multiplication from the information theoretic point of view.
- [Chang, Tandon, '18]: presents a scheme for $K = L$ with download rate
$$\mathcal{R} = \frac{K^2}{(K+T)^2}$$
- [Kakar, Ebadifar, Sezgin, '18]: presents a scheme with download rate
$$\mathcal{R} = \frac{KL}{(K+T)(L+1)-1}$$
- [Yang, Lee, '19]: presents a scheme for $T = 1$ with download rate
$$\mathcal{R} = \frac{KL}{KL+K+L}$$

Gap Additive Secure Polynomial Codes

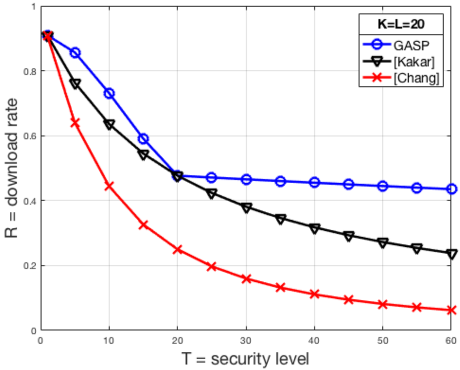
- We present “GASP Codes for Secure Distributed Matrix Multiplication”, soon on Arxiv.

Gap Additive Secure Polynomial Codes

- We present "GASP Codes for Secure Distributed Matrix Multiplication", soon on Arxiv.

Theorem

GASP codes outperform all previous schemes in terms of communication cost.



Polynomial Code with $K = L = 3$ and $T = 2$.

- Partition A and B as follows.

$$A = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix}, \quad B = \begin{bmatrix} B_1 & B_2 & B_3 \end{bmatrix}$$

- The product AB is given by

$$AB = \begin{bmatrix} A_1 B_1 & A_1 B_2 & A_1 B_3 \\ A_2 B_1 & A_2 B_2 & A_2 B_3 \\ A_3 B_1 & A_3 B_2 & A_3 B_3 \end{bmatrix}$$

Can't Choose Any Polynomial

- $f(x) = A_1 + A_2x + A_3x^2 + R_1x^3 + R_2x^4$
- $g(x) = B_1 + B_2x + B_3x^2 + S_1x^3 + S_2x^4$
- Let $h(x) = f(x)g(x)$. Then,

  $$h(x) = A_1B_1 + (A_1B_2 + A_2B_1)x + (A_1B_3 + A_2B_2 + A_3B_1)x^2 + \ldots$$

- Can't retrieve A_1B_2, for example.

Polynomial Code with $K = L = 3$ and $T = 2$.

- Partition A and B as follows.

$$A = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix}, \quad B = \begin{bmatrix} B_1 & B_2 & B_3 \end{bmatrix}$$

- The product AB is given by

$$AB = \begin{bmatrix} A_1B_1 & A_1B_2 & A_1B_3 \\ A_2B_1 & A_2B_2 & A_2B_3 \\ A_3B_1 & A_3B_2 & A_3B_3 \end{bmatrix}$$

Polynomial Code with $K = L = 3$ and $T = 2$.

- Partition A and B as follows.

$$A = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix}, \quad B = \begin{bmatrix} B_1 & B_2 & B_3 \end{bmatrix}$$

- The product AB is given by

$$AB = \begin{bmatrix} A_1B_1 & A_1B_2 & A_1B_3 \\ A_2B_1 & A_2B_2 & A_2B_3 \\ A_3B_1 & A_3B_2 & A_3B_3 \end{bmatrix}$$

- $f(x) = A_1 + A_2x + A_3x^2 + R_1x^3 + R_2x^4$
- $g(x) = B_1 + B_2x^5 + B_3x^{10} + S_1x^{13} + S_2x^{14}$
- Then, A_iB_j appear in distinct terms of $h = fg$.

Polynomial Code with $K = L = 3$ and $T = 2$.

- Partition A and B as follows.

$$A = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix}, \quad B = \begin{bmatrix} B_1 & B_2 & B_3 \end{bmatrix}$$

- The product AB is given by

$$AB = \begin{bmatrix} A_1 B_1 & A_1 B_2 & A_1 B_3 \\ A_2 B_1 & A_2 B_2 & A_2 B_3 \\ A_3 B_1 & A_3 B_2 & A_3 B_3 \end{bmatrix}$$

- $f(x) = A_1 + A_2 x + A_3 x^2 + R_1 x^3 + R_2 x^4$
- $g(x) = B_1 + B_2 x^5 + B_3 x^{10} + S_1 x^{13} + S_2 x^{14}$
- Then, $A_i B_j$ appear in distinct terms of $h = fg$.
- We need $N = \deg h + 1 = 19$ servers.

It is not about the degree.

- **Previously:**
  - $f(x) = A_1 + A_2 x + A_3 x^2 + R_1 x^3 + R_2 x^4$
  - $g(x) = B_1 + B_2 x^5 + B_3 x^{10} + S_1 x^{13} + S_2 x^{14}$
  - $N_h = \deg h + 1 = 19$ servers.

It is not about the degree.

- **Previously:**
  - $f(x) = A_1 + A_2 x + A_3 x^2 + R_1 x^3 + R_2 x^4$
  - $g(x) = B_1 + B_2 x^5 + B_3 x^{10} + S_1 x^{13} + S_2 x^{14}$
  - $N_h = \deg h + 1 = 19$ servers.
- **Consider:**
  - $f^*(x) = A_1 + A_2 x + A_3 x^2 + R_1 x^9 + R_2 x^{12}$
  - $g^*(x) = B_1 + B_2 x^3 + B_3 x^6 + S_1 x^9 + S_2 x^{10}$
- $\deg h^* = 22$

It is not about the degree.

- **Previously:**
  - $f(x) = A_1 + A_2x + A_3x^2 + R_1x^3 + R_2x^4$
  - $g(x) = B_1 + B_2x^5 + B_3x^{10} + S_1x^{13} + S_2x^{14}$
  - $N_h = \deg h + 1 = 19$ servers.
- **Consider:**
  - $f^*(x) = A_1 + A_2x + A_3x^2 + R_1x^9 + R_2x^{12}$
  - $g^*(x) = B_1 + B_2x^3 + B_3x^6 + S_1x^9 + S_2x^{10}$
- $\deg h^* = 22 > 18 = \deg h$

It is not about the degree.

