Sino-Russian Mathematical Challenge Fund Project Application Guide in 2024

1. Overall objective

The Sino-Russian Mathematical Challenge Fund aims to promote innovation and development in the field of mathematics and promote the practical application of mathematical achievements in the industry. Industry and academia collaborate to consolidate and release mathematical problems; At the same time, researchers from cross-border and different fields are encouraged to apply for projects to jointly solve mathematical problems in the industry.

2. Challenge direction

In 2024, the mathematical challenge fund totals CNY 7 million and is planned to support 14 projects, each of which is planned to support CNY 500,000(before tax). The project implementation cycle is generally one year.

No.	Challenge Puzzle List
1	Optimal Shortening of Linear Codes and Related Applications
2	Large-scale optical network planning, protection, and route optimization
3	Optimization Algorithm for Large-Scale Customization Layout
4	Efficient Computing - Efficient Search Algorithm for Ultra-Large Combination Space
5	Efficient Computing - Low-Power Chip Algorithms-Low-Bit Width Forward Network
6	Optimal Matrix-Vector Operation
7	Finding a Solution to Fiber Nonlinear Schrodinger Equation
8	Radio Channel Reconstruction Under Limited Observation -In Search of a More Efficient Estimation Model
9	Multi-Channel Joint Resource Allocation Algorithm Based on Mathematical Programming
10	Various AI-based RRM Decision Tasks in Dynamic Wireless Environments
11	Acceleration of Score-based diffusion process
12	AI_AI-enabled Full Real-Time CSI Acquisition in the HBF Architecture
13	Accelerated Iterative Algorithm for Solving Large Complex Sparse Matrix
14	Numerical Stability Analysis of Communication Avoiding LU (CALU) factorization

Note: The intellectual property rights of the project-related achievements formed by the Funding project during the project research process, including but not limited to papers,

works, source code, etc., shall be shared by the Funder, the applicant and the unit to which the applicant belongs to. Detailed terms are specifically agreed by the project funding agreement.

01 Optimal Shortening of Linear Codes and Related Applications

1. Background

The study of error-correcting codes with good parameters and low-complexity coding/decoding algorithms is the core topic of coding research. Using codes with a better decoding algorithm to construct new codes with better parameters is an important direction of coding research. Generally, operation methods such as puncture, shorten, direct sum, and tensor product are used. It is a common practice to shorten information bits of a code having a relatively good decoding algorithm to optimize code parameters. In this way, the existing decoding algorithm usually only needs to be simply adjusted for further usage. During code shortening, a code with a larger minimum distance may be obtained. In most cases, a relatively-large number of coordinate positions need to be processed, leading to a significant decrease in code rate. In specific applications, it is expected that a code with optimal parameters is found when a certain code rate is maintained (that is, a quantity of acceptable coordinate positions is shortened).

Due to the application of a soft decoding algorithm, the number of codewords whose weight reaches the minimum distance of codes also becomes an important indicator for measuring the error detection and correction capabilities of codes. In specific applications, it is expected that new codes with a minimum distance of codes as large as possible are obtained in a shortening manner. If the minimum distance of new codes cannot be increased, it is expected that new codes with a minimum number of codewords whose weight reaches the minimum distance can be obtained based on the same minimum distance and code rate.

Currently, some related results about binary Hamming codes, binary Extended Hamming codes, and the like are available. There is a clear shortening solution for binary Hamming codes. For the shortened codes of binary extended Hamming codes, there is a partial description of the upper and lower bounds of counting the number of minimum-distance codewords. It is a challenging and promising research topic to further study the exact value and the upper and lower bounds for the number of minimum-distance codewords of the shortened codes of extended Hamming codes and general linear codes.

2. Challenge Description

Assume that C is the linear code $[n, k, d]$ in the finite field $GF(q)$, and C_T represents a shortened code of the C code on the coordinate position set $T \subseteq \{1, 2, \ldots, n\}$. It is assumed that the generator matrix of C is $G = [-P | I_k]$.

Problem: For a label of a column whose number of bases is $|T| = s$ in $T = \{i_1, \ldots, i_s\} \subseteq \{n - k + 1, \ldots, n\}$, a proper policy is used to search for a special set T' whose number of bases is s, so that $C_{T'}$ is a linear code with a minimum number of codewords whose weight is d in all sets $\{C_T\}$.

3. Demand

a) **Theoretical objective:** To study the characterization of the number of codewords whose weight of a shortened code reaches the minimum distance of a general linear code or a linear code with

rich algebraic structures (such as a cyclic code, a BCH code, and a tensor product code) in a finite field. Specifically:

1) Provide the upper and lower bounds of the number of codewords whose weight reaches the minimum distance of codes in corresponding shortened codes.

2) In some special cases, provide a formula for counting the number of codewords in (1).

3) Provide the optimal shortening algorithm and compare it with existing results.

b) **Application objective:** To provide the optimal shortening scheme and algorithm for binary extended Hamming codes and correlation codes, and compare them with the results obtained by general linear codes. For example, when the number of bits to be shortened ranges from 30 to 40, provide the optimal shortening algorithm for binary Extended Hamming codes with a parameter of [128,120,4].

02 Large-scale optical network planning, protection, and route optimization

1. Background

In an optical network, demands between nodes are assigned to a route through the network and a specific wavelength. The route should be a simple path which connects source node and destination node. To cope with fiber cut, working path is needed as well as the recovery path.

2. Challenge Description

Given:

Network: graph G with nodes V and edges E with lengths.

Constants: the maximal lightpath length L_{max}; the set of available frequencies $[M] = \{1,2,\ldots,M\}$, M=80; the capacity of a frequency $C_{max} = 80$.

Demands **D.** A demand $d \in D$ is a triple (v, u, c), where $v, u \in V$ are the end-nodes, and c is a capacity of the demand, $c \in C$, $C = \{1, 2, 8, 32, 80\}$.

Definitions:

Lightpath is a tuple (P, f), where $P \subseteq E$ is a simple (without self-intersections) path with length $\leq L_{max}$ and f is a frequency $f \in [M]$.

Scenario $S \subseteq E$ is a network state when the edges S is temporarily removed from the network. Such edges are called failed. All lightpaths that use a failed edge are invalid. We only consider scenarios with no more than one edge.

Working path w(d) for a demand d=(v,u,c) is a sequence of lightpaths s.t. these lightpaths form a simple path from v to u. The path w(d) is **active** if and only if there are no failed edges (i.e. in scenario S=∅).

Recovery path r(d,S) for a demand d=(v,u,c) and a scenario S is a sequence of lightpaths s.t. they form a simple path from v to u and do not use failed edges S. The path r(d, S) is **active** in scenario S only.

Temporary configuration g(h,S) for a lightpath h in scenario S is a vector $(g_1, g_2, g_8, g_{32}, g_{80})$ where g_c is the number of demands with capacity c whose **active** paths use the lightpath h in the scenario S.

Configuration G(h) of a lightpath h is equal to the componentwise maximum of g(h,S) for all scenarios S. The **capacity** of a configuration G(h) = $(g_1, g_2, \ldots, g_{80})$ is a weighted sum $C(h) = \sum_{c \in C} c \cdot g_c$.
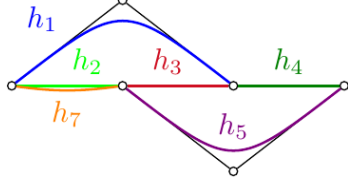
Network
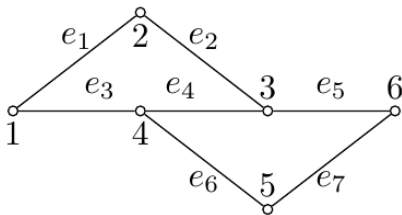
Demands

$$d_1 = (1, 6, 8), \quad d_2 = (1, 6, 8),$$
$$d_3 = (1, 3, 32), \quad d_4 = (1, 3, 32).$$
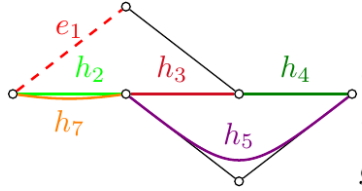
Lightpaths

Working paths

$$w(d_1) = (h_2, h_3, h_4), \quad w(d_2) = (h_2, h_5),$$
$$w(d_3) = (h_1), \quad w(d_4) = (h_1),$$
$$g(h_3, \emptyset) = (0, 0, 1, 0, 0).$$
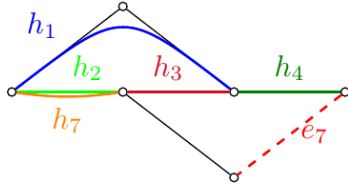
Scenario{e_1}

$$r(d_1, \{e_1\}) = (h_2, h_3, h_4), \quad r(d_2, \{e_1\}) = (h_2, h_5),$$
$$r(d_3, \{e_1\}) = (h_7, h_3), \quad r(d_4, \{e_1\}) = (h_7, h_3),$$
$$g(h_3, \{e_1\}) = (0, 0, 1, 2, 0).$$

Scenario{e_7}

$$r(d_1, \{e_7\}) = (h_2, h_3, h_4), \quad r(d_2, \{e_7\}) = (h_2, h_3, h_4),$$
$$r(d_3, \{e_7\}) = (h_1), \quad r(d_4, \{e_7\}) = (h_1),$$
$$g(h_3, \{e_7\}) = (0, 0, 2, 0, 0).$$

Configuration of h_3 $\quad$ G(h_3) = (0,0,2,2,0)

3. Demand

Find:

The set of lightpaths H.

Working path w(d) for each demand d.

Recovery path r(d,S) for each demand d and each scenario $S = \{e\}$, $e \in E$.

Minimize |H|.

Constraints:

1) Any edge $e \in E$ and any frequency $f \in [M]$ can not be simultaneously used by two or more lightpaths in H.

2) Working and recovery paths can only consist of lightpaths from H.

3) If lightpaths in w(d) use no failed edges S, then r(d,S) = w(d).

4) For any lightpath $h \in H$, $C(h) \leq C_{max}$.

5) |V| < 5000, |E| < 10000, |D| < 100000.

6) The time limit is 1 hour for a regular PC.

- **Reference**

[1] Benoit Vignac, François Vanderbeck, Brigitte Jaumard, Reformulation and Decomposition Approaches for Traffic Routing in Optical Networks.

- **Reference**

[1] Benoit Vignac, François Vanderbeck, Brigitte Jaumard, Reformulation and Decomposition Approaches for Traffic Routing in Optical Networks.

03 Optimization Algorithm for Large-Scale Customization Layout

1. Background

In the conventional chip algorithm implementation process, designers split the entire chip algorithm system into multiple high level blocks (HLBs) based on certain scale constraints, plan the layout and I/O interfaces between HLBs, and use EDA tools to complete the internal layout and routing of each HLB automatically. During the process, the designers need to optimize and resolve the layout bottlenecks identified by the EDA tools. Currently, most chip design tools perform optimization for general-purpose computing architectures such as GPU and NPU. Optimizing layout and routing for the oDSP algorithm architecture will be an effective way to reduce the power consumption of oDSP introduced by connections.