- **Previously:**
  - $f(x) = A_1 + A_2x + A_3x^2 + R_1x^3 + R_2x^4$
  - $g(x) = B_1 + B_2x^5 + B_3x^{10} + S_1x^{13} + S_2x^{14}$
  - $N_h = \deg h + 1 = 19$ servers.
- **Consider:**
  - $f^*(x) = A_1 + A_2x + A_3x^2 + R_1x^9 + R_2x^{12}$
  - $g^*(x) = B_1 + B_2x^3 + B_3x^6 + S_1x^9 + S_2x^{10}$
- $\deg h^* = 22 > 18 = \deg h$
- But h^* has gaps in the degrees.
- No term of degrees 13, 14, 16, 17 or 20.

It is not about the degree.

- **Previously:**
  - $f(x) = A_1 + A_2x + A_3x^2 + R_1x^3 + R_2x^4$
  - $g(x) = B_1 + B_2x^5 + B_3x^{10} + S_1x^{13} + S_2x^{14}$
  - $N_h = \deg h + 1 = 19$ servers.
- **Consider:**
  - $f^*(x) = A_1 + A_2x + A_3x^2 + R_1x^9 + R_2x^{12}$
  - $g^*(x) = B_1 + B_2x^3 + B_3x^6 + S_1x^9 + S_2x^{10}$
- $\deg h^* = 22 > 18 = \deg h$
- But h^* has gaps in the degrees.
- No term of degrees 13, 14, 16, 17 or 20.
- Thus, only 18 points needed to interpolate h^*.
- $N_{h^*} = 18 < 19 = N_h$.

What is it about?

- It is about the number of terms in the polynomial.

What is it about?

- It is about the number of terms in the polynomial.
- Consider the polynomial $f(x) = ax^6 + bx^5 + cx$.
- We need $3 < \deg f + 1$ points to interpolate this polynomial.

What is it about?

- It is about the number of terms in the polynomial.
- Consider the polynomial $f(x) = ax^6 + bx^5 + cx$.
- We need $3 < \deg f + 1$ points to interpolate this polynomial.
- Not any points! What does $f(0)$ tell you?

How many terms does $f(x)g(x)$ have?

- $f(x) = A_1 x^{\alpha_1} + A_2 x^{\alpha_2} + A_3 x^{\alpha_3}$
- $g(x) = B_1 x^{\beta_1} + B_2 x^{\beta_2} + B_3 x^{\beta_3}$

Then $h(x) = f(x)g(x)$ will be

$$\begin{aligned} h(x) = {} & A_1 B_1 x^{\alpha_1+\beta_1} + A_1 B_2 x^{\alpha_1+\beta_2} + A_1 B_3 x^{\alpha_1+\beta_3} \\ & + A_2 B_1 x^{\alpha_2+\beta_1} + A_2 B_2 x^{\alpha_2+\beta_2} + A_2 B_3 x^{\alpha_2+\beta_3} \\ & + A_3 B_1 x^{\alpha_3+\beta_1} + A_3 B_2 x^{\alpha_3+\beta_2} + A_3 B_3 x^{\alpha_3+\beta_3} \end{aligned}$$

The Degree Table

We begin with an example.

- $f(x) = A_1 x^{\alpha_1} + A_2 x^{\alpha_2} + A_3 x^{\alpha_3}$
- $g(x) = B_1 x^{\beta_1} + B_2 x^{\beta_2} + B_3 x^{\beta_3}$

Then terms in h appear in the following table.

	β_1	β_2	β_3
α_1	$\alpha_1 + \beta_1$	$\alpha_1 + \beta_2$	$\alpha_1 + \beta_3$
α_2	$\alpha_2 + \beta_1$	$\alpha_2 + \beta_2$	$\alpha_2 + \beta_3$
α_3	$\alpha_3 + \beta_1$	$\alpha_3 + \beta_2$	$\alpha_3 + \beta_3$

The Degree Table

We begin with an example.

- $f(x) = A_1 x^{\alpha_1} + A_2 x^{\alpha_2} + A_3 x^{\alpha_3}$
- $g(x) = B_1 x^{\beta_1} + B_2 x^{\beta_2} + B_3 x^{\beta_3}$

Then terms in h appear in the following table.

	β_1	β_2	β_3
α_1	$\alpha_1 + \beta_1$	$\alpha_1 + \beta_2$	$\alpha_1 + \beta_3$
α_2	$\alpha_2 + \beta_1$	$\alpha_2 + \beta_2$	$\alpha_2 + \beta_3$
α_3	$\alpha_3 + \beta_1$	$\alpha_3 + \beta_2$	$\alpha_3 + \beta_3$

- We call this a degree table.

Revisiting the Previous Examples

- **Previously:**
  - $f(x) = A_1 + A_2x^1 + A_3x^2 + R_1x^3 + R_2x^4$
  - $g(x) = B_1 + B_2x^5 + B_3x^{10} + S_1x^{13} + S_2x^{14}$

h	0	5	10	13	14
0	0	5	10	13	14
1	1	6	11	14	15
2	2	7	12	15	16
3	3	8	13	16	17
4	4	9	14	17	18

Revisiting the Previous Examples

- **Previously:**

h	0	5	10	13	14
0	0	5	10	13	14
1	1	6	11	14	15
2	2	7	12	15	16
3	3	8	13	16	17
4	4	9	14	17	18

- **Consider:**
  - $f^*(x) = A_1 + A_2x^1 + A_3x^2 + R_1x^9 + R_2x^{12}$
  - $g^*(x) = B_1 + B_2x^3 + B_3x^6 + S_1x^9 + S_2x^{10}$

h^*	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
12	10	15	18	21	22

Why do They Work?

h	0	5	10	13	14
0	0	5	10	13	14
1	1	6	11	14	15
2	2	7	12	15	16
3	3	8	13	16	17
4	4	9	14	17	18

h^*	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
12	10	15	18	21	22

- Decodability: Red cells unique.

Why do They Work?

h	0	5	10	13	14
0	0	5	10	13	14
1	1	6	11	14	15
2	2	7	12	15	16
3	3	8	13	16	17
4	4	9	14	17	18

h^*	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
12	10	15	18	21	22

- Decodability: Red cells unique.
- Security A: Green cells distinct.

Why do They Work?

h	0	5	10	13	14
0	0	5	10	13	14
1	1	6	11	14	15
2	2	7	12	15	16
3	3	8	13	16	17
4	4	9	14	17	18

h^*	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
12	10	15	18	21	22

- Decodability: Red cells unique.
- Security A: Green cells distinct.
- Security B: Blue cells distinct.

Why do They Work?

h	0	5	10	13	14
0	0	5	10	13	14
1	1	6	11	14	15
2	2	7	12	15	16
3	3	8	13	16	17
4	4	9	14	17	18

h^*	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
12	10	15	18	21	22

- Decodability: Red cells unique.
- Security A: Green cells distinct.
- Security B: Blue cells distinct.

- **Goal:** Minimize distinct cells.

How Many Terms?

h	0	5	10	13	14
0	0	5	10	13	14
1	1	6	11	14	15
2	2	7	12	15	16
3	3	8	13	16	17
4	4	9	14	17	18

- $|\textsf{terms}\, h| =$ Purple Area $= 19$

How Many Terms?

h	0	5	10	13	14
0	0	5	10	13	14
1	1	6	11	14	15
2	2	7	12	15	16
3	3	8	13	16	17
4	4	9	14	17	18

- $|\text{terms } h| = \text{Purple Area} = 19$

h^*	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
12	10	15	18	21	22

- $|\text{terms } h^*| = \text{Purple Area} = 18$

Problem Restatement: The Degree Table

	β_1	$\cdots$	β_L	β_{L+1}	$\cdots$	β_{L+T}
α_1	$\alpha_1 + \beta_1$	$\cdots$	$\alpha_1 + \beta_L$	$\alpha_1 + \beta_{L+1}$	$\cdots$	$\alpha_1 + \beta_{L+T}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
α_K	$\alpha_K + \beta_1$	$\cdots$	$\alpha_K + \beta_L$	$\alpha_K + \beta_{L+1}$	$\cdots$	$\alpha_K + \beta_{L+T}$
α_{K+1}	$\alpha_{K+1} + \beta_1$	$\cdots$	$\alpha_{K+1} + \beta_L$	$\alpha_{K+1} + \beta_{L+1}$	$\cdots$	$\alpha_{K+1} + \beta_{L+T}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
α_{K+T}	$\alpha_{K+T} + \beta_1$	$\cdots$	$\alpha_{K+T} + \beta_L$	$\alpha_{K+T} + \beta_{L+1}$	$\cdots$	$\alpha_{K+T} + \beta_{L+T}$

- **Goal:** Minimize number of distinct terms.
- Subject to:
  - Decodability: Numbers in the red region are all unique.
  - *A*-Security: Numbers in the green region are all distinct.
  - *B*-Security: Numbers in the blue region are all distinct.

Finding Solutions

- Consider $K = L = 3$ and $T = 1$

	0	3	6	
0	0	3	6	
1	1	4	7	
2	2	5	8	

Finding Solutions

- Consider $K = L = 3$ and $T = 1$

	0	3	6	9
0	0	3	6	
1	1	4	7	
2	2	5	8	
9				

Finding Solutions

- Consider $K = L = 3$ and $T = 1$

	0	3	6	9
0	0	3	6	9
1	1	4	7	10
2	2	5	8	11
9	9	12	15	18

Finding Solutions

- Consider $K = L = 3$ and $T = 1$

	0	3	6	9
0	0	3	6	9
1	1	4	7	10
2	2	5	8	11
9	9	12	15	18

- $N = 15$.

Finding Solutions

- Consider $K = L = 3$ and $T = 2$

	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
10	10	13	16	19	20

- $N = 19$.

Finding Solutions

- Consider $K = L = 3$ and $T = 3$

	0	3	6	9	10	11
0	0	3	6	9	10	11
1	1	4	7	10	11	12
2	2	5	8	11	12	13
9	9	12	15	18	19	20
10	10	13	16	19	20	21
11	11	14	17	20	21	22

- $N = 23$.

GASP$_{\text{big}}$ (Gap Additive Secure Polynomial)

- For $L \leq K$, GASP$_{\text{big}}$ is the following scheme.

	$\beta_1 = 0$	$\cdots$	$\beta_L = K(L-1)$	$\beta_{L+1} = KL$	$\beta_{L+2} = KL+1$	$\cdots$	$\beta_{L+T} = KL+T-1$
$\alpha_1 = 0$	0	$\cdots$	$K(L-1)$	KL	$KL+1$	$\cdots$	$KL+T-1$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\alpha_K = K-1$	$K-1$	$\cdots$	$KL-1$	$KL+K-1$	$KL+K$	$\cdots$	$KL+K+T-2$
$\alpha_{K+1} = KL$	KL	$\cdots$	$2KL-K$	$2KL$	$2KL+1$	$\cdots$	$2KL+T-1$
$\alpha_{K+2} = KL+1$	$KL+1$	$\cdots$	$2KL-K+1$	$2KL+1$	$2KL+2$	$\cdots$	$2KL+T$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\alpha_{K+T} = KL+T-1$	$KL+T-1$	$\cdots$	$2KL-K+T-1$	$2KL+T-1$	$2KL+T$	$\cdots$	$2KL+2T-2$

- For $K < L$, permute α and β.

$T = 1$	0	3	6	9
0	0	3	6	9
1	1	4	7	10
2	2	5	8	11
9	9	12	15	18

$T=1$	0	3	6	9
0	0	3	6	9
1	1	4	7	10
2	2	5	8	11
9	9	12	15	18

$T=2$	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
10	10	13	16	19	20

$T=1$	0	3	6	9
0	0	3	6	9
1	1	4	7	10
2	2	5	8	11
9	9	12	15	18

$T=2$	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
10	10	13	16	19	20

$T=3$	0	3	6	9	10	11
0	0	3	6	9	10	11
1	1	4	7	10	11	12
2	2	5	8	11	12	13
9	9	12	15	18	19	20
10	10	13	16	19	20	21
11	11	14	17	20	21	22

$T=1$	0	3	6	9
0	0	3	6	9
1	1	4	7	10
2	2	5	8	11
9	9	12	15	18

$T=2$	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
10	10	13	16	19	20

$T=3$	0	3	6	9	10	11
0	0	3	6	9	10	11
1	1	4	7	10	11	12
2	2	5	8	11	12	13
9	9	12	15	18	19	20
10	10	13	16	19	20	21
11	11	14	17	20	21	22