2. Current Result

About 300 algorithm units can be placed and optimized through phased optimization, heuristic initial placement[1][2], and local search.
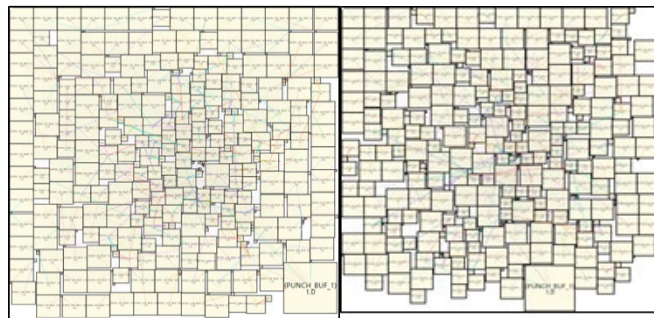

Figure 1 Comparison between two placement methods

3. Challenge Description

Challenge: Implement the macro layout of 3000 units to minimize the total routing cost while complying with area utilization constraints.

This challenge aims to maximize the area utilization and minimize the total routing cost by splitting the and layout of a specified set of chip algorithm units.

1. The existing solution supports a large granularity of units to be processed. If these algorithm units are further split, the number of target units is expected to increase by 10 times, and the number of placement combinations for units layout increases exponentially.

2. To improve the area utilization, adjust unit shapes in addition to the placement (Area utilization = Total area of units/Total area of the layout rectangle x 100%).

3. To reduce the routing cost, place logically connected units close to each other. But, adjusting the positions of the units may affect the area utilization. In actual design, we need to consider both the routing cost and the area utilization.

4. Comprehensive optimize the routing cost and area utilization based on the unit set after automatic

combination and splitting.

Input: initial parameters of specified layout units

1. Initial unit set ($|V| \leq 3000$), and initial unit area a_i, $\forall i \in V$. The following lists the initial input parameters of a 500-element unit set as an example.

| NodeID	NodeArea	Parallel	NodeType
0	0	400	IO-I
1	62400	400	FuncUnit
2	84800	400	FuncUnit
3	16400	1	FuncUnit
4	20000	1	FuncUnit
...	...	...	...
500	0	400	IO-O

    a. NodeID: initial unit ID.

    b. NodeArea: initial unit area.

    c. Parallel: parallelism of initial units. For example, Node2 contains 400 same units that work in parallel and are independent of each other, and has a total area of 84800. The units can be split as required during layout. The split units are independent of each other and are not connected. A unit with parallelism 1 is a whole unit whose internal submodules are interconnected and cannot be automatically split.

    d. NodeType: node type. IO-I indicates data input, and IO-O indicates data output. I/O interfaces are distributed at the edges of the overall layout rectangle and do not occupy any area. FuncUnit indicates a functional unit.

2. E refers to the connection relationships between units. The edge weight $w: E \rightarrow \mathbb{N}$ of each connection indicates the number of wires. The following table lists input parameters of four connection relationships as an example.

| SrcNodeID	DstNodeID	ConnectionWeight
0	1	7200
1	2	7200
1	3	7200
2	4	4800
3	4	3200
...	...	...
499	500	8000

    a. SrcNodeID: ID of the source unit for a connection.

    b. DstNodeID: ID of the destination unit for a connection.

    c. Connection Weight: connection cable weight. This parameter can be split based on the parallelism of the source unit or destination unit.

3. Optional prompt: initial unit width and height $set(w_{i0}, h_{i0})_{i \in V}$ and $w_{i0}h_{i0} = a_i$. The total width and height of the initial layout area are W_0, H_0.

The following figure shows the logical connections of the subsystems described in the table.
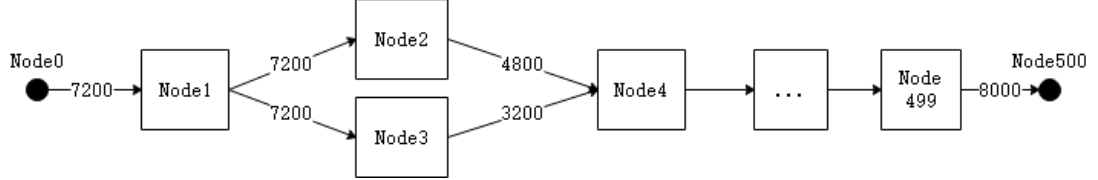

Example of parallelism-based splitting: In the third group of connection relationships, Node1 and Node2 are evenly split into four nodes, and Node3 does not support splitting. The following figure shows the comparison before and after splitting.
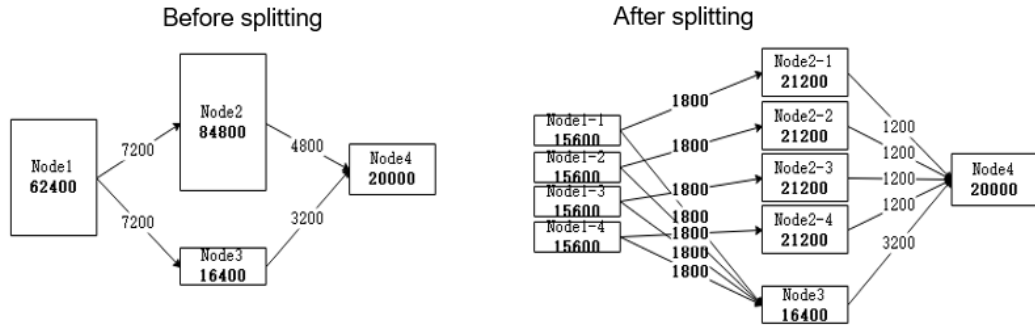
Output:

1. Total width and height of the final layout area W, H and layout area utilization

2. Horizontal/Vertical coordinates of each unit in the final unit set $x: V \rightarrow [0, W]$, $y: V \rightarrow [0, H]$

   (If units are split, the coordinates of the unit set after splitting are output.)

3. Final unit width and height $set\ (w_{i1}, h_{i1})_{i \in V}$

4. Layout effect drawing. The following is an example.
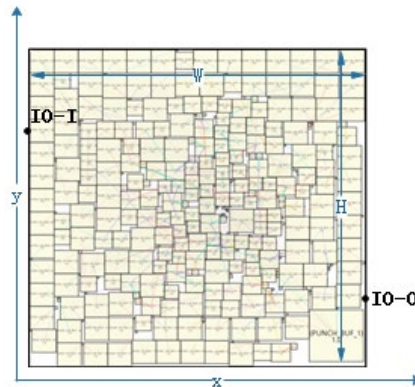
4. **Demand**

Research objectives:

1. In theory: Lower-bound theoretical analysis of the area utilization and total wire length

   a. Theoretical analysis of the relationship between unit-granularity area distribution and area utilization

   b. Lower-bound analysis of the total connection cost with specified inputs $\sum_{e \in E} \boldsymbol{w_e} \cdot \boldsymbol{HPWL_e}(i_1, i_2)$, $\forall i \in V$

      $\boldsymbol{e}$: the edge which connect units i_1 and i_2

      $\boldsymbol{w_e}$: weight of wire between units i_1 and i_2

      $\boldsymbol{HPWL_e}$: Half Perimeter Wire Length of $\boldsymbol{e}$

2. In practice: Complete the unit layout with specified inputs and minimize the total cable length $\sum_{e \in E} \boldsymbol{w_e} \cdot \boldsymbol{HPWL_e}(i_1, i_2)$, $\forall i \in V$.

Constraints:

1. I/O nodes are distributed on the left and right edges of the entire layout rectangle. The I/O nodes are routed in from the left and out from the right. The specific positions are not restricted.

2. Layout area utilization $\geq 90\%$: $\frac{\sum_{i \in V} a_i}{WH} \geq 0.9$

3. Constant constraints on the unit area: $w_{i1} h_{i1} = a_i$, $\forall i \in V$

4. Restrictions on the total aspect ratio of the layout area: $2:1 < W:H < 1:2$

5. All modules are located in the layout area: $\forall i \in V$, $x_i + \frac{w_i}{2} \leq W$, $y_i + \frac{h_i}{2} \leq H$

6. Units do not overlap each other: $\forall i \neq j \in V$, $R_i(x, y) \cap R_j(x, y) = \emptyset$

   $R_i(x, y) = \left\{ (a, b): x_i - \frac{w_i}{2} \leq a \leq x_i + \frac{w_i}{2}, y_i - \frac{h_i}{2} \leq b \leq y_i + \frac{h_i}{2} \right\}$ is a rectangle corresponding to unit i.

- **Reference**

[1] Module Placement on BSG-Structure and IC Layout Applications. Shigetoshi NAKATAKEy Department of Electrical and Electronic Engineering Tokyo Institute of Technology;
https://dl.acm.org/doi/pdf/10.5555/244522.244865

[2] TCG: A Transitive Closure Graph-Based Representation for General Floorplans. Jai-Ming Lin and Yao-Wen Chang, Member, IEEE http://cc.ee.ntu.edu.tw/~ywchang/Papers/tvlsi-tcg.pdf

04 Efficient Computing- Efficient Search Algorithm for Ultra-Large Combination Space

1. Background

Multiple-antenna systems will continue to evolve toward larger bandwidths and more antennas. There always is a strong requirement for accelerating medium-sized (X100-X1000) matrix computation. The acceleration of matrix pseudo-inverse is the one of the core requirements. The objective of this problem is to explore the automatic speedup of the pseudo-inverse operation on a set of matrices over a given set of atomic operators.
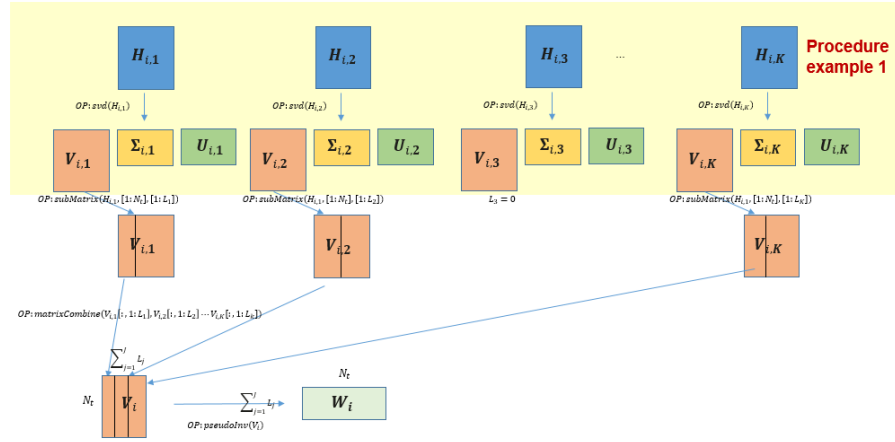
2. Current Result

Figure 1. An example of implementation procedure

Based on deep learning, the industry has conducted research on automatic acceleration of matrix multiplication [1] and sorting algorithms [2]. However, there are no good results for the pseudo-inverse operation of matrices. As shown in Figure 1, the optimization of the pseudo-inverse operation for a medium-sized matrix set mainly depends on manual optimization at present.