$T=4$	0	3	6	9	10	11	12
0	0	3	6	9	10	11	12
1	1	4	7	10	11	12	13
2	2	5	8	11	12	13	14
9	9	12	15	18	19	20	21
10	10	13	16	19	20	21	22
11	11	14	17	20	21	22	23
12	12	15	18	21	22	23	24

$T = 1$	0	3	6	9
0	0	3	6	9
1	1	4	7	10
2	2	5	8	11
9	9	12	15	18

$T = 2$	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
10	10	13	16	19	20

$T = 3$	0	3	6	9	10	11
0	0	3	6	9	10	11
1	1	4	7	10	11	12
2	2	5	8	11	12	13
9	9	12	15	18	19	20
10	10	13	16	19	20	21
11	11	14	17	20	21	22

$T = 4$	0	3	6	9	10	11	12
0	0	3	6	9	10	11	12
1	1	4	7	10	11	12	13
2	2	5	8	11	12	13	14
9	9	12	15	18	19	20	21
10	10	13	16	19	20	21	22
11	11	14	17	20	21	22	23
12	12	15	18	21	22	23	24

$T = 5$	0	3	6	9	10	11	12	13
0	0	3	6	9	10	11	12	13
1	1	4	7	10	11	12	13	14
2	2	5	8	11	12	13	14	15
9	9	12	15	18	19	20	21	22
10	10	13	16	19	20	21	22	23
11	11	14	17	20	21	22	23	24
12	12	15	18	21	22	23	24	25
13	13	16	19	22	23	24	25	26

$T=1$	0	3	6	9
0	0	3	6	9
1	1	4	7	10
2	2	5	8	11
9	9	12	15	18

$T=2$	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
10	10	13	16	19	20

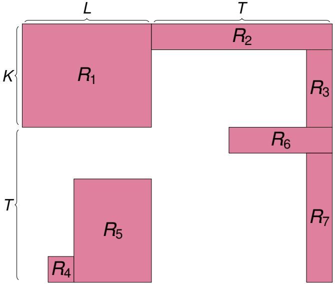
$T=3$	0	3	6	9	10	11
0	0	3	6	9	10	11
1	1	4	7	10	11	12
2	2	5	8	11	12	13
9	9	12	15	18	19	20
10	10	13	16	19	20	21
11	11	14	17	20	21	22

$T=4$	0	3	6	9	10	11	12
0	0	3	6	9	10	11	12
1	1	4	7	10	11	12	13
2	2	5	8	11	12	13	14
9	9	12	15	18	19	20	21
10	10	13	16	19	20	21	22
11	11	14	17	20	21	22	23
12	12	15	18	21	22	23	24

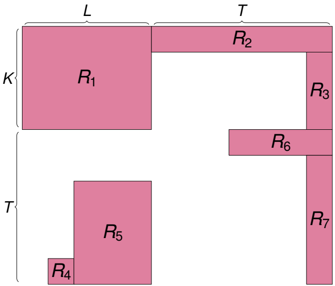
$T=5$	0	3	6	9	10	11	12	13
0	0	3	6	9	10	11	12	13
1	1	4	7	10	11	12	13	14
2	2	5	8	11	12	13	14	15
9	9	12	15	18	19	20	21	22
10	10	13	16	19	20	21	22	23
11	11	14	17	20	21	22	23	24
12	12	15	18	21	22	23	24	25
13	13	16	19	22	23	24	25	26

$T=6$	0	3	6	9	10	11	12	13	14
0	0	3	6	9	10	11	12	13	14
1	1	4	7	10	11	12	13	14	15
2	2	5	8	11	12	13	14	15	16
9	9	12	15	18	19	20	21	22	23
10	10	13	16	19	20	21	22	23	24
11	11	14	17	20	21	22	23	24	25
12	12	15	18	21	22	23	24	25	26
13	13	16	19	22	23	24	25	26	27
14	14	17	20	23	24	25	26	27	28

Computing the Number of Terms ($L \leq K$)

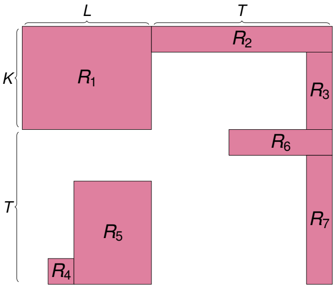


Computing the Number of Terms ($L \leq K$)



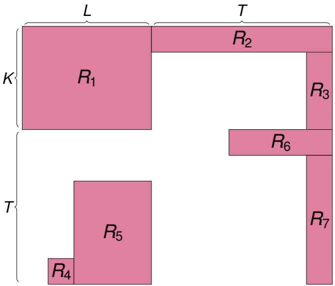
- $R_1 = K \times L$

Computing the Number of Terms ($L \leq K$)



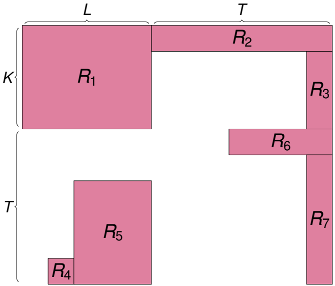
- $R_1 = K \times L$
- $R_2 = 1 \times T$

Computing the Number of Terms ($L \leq K$)



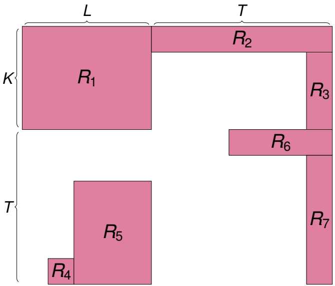
- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$

Computing the Number of Terms ($L \leq K$)



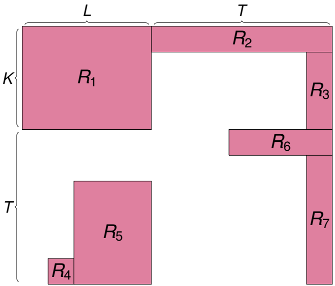
- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$
- $R_4 = 1 \times 1$ if $L \geq 2$

Computing the Number of Terms ($L \leq K$)



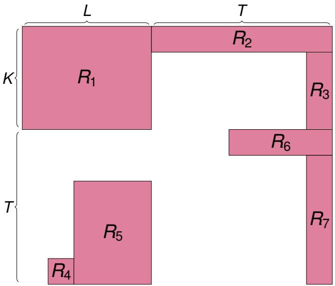
- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$
- $R_4 = 1 \times 1$ if $L \geq 2$
- $R_5 = \min\{T, K\} \times \max\{(L-2), 0\}$

Computing the Number of Terms ($L \leq K$)



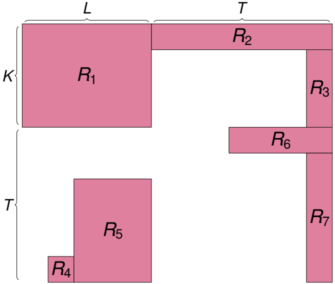
- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K-1) \times 1$
- $R_4 = 1 \times 1$ if $L \geq 2$
- $R_5 = \min\{T, K\} \times \max\{(L-2), 0\}$
- $R_6 = 1 \times \min\{T, K\}$

Computing the Number of Terms ($L \leq K$)



- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$
- $R_4 = 1 \times 1$ if $L \geq 2$
- $R_5 = \min\{T, K\} \times \max\{(L-2), 0\}$
- $R_6 = 1 \times \min\{T, K\}$
- $R_7 = (T - 1) \times 1$

Computing the Number of Terms ($L \leq K$)



- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$
- $R_4 = 1 \times 1$ if $L \geq 2$
- $R_5 = \min\{T, K\} \times \max\{(L-2), 0\}$
- $R_6 = 1 \times \min\{T, K\}$
- $R_7 = (T - 1) \times 1$

Theorem

$N = |\text{terms in GASP}_{\text{big}}| = R_1 + \ldots + R_7$.

Number of Terms

Theorem

The number of terms in GASP_{big}, for $L \leq K$, is

$$N = \begin{cases} 2K + T & \text{if } L = 1, T < K \\ K + 2T & \text{if } L = 1, T \geq K \\ (K + T)(L + 1) - 1 & \text{if } L \geq 2, T < K \\ 2KL + 2T - 1 & \text{if } L \geq 2, T \geq K \end{cases}$$

Number of Terms

Theorem

The number of terms in GASP_{big}, for $L \leq K$, is

$$N = \begin{cases} 2K + T & \text{if } L = 1, T < K \\ K + 2T & \text{if } L = 1, T \geq K \\ (K+T)(L+1) - 1 & \text{if } L \geq 2, T < K \\ 2KL + 2T - 1 & \text{if } L \geq 2, T \geq K \end{cases}$$

- For big T, $N = 2KL + 2T - 1$.

Number of Terms

Theorem

The number of terms in $\mathrm{GASP}_{\mathrm{big}}$, for $L \leq K$, is

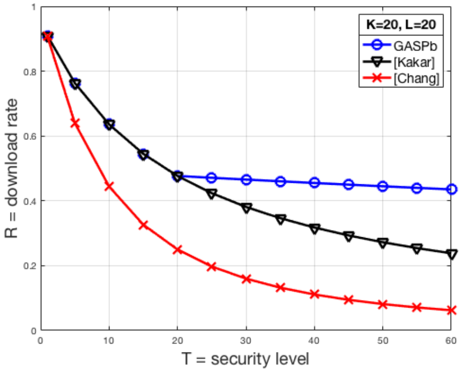
$$N = \begin{cases} 2K + T & \text{if } L = 1, T < K \\ K + 2T & \text{if } L = 1, T \geq K \\ (K + T)(L + 1) - 1 & \text{if } L \geq 2, T < K \\ 2KL + 2T - 1 & \text{if } L \geq 2, T \geq K \end{cases}$$

- For big T, $N = 2KL + 2T - 1$.
- The download rate is $\mathcal{R} = KL/N$.

How good is GASP_{big}?

Theorem

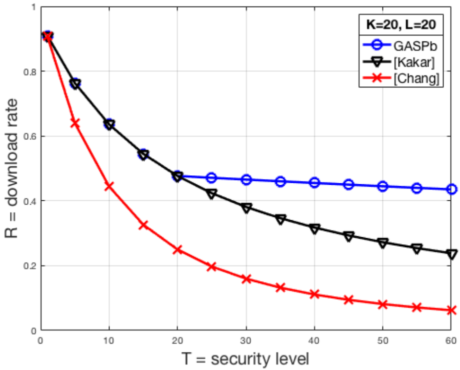
GASP_{big} outperforms all previous schemes for all parameters.



How good is GASP$_{\text{big}}$?

Theorem

GASP$_{\text{big}}$ outperforms all previous schemes for all parameters.



- Can we do better?

GASP$_{\text{small}}$

- For $K \leq L$, GASP$_{\text{small}}$ is the following scheme.

	$\beta_1 = 0$	$\cdots$	$\beta_L = K(L-1)$	$\beta_{L+1} = KL$	$\beta_{L+2} = KL+1$	$\cdots$	$\beta_{L+T} = KL+T-1$
$\alpha_1 = 0$	0	$\cdots$	$K(L-1)$	KL	$KL+1$	$\cdots$	$KL+T-1$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\alpha_K = K-1$	$K-1$	$\cdots$	$KL-1$	$KL+K-1$	$KL+K$	$\cdots$	$KL+K+T-2$
$\alpha_{K+1} = KL$	KL	$\cdots$	$2KL-K$	$2KL$	$2KL+1$	$\cdots$	$2KL+T-1$
$\alpha_{K+2} = KL+K$	$KL+K$	$\cdots$	$2KL$	$2KL+K$	$2KL+K+1$	$\cdots$	$2KL+K+T-1$
$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$\alpha_{K+T} = KL+K(T-1)$	$KL+K(T-1)$	$\cdots$	$2KL+K(T-2)$	$2KL+K(T-1)$	$2KL+K(T-1)+1$	$\cdots$	$2KL+(K+1)(T-1)$

$T = 1$	0	3	6	9
0	0	3	6	9
1	1	4	7	10
2	2	5	8	11
9	9	12	15	18

$T = 2$	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
12	12	15	18	21	22

$T = 3$	0	3	6	9	10	11
0	0	3	6	9	10	11
1	1	4	7	10	11	12
2	2	5	8	11	12	13
9	9	12	15	18	19	20
12	12	15	18	21	22	23
15	15	18	21	24	25	26

$T = 4$	0	3	6	9	10	11	12
0	0	3	6	9	10	11	12
1	1	4	7	10	11	12	13
2	2	5	8	11	12	13	14
9	9	12	15	18	19	20	21
12	12	15	18	21	22	23	24
15	15	18	21	24	25	26	27
18	18	21	24	27	28	29	30