3. Challenge Description

a) **Challenge definition** :

Given a matrix set $\{H_{m,k}\}$, a real diagonal matrix D, and a nonnegative integer set $\{L_{m,k}\}$:

$$H_{m,k} \in \mathbb{C}^{N_t \times N_r}, \ m \in \{0, \dots, M-1\}, \ k \in \{0, \dots, K-1\},$$
$$D = \alpha \cdot I, \ \alpha \in \mathbb{R}^+,$$
$$L_{m,k} \in \{0, 1, \dots, N_r\},$$

where

$$N_t \geq \sum_k L_{m,k}, m \in \{0, \dots, M-1\}$$

The following calculation is to be performed for all m values:

$$W_m = V_m \left(V_m{}^H V_m + D \right)^{-1}, \ m \in \{0, \dots, M-1\}$$

Find computational graphs that produce correct results using the given operator set at minimum computation costs. In the target function:

$$\boldsymbol{V}_m = [\cdots \quad \overline{\boldsymbol{V}}_{m,k} \quad \cdots]$$
$$\overline{\boldsymbol{V}}_{m,k} = \boldsymbol{V}_{m,k}[:, \boldsymbol{0}:\boldsymbol{L}_{m,k} - \boldsymbol{1}]$$
$$\{\boldsymbol{U}_{m,k}, \boldsymbol{\Sigma}_{m,k}, \boldsymbol{V}_{m,k}\} = \text{SVD}(\boldsymbol{H}_{m,k})$$

b) **Parameter dimensions：**

$$N_t \in \{64,128,256\},$$
$$N_r \in \{4,8,16\},$$
$$M = \{192,384,768,1536\},$$
$$K = 100$$

As shown in Figure 2,the basic information of atomic operator sets includes:

➢ Operator set & typical specifications: (partly from) LAPACK

    ✓ Matrix multiplication: $matrixMul$

    ✓ Matrix factorization: $subMatrix$

    ✓ Matrix combination: $matrixCombine$

    ✓ Conjugate multiplication: $conjugateMul$

    ✓ Matrix inversion: inv

    ✓ Matrix pseudo-inversion: $pseudoInv$

    ✓ ...

➢ Operator cost

    ✓ By functions

    ✓ By tables

Operator Name	Operator Format	LAPACK Function	Link to Detailed Information	Operator Output	Cost (by Functions)	Cost (by Tables)
Matrix multiplication	OP:matMul(A,B)	zgemm		C: calculation result	$m \times n \times p$, when the size of matrix A is $m \times n$ and the size of matrix B is $n \times p$	Refer to the table.
Full-rank matrix multiplication by a lower triangular matrix	OP:matMulTriangle(A,B)	dtrmm		C: calculation result	$0.5 \times m \times n^2$, when the size of matrix A is $m \times n$ and the size of matrix B is $n \times n$	Refer to the table.
Lower triangular matrix multiplication by a full-rank matrix	OP:triangleMulMat(A,B)	trmm		C: calculation result	$0.5 \times n^2 \times p$, when the size of matrix A is $n \times n$ and the size of matrix B is $n \times p$	Refer to the table.
Matrix multiplication by a vector	OP:matMulVec(A,B)	dgemv		C: calculation result	$n \times m$, when the size of matrix A is $n \times m$ and the size of vector B is m	Refer to the table.
Inner product of a conjugate vector	OP:conjugateInnerProduct(A)	zdotc		B: calculation result	n^2, when the size of vector A is n	Refer to the table.
Vector multiplication	OP:vectorMul(A,B)	axpy		C: calculation result	n^2, when the size of vector A is n and the size of vector B is n	Refer to the table.
Outer product of a conjugate vector	OP:conjugateOutProduct(A)	Hadamard (not LAPACK function)		B: calculation result	n^2, when the size of vector A is n	Refer to the table.
Vector inner product	OP:vectorInnerProduct(A,B)	dot		C: calculation result	n^2, when the size of vector A is n and the size of vector B is n	Refer to the table.
Vector addition	OP:vectorAdd(A,B)	Vadd (not LAPACK function)		C: calculation result	n, when the size of vector A is n and the size of vector B is n	Refer to the table.
SVD	OP:svd(A)	dgesvd		V, Sigma, and U, which are the left singular vector, singular value matrix, and right singular vector, respectively	Refer to the table.	Refer to the table.
Submatrix extraction	OP:subMatrix(A,[x_start, x_end],[y_start,y_end])	Submatrix (not LAPACK function)		B: calculation result	$(x\_end - x\_start) \times (y\_end - y\_start)$	Refer to the table.
Matrix combination	OP:matrixCombine(A,B, C...)	Mergedmatrix (not LAPACK function)		res: calculation result	$n \times (sum\_i \, m\_i)$, when the size of each matrix is $n \times m\_i$	Refer to the table.
Cholesky decomposition	OP:chol(A)	dpotrf		L: calculation result	Refer to the table.	Refer to the table.

Figure 2. Example of the atomic operator sets and corresponding cost

4. Demand

a) **Theoretical Objectives：**

The theories or methods which could automatically optimize implementation of the specified matrix function based on a set of given operators (including but not limited to AI methods).

b) **Application Objectives**

Design an searching algorithm，which could output：

- ➢ Top 3 formal computation processes (for example, in computational graphs) that use operators in a given set and compute at minimum costs: When a data matrix set $\{\boldsymbol{H}_{m,k}\}$ is input, such a process can output a result set $\{\widehat{\boldsymbol{W}}_m\}$ that meets the following condition:

$$\max_m \left| \widehat{W}_m - W_m \right| < \sigma$$

Where，σ is the required computation precision.

With the following inputs:

- ➢ Data matrix sets: $\{\boldsymbol{H}_{m,k}\}$, $\boldsymbol{D}$, and a non-negative integer set $\{L_{m,k}\}$

- ➢ Formula of the target matrix: $\boldsymbol{W}_m = \boldsymbol{V}_m \left(\boldsymbol{V}_m{}^{\boldsymbol{H}} \boldsymbol{V}_m + \boldsymbol{D} \right)^{-1}$

- ➢ Available operator set

- ➢ Cost of each operator in the set

- ➢ Number of computing cores (C): The cores are homogeneous, and data migration between cores is cost-free.

- **Reference**

[1]. Discovering faster matrix multiplication algorithms with reinforcement learning, Alhussein Fawzi, Matej Balog, Aja Huang, Thomas Hubert, …, Nature volume 610, pages47–53 (2022)

[2]. Faster sorting algorithms discovered using deep reinforcement learning, Daniel J. M., Andrea M., Anton Z., …., Nature volume 618, pages257–263 (2023)

# 05 Efficient Computing - Low-Power Chip Algorithms-Low-Bit Width Forward Network

1. Background

The bandwidth and rate supported by the communication algorithm are increasing, and the power consumption/chip area required by the algorithm is also increasing. Bit-width design is an important technical path to reduce hardware overhead. We want to Study the typical bit width design algorithm to reduce the hardware overhead.

2. Current Result

The model shown in the following figure is a common cascaded model structure for modeling nonlinearities. The modeling capability of the cascaded model is stronger than that of the non-cascaded model. The bit width in the model is generally 12-16 bits, which directly affects the chip resources.
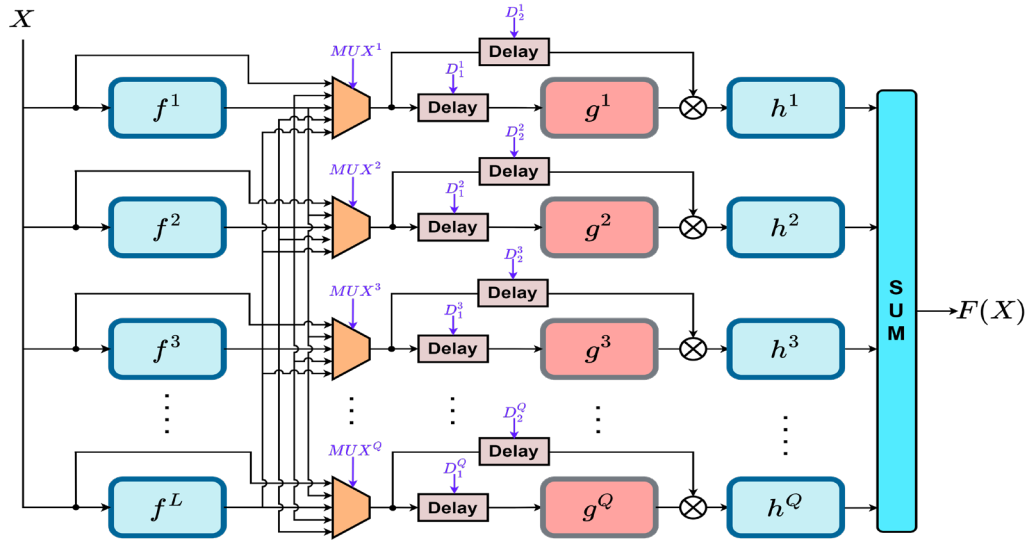
3. Challenge Description

- **Optimization goals:**

$$\min_{B_X, B_\alpha, B_c, B_\beta} Cost(F, X)$$

$$\text{st. } \|F(X) - F_\infty(X_\infty)\| \le e$$

- **Parameter description:**

$X = [X(1), X(2), \dots] \in \mathbb{C}^N$ is an input signal sequence, wherein N is the length of the input signal. X_∞ is a floating point of X. $e \in \mathbb{R}^+$ is an expected error upper limit. $Cost(\cdot)$ is a cost function.

The forward function $F(\cdot)$ is formed after the following subfunctions are combined according

to the topology shown on the left. $F_\infty(\cdot)$ is the floating point of $F(\cdot)$.

$$f^i: y^i(n) = \sum_{m=0}^{M} \alpha_m^i x^i(n-m), i = 1,2,\dots,L$$

$$D_j^i: y^i(n) = x^i(n - D_j^i), \ i = 1,2,\dots,Q, j = 1,2$$

$$g^i: y^i(n) = \sum_{p=0}^{P} c_p^i |x^i(n)|^{2p}, i = 1,2,\dots,Q$$

$$h^i: y^i(n) = \sum_{m=0}^{M} \beta_m^i x^i(n-m), i = 1,2,\dots,Q$$

$D_j^i, p \in \mathbb{z}^+; \alpha_m^i, c_p^i, \beta_m^i \in \mathbb{C}; B_X, B_\alpha, B_c, B_\beta$ are the bit widths of signal X, subfunction coefficient α_m^i, subfunction coefficient c_p^i, and subfunction coefficient β_m^i, respectively.

4. Demand

■ A bit width search and analysis method for the forward network, the bit width is optimized by 40% compared with the baseline.

■ Minimize the signal and coefficient bit widths of each node in the above network, and ensure that the error between the output signal after bit width compression $F(X)$ and that of arbitrary-precision floating point $F_\infty(X_\infty)$ is less than the present value $e(e <0.5\text{dB})$.