$T = 5$	0	3	6	9	10	11	12	13
0	0	3	6	9	10	11	12	13
1	1	4	7	10	11	12	13	14
2	2	5	8	11	12	13	14	15
9	9	12	15	18	19	20	21	22
12	12	15	18	21	22	23	24	25
15	15	18	21	24	25	26	27	28
18	18	21	24	27	28	29	30	31
21	21	24	27	30	31	32	33	34

$T = 6$	0	3	6	9	10	11	12	13	14
0	0	3	6	9	10	11	12	13	14
1	1	4	7	10	11	12	13	14	15
2	2	5	8	11	12	13	14	15	16
9	9	12	15	18	19	20	21	22	23
12	12	15	18	21	22	23	24	25	26
15	15	18	21	24	25	26	27	28	29
18	18	21	24	27	28	29	30	31	32
21	21	24	27	30	31	32	33	34	35
24	24	27	30	33	34	35	36	37	38

$T=3$	0	3	6	9	10	11
0	0	3	6	9	10	11
1	1	4	7	10	11	12
2	2	5	8	11	12	13
9	9	12	15	18	19	20
12	12	15	18	21	22	23
15	15	18	21	24	25	26

$T=4$	0	3	6	9	10	11	12
0	0	3	6	9	10	11	12
1	1	4	7	10	11	12	13
2	2	5	8	11	12	13	14
9	9	12	15	18	19	20	21
12	12	15	18	21	22	23	24
15	15	18	21	24	25	26	27
18	18	21	24	27	28	29	30

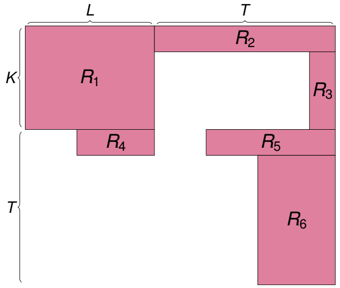
$T=5$	0	3	6	9	10	11	12	13
0	0	3	6	9	10	11	12	13
1	1	4	7	10	11	12	13	14
2	2	5	8	11	12	13	14	15
9	9	12	15	18	19	20	21	22
12	12	15	18	21	22	23	24	25
15	15	18	21	24	25	26	27	28
18	18	21	24	27	28	29	30	31
21	21	24	27	30	31	32	33	34

$T=6$	0	3	6	9	10	11	12	13	14
0	0	3	6	9	10	11	12	13	14
1	1	4	7	10	11	12	13	14	15
2	2	5	8	11	12	13	14	15	16
9	9	12	15	18	19	20	21	22	23
12	12	15	18	21	22	23	24	25	26
15	15	18	21	24	25	26	27	28	29
18	18	21	24	27	28	29	30	31	32
21	21	24	27	30	31	32	33	34	35
24	24	27	30	33	34	35	36	37	38

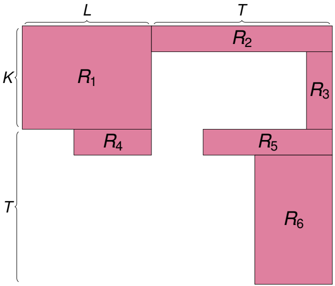
$T=7$	0	3	6	9	10	11	12	13	14	15
0	0	3	6	9	10	11	12	13	14	15
1	1	4	7	10	11	12	13	14	15	16
2	2	5	8	11	12	13	14	15	16	17
9	9	12	15	18	19	20	21	22	23	24
12	12	15	18	21	22	23	24	25	26	27
15	15	18	21	24	25	26	27	28	29	30
18	18	21	24	27	28	29	30	31	32	33
21	21	24	27	30	31	32	33	34	35	36
24	24	27	30	33	34	35	36	37	38	39
27	27	30	33	36	37	38	39	40	41	42

$T=8$	0	3	6	9	10	11	12	13	14	15	16
0	0	3	6	9	10	11	12	13	14	15	16
1	1	4	7	10	11	12	13	14	15	16	17
2	2	5	8	11	12	13	14	15	16	17	18
9	9	12	15	18	19	20	21	22	23	24	25
12	12	15	18	21	22	23	24	25	26	27	28
15	15	18	21	24	25	26	27	28	29	30	31
18	18	21	24	27	28	29	30	31	32	33	34
21	21	24	27	30	31	32	33	34	35	36	37
24	24	27	30	33	34	35	36	37	38	39	40
27	27	30	33	36	37	38	39	40	41	42	43
30	30	33	36	39	40	41	42	43	44	45	46

Computing the Number of Terms ($K \leq L$)

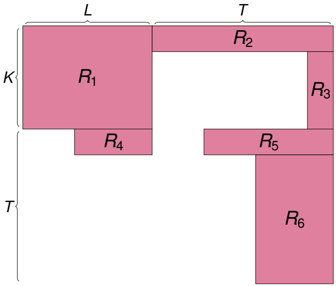


Computing the Number of Terms ($K \leq L$)



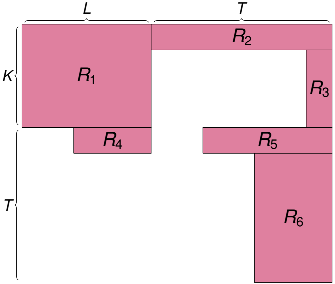
- $R_1 = K \times L$

Computing the Number of Terms ($K \leq L$)



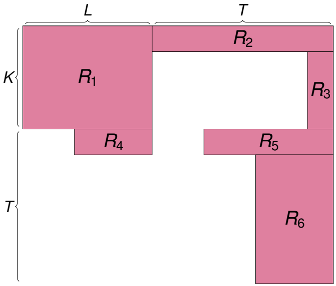
- $R_1 = K \times L$
- $R_2 = 1 \times T$

Computing the Number of Terms ($K \leq L$)



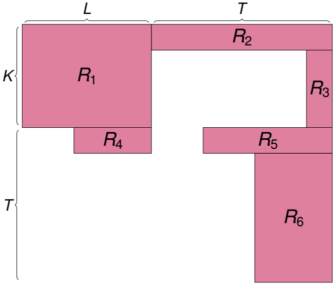
- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$

Computing the Number of Terms ($K \leq L$)