- **Reference**

[1] Dinis D C, Cordeiro R F, Barradas F M, et al. Agile single-and dual-band all-digital transmitter based on a precompensated tunable delta–sigma modulator. IEEE Transactions on Microwave Theory and Techniques, 2016, 64(12): 4720-4730.

[2] Gholami A, Kim S, Dong Z, et al. A survey of quantization methods for efficient neural network inference. Low-Power Computer Vision. Chapman and Hall, CRC, 2022: 291-326.

[3] Nagel M, Fournarakis M, Amjad R A, et al. A white paper on neural network quantization. arXiv preprint arXiv:2106.08295, 2021

[4] X. Liu, W. Chen and Z. Feng, "Broadband Digital Predistortion Utilizing Parallel Quasi- Wiener-Hammerstein Model with Extended Dynamic Range," 2021 IEEE MTT-S International Wireless Symposium (IWS), Nanjing, China, 2021, pp

06 Optimal Matrix-Vector Operation

1. Background

As basic computing units, FIR filter and discrete Fourier transform (DFT) consume huge amounts of energy in modern communication chips. They are widely used in calculations such as feature extraction, channel equalization, modulation and demodulation. Mastering an efficient FIR/DFT structure plays a critical role in improving the energy efficiency ratio of communication chips.

In mathematics, FIR and DFT can be written as

$$\mathbf{y} = \mathbf{H}\mathbf{x}$$

where $\mathbf{x}^T = \{x_0 \quad x_1 \quad \cdots \quad x_{N-1}\}$ is used as an input vector of the system, $\mathbf{H}$ is used as a compute matrix, and $\mathbf{y}$ is used as an output vector. For DFT, $\mathbf{H}$ is a square matrix whose size is N with each element value being $(\mathbf{H}_N)_{mn} = e^{-j\frac{mn2\pi}{N}}$. For an FIR filter whose length is L, $\mathbf{H}$ is a Toeplitz matrix whose size is L × N.

As the compute matrix $\mathbf{H}$ in DFT/FIR has certain structural features, if we can find a smart way to decompose the matrix $\mathbf{H}$, the DFT/FIR circuit can be greatly simplified.

For example, the classical Toom-Cook algorithm uses the Lagrange polynomial to decompose the matrix $\mathbf{H}$ to slash the times of multiplication operations [1].

$$\mathbf{y} = \mathbf{H}\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} s(\beta_0) \\ 0.5s(\beta_1) \\ 0.5s(\beta_2) \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ -1 & 1 & 1 \end{bmatrix} \begin{bmatrix} h_0 & 0 & 0 \\ 0 & 0.5(h_0 + h_1) & 0 \\ 0 & 0 & 0.5(h_0 - h_1) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 1 & -1 \end{bmatrix} \mathbf{x}$$

where $\mathbf{H} = \begin{bmatrix} h_0 & 0 \\ h_1 & h_0 \\ 0 & h_1 \end{bmatrix}$, $\beta_0, \beta_1, \beta_2$ is a group of different real numbers, and $\mathbf{s}(\boldsymbol{\beta})$ is a Lagrange interpolation result about H and x [1].

In the formula above, the decomposition calculation of matrix vectors is mapped to the microarchitecture of FIR/DFT circuits. However, it can be difficult to find an efficient decomposition method for matrix vectors within a short period. First, the search space is huge. Taking the 3x3 convolution as an example, when a constraint is applied, the value range of the decomposed matrix element is $\mathbf{F} = \{0, \pm 1\}$. In this way, multiplication is not involved in chip implementation. Even in this case, the number of dimensions of the potential decomposition result still exceeds 10^{23}. In addition, we expect to achieve an energy-efficient circuit architecture by looking for a good decomposition method. Such being the case, when calculating efficient decomposition methods, we must consider the impact and constraints of ASIC, such as line length, interconnection, and drive strength. This adds more burden to our search. In many cases, circuit designers find that theoretically, a simplified decomposition method may not necessarily deliver a circuit architecture with high hardware efficiency.

2. Current Result

Currently, FIR/FFT design relies heavily on algorithm and chip designers' manual design of specific chip indicators based on some typical structures, such as split-radix FFT structure or Winograd FIR. For matrix operation optimization, a large number of reinforcement learning-based methods are available, such as "Discovering novel algorithms with AlphaTersnor".

3. Demand

For $\mathbf{y} = \mathbf{Hx}$, it is assumed that the size of $\mathbf{H}$ is no greater than 100×100 and the flip rate of $\mathbf{H}$ is 1% of x. We expect to find a collection of solutions that span from the least times of multiplications to most times of multiplications and meet actual requirements. The definition of the flip rate means that the cost of calculation operations on elements in the matrix $\mathbf{H}$ is 1% the cost of elements in the vector x.

- **Reference**

[1] www.ece.umn.edu/users/parhi/SLIDES/chap8.pdf

07 Finding a Solution to Fiber Nonlinear Schrodinger Equation

1. Background

In a high-speed optical fiber link, light intensity affects a refractive index of light, thereby causing a Kerr nonlinear effect. This effect is one of the main constraints limiting the capacity growth of WDM communication systems. Based on Maxwell's equations and the crystalline characteristics of the fiber, the nonlinear Schrodinger equation of the fiber can be derived to represent the Kerr nonlinearity. In a WDM system, to implement accurate signal simulation, the first class of nonlinear Schrodinger equations that describe optical signals, that is, Manakov-PMD equations, need to be effectively solved. However, accurate solutions of the equation in a real communication scenario usually involve many calculations. For example, if a single-core processor is used, it usually takes several weeks to complete calculations in the corresponding scenario.

2. Current Result

Because the nonlinear channel model of ultra-broadband optical fiber is very complex, the calculation workload of digital simulation is huge, which hinders the development of efficient and high-precision solution algorithms in the optics field. Currently, the fiber nonlinear Schrodinger equation is solved by using the so-called split-step (Fourier) method (Govind p. Agrawal, Nonlinear Fiber Optics, Academic Pres, 2013), which is developed based on the fiber signal transmission characteristics and the advantages of Fast Fourier Transform (FFT). Even so, the simulation of a single wavelength over 2000 km still takes eight hours, and the simulation of five wavelengths takes one week. However, the actual system can have 80 wavelengths at most.

3. Challenge Description

We are seeking for a more efficient method for solving the Manakov-PMD equation. The equation is as follows:

$$\frac{\partial \mathbf{E}(z,t)}{\partial z} + j\frac{\beta_2}{2}\frac{\partial^2 \mathbf{E}(z,t)}{\partial t^2} + \frac{\alpha}{2}\mathbf{E}(z,t) = j\gamma|\mathbf{E}(z,t)|^2\mathbf{E}(z,t), \qquad z > 0, \qquad t > 0$$

$E(z,t) = [E_x(z,t), E_y(z,t)]^T \in$ L2(R) is the envelope function of the optical signal, in the unit of watts. z represents the transmission distance, and its maximum distance Z can reach thousands of kilometers. t represents the transmission time, which is a variable in the unit of second and determined by the input data sequence and is related to the symbol rate. β_2 is a positive real number indicating the dispersion feature and ranges from 10-27 s^2/m to 10-26 s^2/m. α is a positive real number indicating the attenuation coefficient and is around 10-5. γ is a positive real number indicating the nonlinear feature of a fiber. It ranges from 10-4 to 10-3, in the unit of $1/(m \cdot w)$. The parameters β_2, α, γ, and Z and the transmitted signal $E(0,t)$ in the equation are given. During $E(0,t)$ signal transmission, the signal is lumped and amplified at a relay site. Assume that the link has equal spans and the fiber length of each span is L. When the transmission distance is the end point distance of each span ($z = k \cdot L, k = 0, 1, \ldots Z/L$), the signal amplitude is amplified by $G = e^{\alpha L/2}$ times.

4. Demand

a) **In theory:** We are seeking for an efficient new method for solving the nonlinear Schrodinger equation. This method must support 1/10 (or even less) calculation workload of that in the split-step (Fourier) method with the precision unchanged. The reliability of the solution must be proved theoretically.

b) **In practice:** When the transmission distance is greater than 1200 km or less than 160 km, design an efficient channel solution mode for specific scenarios.

08 Radio Channel Reconstruction Under Limited Observation In Search of a More Efficient Estimation Model

1. Background

As shown in Figure 1, massive MIMO is an important direction of future evolution of a radio access network, and performance improvement of the massive MIMO heavily depends on precision of channel information obtained. As the dimension of massive MIMO and UEs increases, the traditional algorithms to obtaining high-precision channel information meets the challenges of increasingly high air interface resource overheads and computational complexity, which severely restricts the evolution of massive MIMO. This topic aims to explore new methods and theoretical performance limits for obtaining massive MIMO channel information.
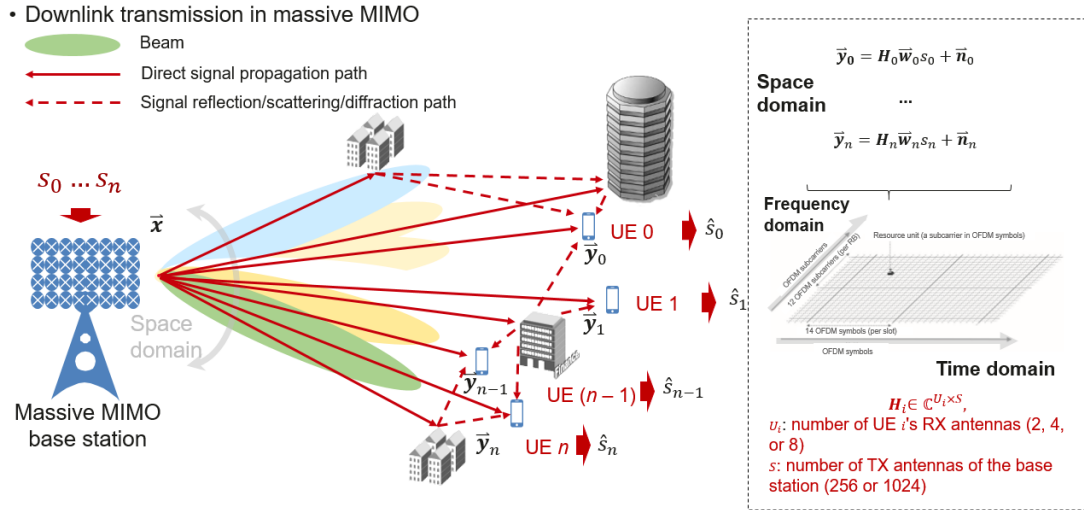

Figure 1.   Downlink transmission in Massive MIMO

2. Current Result

In addition to classical linear methods such as LS and MMSE, the academic community has proposed compressive sensing [1] methods to enhance channel estimation by utilizing sparsity features, and proposed atomic norm and other methods to solve off-grid problems [2][3]. The performance bounds of the sparsity methods have also been studied from an optimization perspective [4]. However, because a wireless channel is not strictly sparse as academically defined, and the complexity of the foregoing algorithm is relatively high, compressed sensing is not well applied to reality.

3. Challenge Description

a)   Challenge Definition:

Massive MIMO channel reconstruction may be described by using an inverse problem 错误!

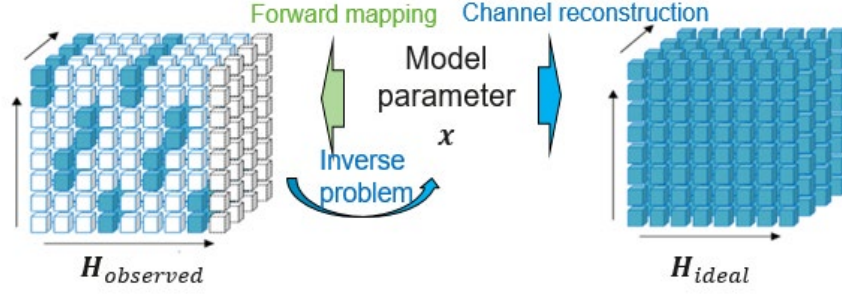未找到引用源。 model shown in Figure 2. The target is to obtain a parameter space with the minimum dimension.

Figure 2. The inverse problem model of channel reconstruction

$$\min_{x,D}(N)$$

$$s.t. H_{est} = D(x)$$
$$||H_{est} - H_{ideal}|| \leq e$$
$$H_{observed} = \mathcal{P}_\Omega (H_{ideal} + n)$$

where

$$x \in \mathbb{C}^N$$
$$D \in \mathbb{C}^N \to \mathbb{C}^{U \times S \times F \times T}$$
$$e \in \mathcal{R}^+$$
$$n \in \mathcal{N}(0, \sigma^2 1)$$
$$[\mathcal{P}_\Omega (H)]_{u,s,f,t} = \begin{cases} H_{u,s,f,t} & \text{if } (u,s,f,t) \in \Omega \\ 0 & \text{if } (u,s,f,t) \notin \Omega \end{cases}$$

And Ω indicates various observation patterns

b) Parameter dimensions

$$U \in [4,8]$$
$$S \in [128,256,1024]$$
$$F \in [1024,2048,4096]$$
$$T \in [10,20,50]$$

4. Demand

a) Theoretical Objectives：

> To build a low-complexity model that reconstructs multi-domain and structured channels: The space of the parameter x for this low-dimensional model and the mapping function D are to be found to reconstruct the target matrix H_{ideal} based on noise-afflicted observation $H_{observed}$. The reconstruction error must be less than or equal to e.

> The algorithms to get the model parameter x based on the noise-afflicted matrix $H_{observed}$

> To analyze the accuracy boundary of the reconstruction: The conditions for meeting $||H_{est} - H_{ideal}|| \leq e$ are to be found. The relationships of reconstruction accuracy e with the number of dimensions N and noise variance σ^2 are to be found.

b) Application Objectives

- ➢ The recommended model and reconstruction algorithm should work well the following scenarios[6][7]:
  - ✓ Channel responses are distributed in a hyperspace spanned by the domains of Doppler, delay, azimuth, tilt angle, and polarization.
  - ✓ The channel response is not ideally sparse in every single dimension, and may even be dense in a single dimension, but is approximately sparse or sparse in the space spanned by multiple domains.
- ➢ Non-ideal observation：
  - ✓ $H_{observed}$ are afflicted by noise due to low SNRs
  - ✓ The initial phases of the observation matrix are unknown and discontinuous in the time domain
- ➢ Compared with the MMSE algorithm, the channel reconstruction based on the recommended model could obtain accuracy gain.
- ➢ The channel reconstruction based on the recommended model has relative low complexity: constant × MMSE estimation > recommended model > LS

- **Reference**

[1]. Christian R. Berger,…, Application of Compressive Sensing to Sparse Channel Estimation, 2010；

[2]. Yingming Tsai,…, Millimeter-Wave Beamformed Full-dimensional MIMO Channel Estimation Based on Atomic Norm Minimization，2017

[3]. JunQuan Deng….MmWave Channel Estimation via Atomic Norm Minimization for Multi-User Hybrid Precoding，2018 (WCNC)；

[4]. E. J. Candès and T. Tao, "Decoding by Linear Programming," IEEE Trans. Info. Theory, vol. 51, no. 12, Dec. 2005

[5]. Tarantola, Albert (2005). "Front Matter" (PDF). Inverse Problem Theory and Methods for Model Parameter Estimation. SIAM

[6]. 3GPP TR 36.873 V12.7.0, Study on 3D channel model for LTE

[7]. 3GPP 3D MIMO channel model: a holistic implementation guideline for open source simulation tools

09 Multi-Channel Joint Resource Allocation Algorithm Based on Mathematical Programming

1. Background

In 5G-NR communication systems, gNodeBs leverage Massive MIMO technology to significantly enhance network capacity. Our focus centers on a pivotal challenge: developing efficient, high-performance air interface channel scheduling algorithms that strike a balance between complexity and performance. Traditional per-channel resource allocation often settles for local optima, falling short of global optimality.

We are exploring an innovative low-complexity approach that jointly optimizes control and shared channel resources. This strategy aims to intelligently consolidate resources for maximizing system throughput while ensuring the algorithm remains simple and efficient, facilitating practical deployment and maintenance without excessive computational overhead. It aspires to chart a new course both theoretically and practically, efficiently unlocking the vast capacity potential of 5G-NR networks.
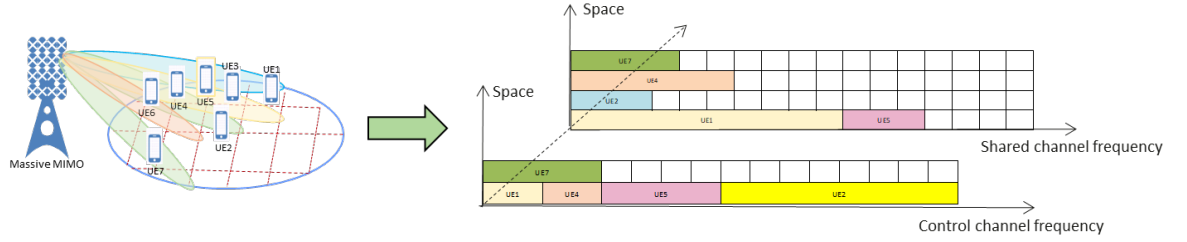
2. Current Result

Conventional joint optimization algorithms tend to employ heuristic approaches, which simplify complexity but often result in substantial gaps from the optimal solution. In contrast, general algorithms based on mathematical solvers strive to approximate the optimal solution through multiple iterations, yet their high complexity hampers efficiency.

Current industry practices largely rely on scalar computations, failing to leverage the vector computing capabilities of modern hardware. Furthermore, traditional algorithms are predominantly serial or employ parallel strategies but with low resource utilization, further limiting the improvement of algorithm performance.

3. Challenge Description

Challenge definition:

$$\max_{A \in \{0,1\}^{U \times C}, B \in \{0,1\}^{U \times R}} \langle x, y \rangle_w$$

$s.t. \ x_u = f(A_{u,:}), u \in \mathbb{U}$

$y_u = g(B, u), u \in \mathbb{U}$

$$g(\boldsymbol{B}, u) = \min\{d_u, \sum_{j=0}^{R-1} \gamma \cdot \log_2\left(1 + s_u \cdot \frac{b_{u,j}}{\|\boldsymbol{B}_{:,j}\|_1} \cdot \prod_{v \in \mathbb{U}, b_{v,j}=1 \text{ and } u!=v} (1 - r_{u,v})\right)\}$$

$$f(\boldsymbol{A}_{u,:})$$
$$= \begin{cases} 1, \text{ where } \|\boldsymbol{A}_{u,:}\|_1 = \max\{j|a_{uj}=1\} - \min\{j|a_{uj}=1\} + 1 = l_u, \min\{j|a_{uj}=1\} \in \mathbb{T}_u \\ 0, \end{cases}$$

$$\|\boldsymbol{B}_{:,j}\|_1 \leq \beta, \ \forall j < R$$

$$\|\boldsymbol{A}_{:,j}\|_1 \leq \alpha, \ \forall j < C$$

$$\forall u, v < U, u \neq v \text{ if } \exists j < C \ s.t. \ a_{uj} = a_{vj} = 1, then \ \mathbb{D}_u \cap \mathbb{D}_v = \emptyset$$

$\boldsymbol{A} \in \{0,1\}^{U \times C}$,and $a_{i,j}$ represents the element on row i and column j in matrix $\boldsymbol{A}$;

$\boldsymbol{B} \in \{0,1\}^{U \times R}$,and $b_{i,j}$ represents the element on row i and column j in matrix $\boldsymbol{B}$

Parameter descriptions:

$\boldsymbol{R} \in \mathbb{N}, \boldsymbol{R} \leq 64$，$\mathbb{S} = \{1,2, \ ..., \ R\}$;

$\boldsymbol{C} \in \mathbb{N}, \boldsymbol{C} \leq 135$，$\mathbb{E} = \{1,2, ..., C\}$;

$\boldsymbol{U} \in \mathbb{N}, \boldsymbol{U} \leq 20, \mathbb{U} = \{1,2, ..., U\}$;

$\boldsymbol{w} \in \mathbb{N}^U, \ \boldsymbol{s} \in \mathbb{N}^U, \boldsymbol{d} \in \mathbb{N}^U$

$\alpha \in \mathbb{N}, \alpha \leq 4$；$\beta \in \mathbb{N}, \beta \leq 8$；$\gamma \in \mathbb{N}$

$\mathbb{F} = \{0,1,2, ...,31\}$,

$$\boldsymbol{V}^{U \times U} = \begin{bmatrix} r_{11} & \cdots & r_{1U} \\ \vdots & \ddots & \vdots \\ r_{U1} & \cdots & r_{UU} \end{bmatrix}, \ 0 \leq r_{i,j} = r_{j,i} \leq 1$$

$l_u \in \{1,2,4,8,16\}$，$\mathbb{T}_u \subset \mathbb{E}$，$\mathbb{D}_u \subset \mathbb{F}$

4. Demand

■ **Theoretical Objectives：**

Investigate the theoretical aspects of matrix computation and parallel computation for discrete optimization problems:

Application of Matrix and Graph theory: Leverage matrix theory and graph theory tools to accelerate the resolution of discrete optimization challenges.

Design and Analysis of Parallel Algorithms: Create algorithms that can be parallelized using matrix operations, suitable for GPUs or distributed systems.

Optimizing Parallel Efficiency versus Communication Overhead: Balance computational efficiency against communication costs, minimizing data exchanges to enhance resource utilization efficiency.

■ **Application Objectives**

Explore low-complexity joint optimization algorithms with a focus on vectorization and parallelization strategies, aiming for the following enhancements:

Vectorized algorithm architecture: Through meticulous algorithm design, ensure direct mapping to efficient vector computations, with vector operations constituting at least 80% of the total algorithm execution, significantly enhancing the speed of compute-intensive tasks.