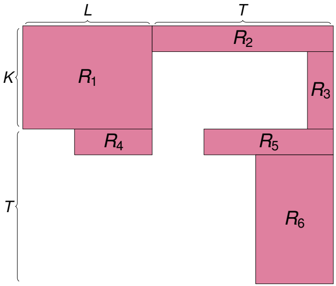
- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$
- $R_4 = 1 \times \max\{L - \left\lfloor \frac{T-2}{K} \right\rfloor - 2, 0\}$

Computing the Number of Terms ($K \leq L$)



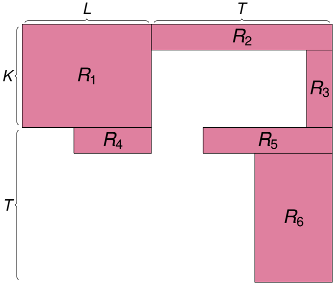
- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$
- $R_4 = 1 \times \max\{L - \left\lfloor \frac{T-2}{K} \right\rfloor - 2, 0\}$
- $R_5 = 1 \times \min\{T, KL - K + 1\}$

Computing the Number of Terms ($K \leq L$)



- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$
- $R_4 = 1 \times \max\{L - \left\lfloor \frac{T-2}{K} \right\rfloor - 2, 0\}$
- $R_5 = 1 \times \min\{T, KL - K + 1\}$
- $R_6 = (T - 1) \times \min\{T, K\}$

Computing the Number of Terms ($K \leq L$)



- $R_1 = K \times L$
- $R_2 = 1 \times T$
- $R_3 = (K - 1) \times 1$
- $R_4 = 1 \times \max\{L - \left\lfloor \frac{T-2}{K} \right\rfloor - 2, 0\}$
- $R_5 = 1 \times \min\{T, KL - K + 1\}$
- $R_6 = (T - 1) \times \min\{T, K\}$

Theorem

$N = |\text{terms in GASP}_{\text{big}}| = R_1 + \ldots + R_6$.

Number of Terms

Theorem

The number of terms in $\text{GASP}_{\text{small}}$, for $K \leq L$, is

$$N = \begin{cases} 2K + T^2 & \text{if } L = 1, T < K \\ KT + K + T & \text{if } L = 1, T \geq K \\ KL + K + L & \text{if } L \geq 2, 1 = T < K \\ KL + K + L + T^2 + T - 3 & \text{if } L \geq 2, 2 \leq T < K \\ KL + KT + L + 2T - 3 - \left\lfloor \frac{T-2}{K} \right\rfloor & \text{if } L \geq 2, K \leq T \leq K(L-1) + 1 \\ 2KL + KT - K + T & \text{if } L \geq 2, K(L-1) + 1 \leq T \end{cases}$$

Number of Terms

Theorem

The number of terms in $\text{GASP}_{\text{small}}$, for $K \leq L$, is

$$N = \begin{cases} 2K + T^2 & \text{if } L = 1, T < K \\ KT + K + T & \text{if } L = 1, T \geq K \\ KL + K + L & \text{if } L \geq 2, 1 = T < K \\ KL + K + L + T^2 + T - 3 & \text{if } L \geq 2, 2 \leq T < K \\ KL + KT + L + 2T - 3 - \left\lfloor \frac{T-2}{K} \right\rfloor & \text{if } L \geq 2, K \leq T \leq K(L-1) + 1 \\ 2KL + KT - K + T & \text{if } L \geq 2, K(L-1) + 1 \leq T \end{cases}$$

- For big T, $N = 2KL + (K+1)T - K$.

Number of Terms

Theorem

The number of terms in $\text{GASP}_{\text{small}}$, for $K \leq L$, is

$$N = \begin{cases} 2K + T^2 & \text{if } L = 1, T < K \\ KT + K + T & \text{if } L = 1, T \geq K \\ KL + K + L & \text{if } L \geq 2, 1 = T < K \\ KL + K + L + T^2 + T - 3 & \text{if } L \geq 2, 2 \leq T < K \\ KL + KT + L + 2T - 3 - \left\lfloor \frac{T-2}{K} \right\rfloor & \text{if } L \geq 2, K \leq T \leq K(L-1)+1 \\ 2KL + KT - K + T & \text{if } L \geq 2, K(L-1)+1 \leq T \end{cases}$$

- For big T, $N = 2KL + (K+1)T - K$.
- This is worse than GASP_{big}.

Number of Terms

Theorem

The number of terms in $\text{GASP}_{\text{small}}$, for $K \leq L$, is

$$N = \begin{cases} 2K + T^2 & \text{if } L = 1, T < K \\ KT + K + T & \text{if } L = 1, T \geq K \\ KL + K + L & \text{if } L \geq 2, 1 = T < K \\ KL + K + L + T^2 + T - 3 & \text{if } L \geq 2, 2 \leq T < K \\ KL + KT + L + 2T - 3 - \left\lfloor \frac{T-2}{K} \right\rfloor & \text{if } L \geq 2, K \leq T \leq K(L-1) + 1 \\ 2KL + KT - K + T & \text{if } L \geq 2, K(L-1) + 1 \leq T \end{cases}$$

- For big T, $N = 2KL + (K+1)T - K$.
- This is worse than GASP_{big}.
- Is $\text{GASP}_{\text{small}}$ always worse?

$\text{GASP}_{\text{small}}$ outperforms GASP_{big} for small T. ($K = L = 3, T = 2$)

	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
10	10	13	16	19	20

- GASP_{big}
- $N =$ Purple Area $= 19$

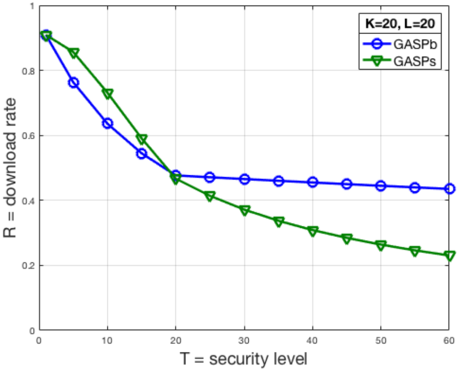
	0	3	6	9	10
0	0	3	6	9	10
1	1	4	7	10	11
2	2	5	8	11	12
9	9	12	15	18	19
12	12	15	18	21	22

- $\text{GASP}_{\text{small}}$
- $N =$ Purple Area $= 18$

What is small T?

Theorem

$\text{GASP}_{\text{small}}$ outperforms GASP_{big} for $T < \min\{K, L\}$.