Parallel algorithm architecture: On the basis of ensuring that at least 80% of system resources are effectively used, parallel algorithms are deployed to achieve a four-fold acceleration effect. That is, the parallel execution time is shortened to one fourth of the serial execution time, and the optimal solution is not less than 80% of the theoretical optimal solution. In this way, the execution time of the algorithm is greatly reduced while the quality of the solution is guaranteed.

● **Reference**

1) P. R. M., M. R., A. Kumar and K. Kuchi, "Downlink Resource Allocation for 5G-NR Massive MIMO Systems,"

   2021 National Conference on Communications (NCC), Kanpur, India, 2021, pp. 1-6, doi:

   10.1109/NCC52529.2021.9530169.

2) Y. -H. Liu and K. C. -J. Lin, "Traffic-Aware Resource Allocation for Multi-User Beamforming," in IEEE

   Transactions on Mobile Computing, vol. 22, no. 6, pp. 3677-3690, 1 June 2023, doi: 10.1109/TMC.2022.3141787.

3) Kepner, J. & Gilbert, J. R. (eds.) (2011). Graph Algorithms in the Language of Linear Algebra (Vol. 22). SIAM.

   ISBN: 978-0-89871-990-1

4) Y. Chen, Y. Wu, Y. T. Hou and W. Lou, "mCore+: A Real-Time Design Achieving ∼ 500 μs Scheduling for 5G MU-

   MIMO Systems,"   2023,

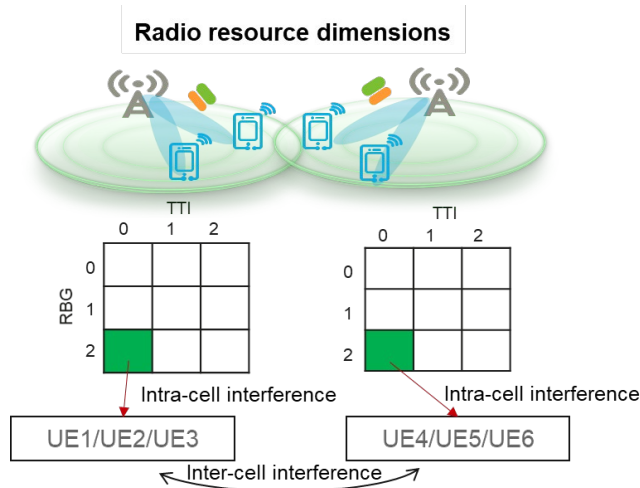# 10 Various AI-based RRM Decision Tasks in Dynamic Wireless Environments

1. Background

Design intelligent architecture and algorithms related to radio resource management (RRM) to optimally allocate radio resources in the time, frequency, space, and power domains and to determine the MCS indexes, ranks, and precoding weights of UEs to be scheduled. The aim is to make optimal resource allocation decision across all dimensions and maximize networkwide experience under a given energy consumption.

2. Current Result

As shown in the figure, radio resources can be classified into the following types:

- Time-frequency resources: Each cell has a certain number of time-frequency resources. Each small grid in the preceding figure represents the smallest time-frequency unit in scheduling.

- Spatial resources: Each cell has multiple beams, and UEs served by different beams can use the same time-frequency resources (generating intra-cell interference). This is referred to as MU pairing.

- Power resources: Power can be allocated to different time-frequency resources or to different UEs using the same time-frequency resource.

Definition of user-perceived rate can be given by

ThpVolDl = $\sum$ ■ (kbits)

Total DL transferred volume = $\sum$ ■ + ▬ (kbits)

参考：3GPP TS32.450

Throughput in DL = ThpVolDl / ThpTimeDl (kbits/s)
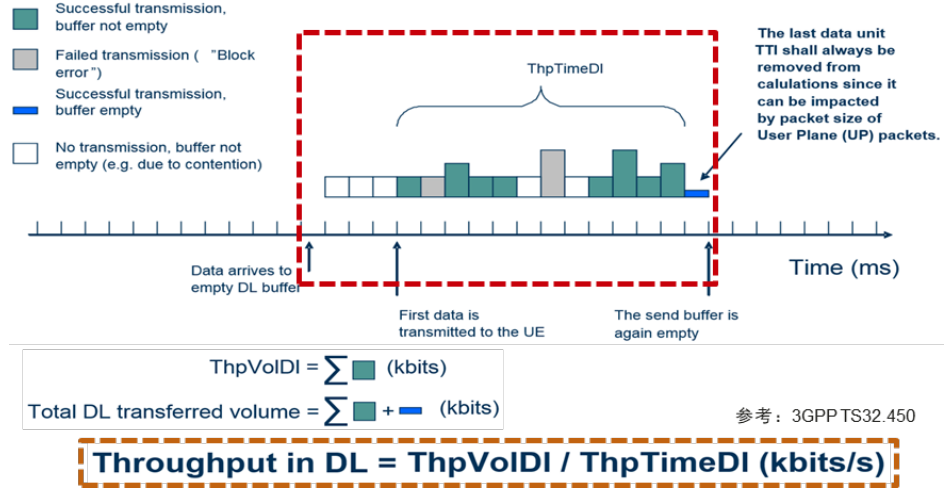
Goal: Improve the user-perceived rate through proper resource orchestration in the time, frequency, space, and power domains.

Baseline approaches:

1. How to determine the scores of UEs on the RBGs? The baseline algorithm uses EPF to calculate the priorities as scores.

2. How to perform scheduling? UEs with the highest priorities are selected in sequence for scheduling by selecting a UE for each RBG or selecting an RBG for each UE.

3. How to perform selection if each RBG can bear multiple UEs? In the baseline approach, scheduling is performed to maximize spatial division spectral efficiency.

3. Challenge Description

The mathematical form of the problem is represented as follows:

Minimum sensing time s. t.	$\min_{P,X,V} \|Y\|_1$
Sensing time	$Y = sign(Q - M)$ $\quad Y \in \{0,1\}^{N_2 \times N_4}$
Residual buffer	$Q_{:,t+1} = Q_{:,t} - M_{:,t} + \lambda_{:,t}$ $\quad Q; M; \lambda \in R^{N_2 \times N_4}$
Data for scheduling	$M = g(R)$
SINR tensor	$R = f(H,P,X,V)$ $\quad R; P \in R^{N_1 \times N_2 \times N_3 \times N_4 \times N_6}$
Energy consumption constraints	$h(P,X,V) \le \delta_1 \mathbf{1}_{N_1 \times N_4}$ $\quad X \in \{0,1\}^{N_1 \times N_2 \times N_3 \times N_4 \times N_6}$
	$\quad H; V \in \mathbb{C}^{N_1 \times N_2 \times N_3 \times N_4 \times N_5 \times N_6}$
Constraint on per-RB power	$P_{XV} \cdot_{2,6} \mathbf{1}_{N_1 \times N_2 \times N_3 \times N_4 \times N_6} \le \delta_2 \mathbf{1}_{N_1 \times N_3 \times N_4}$
Constraint on per-cell power	$P_{XV} \cdot_{2,3,6} \mathbf{1}_{N_1 \times N_2 \times N_3 \times N_4 \times N_6} \le \delta_3 \mathbf{1}_{N_1 \times N_4}$
Constraint on the sum of power in the time domain	$P_{XV} \cdot_{2,3,4,6} \mathbf{1}_{N_1 \times N_2 \times N_3 \times N_4 \times N_6} \le \delta_3 \mathbf{1}_{N_1}$
	$P_X = P \odot X \quad P_{XV} = P \odot X \odot \|V \cdot_5 V\|$

where $f(\cdot)$ is

SINR relationship	$f(H,P,X,V) = A/C + \Theta(Mcs_{lately\ reported})$
Numerator of SINR	$A = P_X \odot \|H \cdot_5 V\|^2$
	$H_{(i)} = H_{:,i,:,:,:,:} \times_2 \mathbf{1}_{1 \times N_2}$ $\quad H_{:,i,:,:,:,:} \in \mathbb{C}^{N_1 \times 1 \times N_3 \times N_4 \times N_5 \times N_6}$
	$\quad H_{(i)} \in \mathbb{C}^{N_1 \times N_2 \times N_3 \times N_4 \times N_5 \times N_6}$
	$B = \left(P_X \odot \|H_{(i)} \cdot_5 V\|^2 \right) \cdot_{12} \mathbf{1}_{N_1 \times N_2 \times N_3 \times N_4 \times N_6} \in \mathbb{C}^{N_3 \times N_4}$
Denominator of SINR	$C = \mathbf{1}_{N_1 \times N_2} \circ B - A + \sigma^2 \mathbf{1}_{N_1 \times N_2 \times N_3 \times N_4 \times N_6}$

Θ is an uncertain parameter and is affected by inaccurate H measurement or receiver. It does not indicate additive effect.

And $h(\cdot)$ is

$$h(\boldsymbol{P}, \boldsymbol{X}, \boldsymbol{V}) = P_{static}\boldsymbol{D} + \varepsilon \boldsymbol{P}_{XV} \cdot_{2,3,6} \mathbf{1}_{N_1 \times N_2 \times N_3 \times N_4 \times N_6}$$

Number of activated antennas

$$d_{k,t} = \left\| \boldsymbol{V}_{k,:,:,t,:,:} \cdot_{2,3,6} \boldsymbol{V}_{k,:,:,t,:,:} \right\|_0 \quad \boldsymbol{D} = tensor(d_{k,t}) \in R^{N_1 \times N_4}$$

And $g(\cdot)$ can be calculated as follows

a) Calculate $10 log_{10}\left(\dfrac{R \cdot_{3,6} \mathbf{1}_{N_1 \times N_2 \times N_3 \times N_4 \times N_6}}{X \cdot_{3,6} \mathbf{1}_{N_1 \times N_2 \times N_3 \times N_4 \times N_6}}\right)$ as the SINR of a UE.

b) Quantize the SINR to an MCS index.

SINR(db)	-6.55	-4.51	-2.8	-0.84	0.98	2.63	4.69	5.59	6.53	7.5
Mcs	0	1	2	3	4	5	6	7	8	9
SINR(db)	8.38	8.92	10.26	11.11	12.03	12.91	13.94	14.96	15.91	16.93
Mcs	10	11	12	13	14	15	16	17	18	19
SINR(db)	18.02	18.93	19.56	20.46	21.45	22.32	23.74	24.54	25.43	
Mcs	20	21	22	23	24	25	26	27	28	

c) Map the MCS index to the data volume.

$$\boldsymbol{M} = \left\{ 144 * 16 * eff * \left(\boldsymbol{X} \cdot_{3,6} \mathbf{1}_{N_1 \times N_2 \times N_3 \times N_4 \times N_6} \right) \right\} \cdot_1 \mathbf{1}_{N_1 \times N_2 \times N_4}$$

The mapping from Mcs to eff is as follows:

Mcs	0	1	2	3	4	5	6	7	8	9
eff	0.1523	0.2344	0.377	0.6016	0.877	1.1758	1.4766	1.6953	1.9141	2.1602
Mcs	10	11	12	13	14	15	16	17	18	19
eff	2.4063	2.5703	2.7305	3.0293	3.3223	3.6094	3.9023	4.2129	4.5234	4.8164
Mcs	20	21	22	23	24	25	26	27	28	
eff	5.1152	5.332	5.5547	5.8906	6.2266	6.5703	6.9141	7.1602	7.4063	

The correlation operators are defined as follows:

:, in $\boldsymbol{Q}_{:,t+1}$ indicates that all elements in the first dimension of $\boldsymbol{Q}$ are not affected

$\boldsymbol{A} \cdot_i \boldsymbol{B}$: tensor product $\boldsymbol{A} \cdot_i \boldsymbol{B} = \sum_{i'=1}^{N_i} (\boldsymbol{A})_{:,\dots,i',\dots,:} \odot (\boldsymbol{B})^*_{:,\dots,i',\dots,:}$

∘: outer product

$\times_n$: modular multiplication (modulo n)

$\odot$: dot product

Descriptions of Variables and Their Dimensions can be given by

$N_1 = 36$

$N_2 = 72$ -- Indicates the number of UEs to be scheduled.