Best of Both Worlds

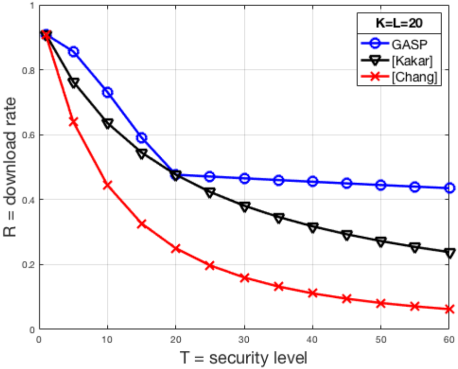
Definition

$$\mathrm{GASP} = \begin{cases} \mathrm{GASP}_{\mathrm{small}} & \text{if } T < \min\{K, L\} \\ \mathrm{GASP}_{\mathrm{big}} & \text{if } \min\{K, L\} \leq T. \end{cases}$$

Best of Both Worlds

Theorem

GASP codes outperform all previous schemes in terms of communication cost.



Open Problems

- Is GASP optimal?

Open Problems

- Is GASP optimal?
- Is GASP asymptotically optimal? (2 servers per collusion)

Open Problems

- Is GASP optimal?
- Is GASP asymptotically optimal? (2 servers per collusion)
- How about total communication cost.

Open Problems

- Is GASP optimal?
- Is GASP asymptotically optimal? (2 servers per collusion)
- How about total communication cost.
- Other matrix divisions.

Open Problems

- Is GASP optimal?
- Is GASP asymptotically optimal? (2 servers per collusion)
- How about total communication cost.
- Other matrix divisions.
- Are polynomial codes optimal?

Open Problems

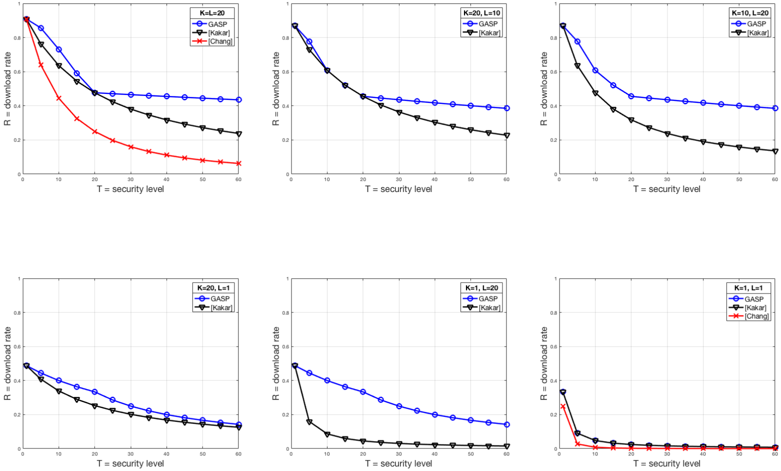
- Is GASP optimal?
- Is GASP asymptotically optimal? (2 servers per collusion)
- How about total communication cost.
- Other matrix divisions.
- Are polynomial codes optimal?
- Other applications for the degree table (ex. Tensor Products).

Open Problems

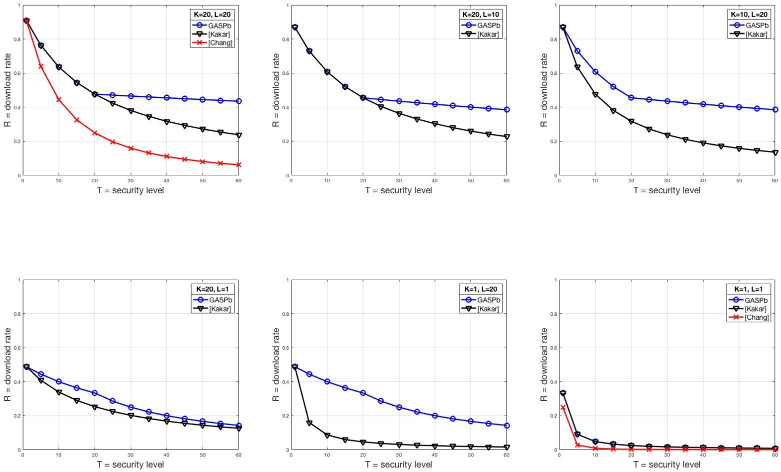
- Is GASP optimal?
- Is GASP asymptotically optimal? (2 servers per collusion)
- How about total communication cost.
- Other matrix divisions.
- Are polynomial codes optimal?
- Other applications for the degree table (ex. Tensor Products).
- Apply GASP to gradient descent.

Danke schön

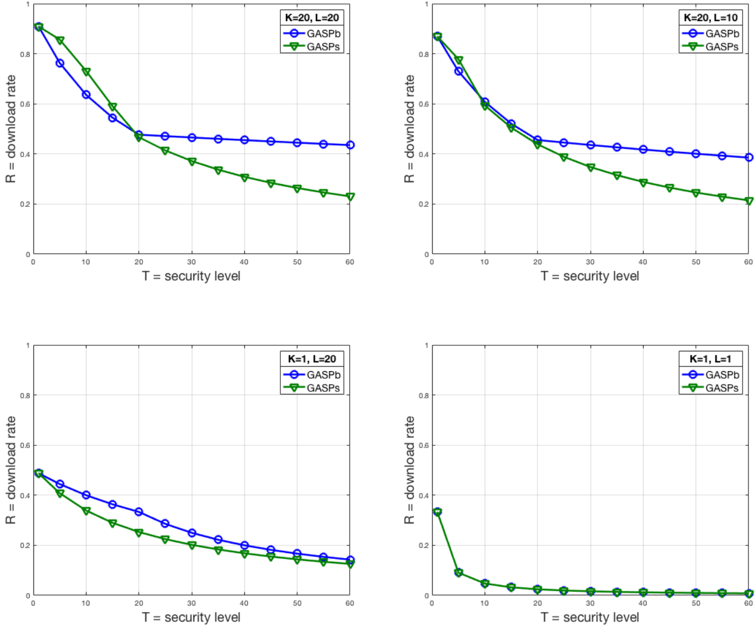
GASP vs World



GASP$_{big}$ vs World



GASP_{big} vs $\text{GASP}_{\text{small}}$



Fixed number of workers ($N = 50$)