$N_3 = 17$ -- Indicates the number of RBGs in each cell.

$N_4 = 100$ -- Indicates the number of slots to be considered.

-- Indicates the number of antennas.

-- Indicates the number of layers of a UE.

$\boldsymbol{Y} \in \{0,1\}^{N_2 \times N_4}$ -- $y_{i,t}$ indicates whether the ith UE has data in the tth TTI.

$\boldsymbol{Q} \in R^{N_2 \times N_4}$ -- $Q_{i,t}$ indicates the residual buffer of the ith UE in the tth TTI.

$\boldsymbol{M} \in R^{N_2 \times N_4}$ -- $M_{i,t}$ indicates the data to be scheduled for the ith UE in the tth TTI.

$\lambda \in R^{N_2 \times N_4}$ -- $\lambda_{i,t}$ indicates the new data for the ith UE in the tth TTI.

$\boldsymbol{R} \in R^{N_1 \times N_2 \times N_3 \times N_4 \times N_6}$ -- The elements indicate the SINR of the lth layer in the tth TTI in the bth RBG of the ith UE in the kth cell.

$\boldsymbol{P} \in R^{N_1 \times N_2 \times N_3 \times N_4 \times N_6}$ -- The elements indicate the transmit power of the lth layer in the tth TTI in the bth RBG of the ith UE in the kth cell.

$\boldsymbol{X} \in \{0,1\}^{N_1 \times N_2 \times N_3 \times N_4 \times N_6}$ -- The elements indicate whether resources are allocated to the lth layer in the tth TTI in the bth RBG of the ith UE in the kth cell.

$\boldsymbol{H} \in \mathbb{C}^{N_1 \times N_2 \times N_3 \times N_4 \times N_5 \times N_6}$ -- The elements indicate the channel fading factor of the lth layer of the ath antenna in the tth TTI in the bth RBG of the ith UE in the kth cell.

$\boldsymbol{V} \in \mathbb{C}^{N_1 \times N_2 \times N_3 \times N_4 \times N_5 \times N_6}$ -- The elements indicate the weighting factor of the lth layer of the ath antenna in the tth TTI in the bth RBG of the ith UE in the kth cell.

$\boldsymbol{\theta} \in R^{N_1 \times N_2 \times N_3 \times N_4 \times N_6}$ -- The elements indicate the SINR error of the lth layer in the tth TTI in the bth RBG of the ith UE in the kth cell.

$\boldsymbol{D} \in R^{N_1 \times N_4}$ -- $d_{i,t}$ indicates the number of activated antennas in the tth TTI of the kth cell.

Technical challenges:

- Incomplete information or inaccurate modeling: AI supervised learning labels are difficult to obtain. The channel information is inaccurate, modelling of terminal receiver demodulation is difficult, and it is difficult to obtain the labels by using the optimization solver. How to interact with the environment and improve modeling (especially the $f(\cdot)$ and $g(\cdot)$ modeling) to

eventually optimize network performance?

    a) Factors such as inaccurate channel information make optimization modeling difficult and the derived result is significantly different from the real optimal solution. When the weight or interference is inaccurate, optimization causes more interference and decreases network performance.

    b) After MU pairing, power allocation, MCS index, rank, and multi-cell coordination are introduced, the spectral efficiency modeling formula cannot reflect the actual benefits.

    c) AMC policy: The current unified OLLA policy saves memory at the cost of performance.

    d) Other factors: The pairing interference estimated by the base station is different from the actual demodulated interference at the receive end.

- Distributed computing: The information interaction between multiple cells is insufficient, and interaction delay exists. How to design a distributed algorithm with low communication overhead?

- High dimensionality: How does AI deal with high dimensionality to ensure that the output is within the feasible region?

- High decision-making space: How to ensure optimal allocation of radio resources such as time, frequency, space, and power resources in multiple cells? How to determine factors such as the MCS indexes, ranks, and precoding weights of UEs to be scheduled? With various decision-making space dimensions, convergence using traditional AI is slow or even impossible.

- High computing power requirements: As the number of cells or UEs increases, the computing complexity increases. The time complexity must be lower than 300 μs given the current product computing power.

4. Demand

- Goal: Maximize user experience on the entire network while ensuring that the energy consumption does not exceed a certain level.
- Input: items sampled within a period of time: estimated channel H and MCS index of each UE in each RBG of each cell, static and dynamic power consumption of modules, and bandwidth of each cell
- Output: occupation of each RBG by each UE in each cell and the related power; MCS index, number of antennas shut down, number of time-domain resources shut down, and precoding weight of each UE in each cell
- Constraints: power, resource, and data volume constraints of modules
- Uncertainty: When scheduling is performed based on the determined MCS index, TBs may

experience exceptions at different probabilities.

- Requirements: Keep the time complexity at 300 µs when there are 6 frequency bands on 2.2 GHz (6 cells and 17 RBGs per band, totaling 72 UEs), and 4 core CPUs.

11 AI_AI-enabled Full Real-Time CSI Acquisition in the HBF Architecture

1. Background

MIMO(Multi-Input Multi-Output) can significantly improve the performance and efficiency of wireless communication system via spatial beamforming and spatial multiplexing. But as antenna number becoming more and more, the system cost will be increased badly. A more economical option is to adopt Hybrid beamforming architecture as shown in figure1,here N_t and N_r is the antenna number of transmitter and receiver respectively, N_{RF} is the corresponding number of digital channel or RF chain, generally $N_t/N_r > N_{RF}$. $\mathbf{F}^{DB}$ and $\mathbf{F}^{AB}$ is the digital precoder and analog precoder at transmitter side, $\mathbf{W}^{DB}$ is $\mathbf{W}^{AB}$ the digital precoder and analog precoder at receiver side. Generally analog precoder can only change the phase of input signal, while digital precoder can change both the phase and amplitude of input signal. The research topic is how to efficiently acquire the full real-time Channel State Information (CSI) $\mathbf{H}$ in HBF system.
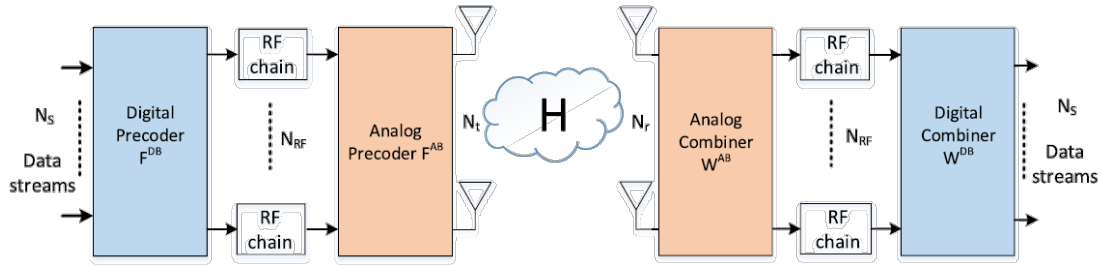

figure 1 transmitter and receiver description in HBF system [1]

2. Current Result

As shown in Figure2, two methods are generally used in existing wireless communication systems to complete the measurement of channel state information $\mathbf{H}$. One is to send SRS measurement signals through the terminal, and the base station directly completes the uplink channel information measurement $\mathbf{H}_{UL}$; the other is to send CSI-RS measurement signals through the base station, the terminal first completes the downlink channel information measurement $\mathbf{H}_{DL}$, and then quantizes the $\mathbf{H}_{DL}$ and feeds it back to the base station. For the TDD system, $\mathbf{H}_{UL} = \mathbf{H}_{DL}{}^H$ is satisfied for the same time. In order to complete the full measurement within the coverage area of the base station, analog precoder generally uses DFT (Discrete Fourier Transform) precoding weights to sequentially scan the coverage area, which is called Analog beam scan. Digital precoder uses the unit matrix to directly obtain the original channel measurements. The measurement overhead of this method is huge, and corresponding measurement resources need to be allocated to all users and locations that need to be measured (mainly including time domain and frequency domain). The actual system can only achieve some sparse measurement, and the default part is maintained through adjacent measurement results, but the accuracy is greatly reduced, thus significantly reducing system performance. How to recover the full amount of high-precision channel information through a small amount of sparse measurements has always been a research hotspot and needs to be broken through.
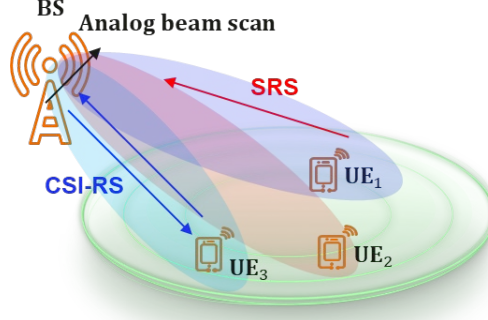
Figure2 Schematic diagram of channel measurement in HBF system

3. Challenge Description

The problems discussed above can be abstractly refined into the following mathematical problems, $\mathbf{X} \in \mathbb{C}^{N_1 \times N_2 \times N_3 \times N_4}$ is a known 4-dimensional complex tensor. For any positive integer $n_1 \in [1, N_1], n_2 \in [1, N_2]$, the measurement of $\mathbf{X}(n_1, n_2, :, :)$ is as follows:

$$\mathbf{Y}(n_1, n_2, :, :) = \mathbf{W}(n_1, :, :)\mathbf{X}(n_1, n_2, :, :)\mathbf{V}(n_1, :, :) + \mathbf{N}(n_1, n_2, :, :) \in \mathbb{C}^{M_3 \times M_4}$$

where $\mathbf{N} \in \mathbb{C}^{N_1 \times N_2 \times M_3 \times M_4}$ represents independent complex white Gaussian noise. That is, any positive integer $n_1 \in [1, N_1], n_2 \in [1, N_2], n_3 \in [1, M_3], n_4 \in [1, M_4]$ is subject to the following: $\mathbf{N}(n_1, n_2, n_3, n_4) \sim \mathcal{CN}(0, \sigma_n^2)$. $\mathbf{W}(n_1, :, :) \in \mathbb{C}^{M_3 \times N_3}$ and $\mathbf{V}(n_1, :, :) \in \mathbb{C}^{N_4 \times M_4}$ respectively represent two independent dimension reduction matrices and the following hold true: $M_3 < N_3$ and $N_4 > M_4$. Under a given value of n_1 and different values of n_2, $\mathbf{X}(n_1, n_2, :, :)$ uses the same dimension reduction weights $\mathbf{W}(n_1, :, :)$ and $\mathbf{V}(n_1, :, :)$; in addition, for any element in $\mathbf{W}$ and $\mathbf{V}$, the following hold true: $|\mathbf{W}(i, j, k)| = 1$ and $|\mathbf{V}(i, j, k)| = 1$. For $(n_1, n_2) \in \mathbf{\Omega}_1$, under a given value of n_1, there is a compression quantization function $g: \mathbb{C}^{|\mathbf{\Omega}_{1, n_2}| \times M_3 \times M_4} \to \mathbf{Q}(n_1, :)$, where $\mathbf{Q}$ indicates 0/1 bit sequence with a length of $|\mathbf{Q}(n_1, :)|$.

Research goals are described as follows:

1. Design the group $(n_1, n_2) \in \mathbf{\Omega}_1$ as well as the corresponding $\mathbf{W}(n_1, :, :)$ and $\mathbf{V}(n_1, :, :)$

   dimension reduction matrix sets as follows:

$$\mathbf{\theta}_1 = \left\{ \mathbf{\Omega}_1 = \{(n_1, n_2)\}, \{\mathbf{W}(n_1, :, :), \mathbf{V}(n_1, :, :)\}_{(n_1, :) \in \mathbf{\Omega}_1} \right\}$$

Design the compression quantization function $g: \mathbb{C}^{|\mathbf{\Omega}_{1, n_2}| \times M_3 \times M_4} \to \boldsymbol{Q}(n_1, :)$ where

$$\boldsymbol{Q} = \left\{ g\big(\{\mathbf{Y}(n_1, n_2, :, :)\}_{(n_1 n_2) \in \mathbf{\Omega}_1}\big) \right\}_{(n_1, :) \in \mathbf{\Omega}_1}$$

2. Design the group $(n_1, n_2) \in \mathbf{\Omega}_2$ as well as the corresponding $\mathbf{W}(n_1, :, :)$ and $\mathbf{V}(n_1, :, :)$

   dimension reduction matrix sets as follows:

$$\mathbf{\theta}_2 = \left\{ \mathbf{\Omega}_2 = \{(n_1, n_2)\}, \{\mathbf{W}(n_1, :, :), \mathbf{V}(n_1, :, :)\}_{(n_1, :) \in \mathbf{\Omega}_2} \right\}$$

The corresponding measurement is as follows:

$$\boldsymbol{\mathcal{O}} = \{\mathbf{Y}(n_1, n_2, :, :)\}_{(n_1 n_2) \in \mathbf{\Omega}_2}$$

3. Design the mapping function $f: \left\{ \mathbb{C}^{|\mathbf{\Omega}_{2, n_1}| \times |\mathbf{\Omega}_{2, n_2}| \times M_3 \times M_4}, \{0, 1\}^{|\boldsymbol{Q}|} \right\} \to \mathbb{C}^{N_1 \times N_2 \times N_3 \times N_4}$, and record

   the mapping result as follows:

$$\hat{\mathbf{X}} = f(\boldsymbol{\mathcal{O}}, \boldsymbol{Q})$$

4. Based on the design in steps 1, 2, and 3, resolve the following target problem:

$$\min_{\boldsymbol{\theta}_1,\boldsymbol{\theta}_2,g,f} \left\| \mathbf{X} - \widehat{\mathbf{X}} \right\|^2 + (\alpha \cdot |\boldsymbol{\Omega}_1| + \beta \cdot |\boldsymbol{\Omega}_2|) + \gamma \cdot |\boldsymbol{Q}|$$

$$s.t. \quad \widehat{\mathbf{X}} = f(\boldsymbol{\mathcal{O}}, \boldsymbol{Q}), \quad f \colon \left\{ \mathbb{C}^{|\Omega_{2,n_1}| \times |\Omega_{2,n_2}| \times M_3 \times M_4}, \{0,1\}^{|\boldsymbol{Q}|} \right\} \to \mathbb{C}^{N_1 \times N_2 \times N_3 \times N_4}$$

$$\boldsymbol{\mathcal{O}} = \{ \mathbf{Y}(n_1, n_2, :, :) \}_{(n_1 n_2) \in \boldsymbol{\Omega}_2} \in \mathbb{C}^{|\Omega_{2,n_1}| \times |\Omega_{2,n_2}| \times M_3 \times M_4}$$

$$\boldsymbol{Q} = \left\{ g\!\left(\{ \mathbf{Y}(n_1, n_2, :, :) \}_{(n_1 n_2) \in \boldsymbol{\Omega}_1} \right) \right\}_{(n_1, :) \in \boldsymbol{\Omega}_1}, g \colon \mathbb{C}^{|\Omega_{1,n_2}| \times M_3 \times M_4} \to \{0,1\}^{|\boldsymbol{Q}(n_1, :)|}$$

$$\forall (n_1, n_2), \mathbf{Y}(n_1, n_2, :, :) = \mathbf{W}(n_1, :, :) \mathbf{X}(n_1, n_2, :, :) \mathbf{V}(n_1, :, :) + \mathbf{N}(n_1, n_2, :, :) \in \mathbb{C}^{M_3 \times M_4}$$

$$\boldsymbol{\theta}_1 = \left\{ \boldsymbol{\Omega}_1 = \{(n_1, n_2)\}, \{ \mathbf{W}(n_1, :, :), \mathbf{V}(n_1, :, :) \}_{(n_1, :) \in \boldsymbol{\Omega}_1} \right\}$$

$$\boldsymbol{\theta}_2 = \left\{ \boldsymbol{\Omega}_2 = \{(n_1, n_2)\}, \{ \mathbf{W}(n_1, :, :), \mathbf{V}(n_1, :, :) \}_{(n_1, :) \in \boldsymbol{\Omega}_2} \right\}$$

$$\forall (i, j, k), \quad |\mathbf{W}(i, j, k)| = |\mathbf{V}(i, j, k)| = 1$$

α, β, γ are given scalar weight coefficients.

5. Based on the problem in step 4, the following **subproblem 1** can be derived with given values

for the parameters $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ assumed:

$$\min_{g,f} \left\| \mathbf{X} - \widehat{\mathbf{X}} \right\|^2 + \gamma \cdot |\boldsymbol{Q}|$$

$$s.t. \quad \widehat{\mathbf{X}} = f(\boldsymbol{\mathcal{O}}, \boldsymbol{Q}), \quad f \colon \left\{ \mathbb{C}^{|\Omega_{2,n_1}| \times |\Omega_{2,n_2}| \times M_3 \times M_4}, \{0,1\}^{|\boldsymbol{Q}|} \right\} \to \mathbb{C}^{N_1 \times N_2 \times N_3 \times N_4}$$

$$\boldsymbol{\mathcal{O}} = \{ \mathbf{Y}(n_1, n_2, :, :) \}_{(n_1 n_2) \in \boldsymbol{\Omega}_2} \in \mathbb{C}^{|\Omega_{2,n_1}| \times |\Omega_{2,n_2}| \times M_3 \times M_4}$$

$$\boldsymbol{Q} = \left\{ g\!\left(\{ \mathbf{Y}(n_1, n_2, :, :) \}_{(n_1 n_2) \in \boldsymbol{\Omega}_1} \right) \right\}_{(n_1, :) \in \boldsymbol{\Omega}_1}, g \colon \mathbb{C}^{|\Omega_{1,n_2}| \times M_3 \times M_4} \to \{0,1\}^{|\boldsymbol{Q}(n_1, :)|}$$

$$\forall (n_1, n_2), \mathbf{Y}(n_1, n_2, :, :) = \mathbf{W}(n_1, :, :) \mathbf{X}(n_1, n_2, :, :) \mathbf{V}(n_1, :, :) + \mathbf{N}(n_1, n_2, :, :) \in \mathbb{C}^{M_3 \times M_4}$$

6. Based on the problem in step 5, the following **subproblem 2** can be derived with a given

value of $g \colon \mathbb{C}^{|\Omega_{1,n_2}| \times M_3 \times M_4} \to \{0,1\}^{|\boldsymbol{Q}(n_1, :)|}$ assumed:

$$\min_{f} \left\| \mathbf{X} - \widehat{\mathbf{X}} \right\|^2$$

$$s.t. \quad \widehat{\mathbf{X}} = f(\boldsymbol{\mathcal{O}}, \boldsymbol{Q}), \quad f \colon \left\{ \mathbb{C}^{|\Omega_{2,n_1}| \times |\Omega_{2,n_2}| \times M_3 \times M_4}, \{0,1\}^{|\boldsymbol{Q}|} \right\} \to \mathbb{C}^{N_1 \times N_2 \times N_3 \times N_4}$$

$$\boldsymbol{\mathcal{O}} = \{ \mathbf{Y}(n_1, n_2, :, :) \}_{(n_1 n_2) \in \boldsymbol{\Omega}_2} \in \mathbb{C}^{|\Omega_{2,n_1}| \times |\Omega_{2,n_2}| \times M_3 \times M_4}$$

$$\boldsymbol{Q} = \left\{ g\!\left(\{ \mathbf{Y}(n_1, n_2, :, :) \}_{(n_1 n_2) \in \boldsymbol{\Omega}_1} \right) \right\}_{(n_1, :) \in \boldsymbol{\Omega}_1}, g \colon \mathbb{C}^{|\Omega_{1,n_2}| \times M_3 \times M_4} \to \{0,1\}^{|\boldsymbol{Q}(n_1, :)|}$$

$$\forall (n_1, n_2), \mathbf{Y}(n_1, n_2, :, :) = \mathbf{W}(n_1, :, :) \mathbf{X}(n_1, n_2, :, :) \mathbf{V}(n_1, :, :) + \mathbf{N}(n_1, n_2, :, :) \in \mathbb{C}^{M_3 \times M_4}$$

Appendix

Table 1. Typical value ranges of parameters

Parameter	Value Range
N_1	1 to 100
N_2	1000~5000
N_3	100~10000

N_4	$1\sim100$
M_3	$10\sim1000$
M_4	$1\sim10$

4. Demand

■ **Theoretical Objectives：**

(1) From the perspective of information recoverability, the minimum measurement overhead required for a given recovery accuracy is theoretically analyzed.

(2) From the perspective of CSI-RS and SRS fusion measurement, corresponding to given measurements Q and O. When the respective cost weights α and β are given, the optimal combined measurement strategy is theoretically analyzed.

■ **Application Objectives**

(1) The measurement ratio corresponding to sparse measurement Q is about 1%, the measurement ratio corresponding to sparse measurement O is about 5%, and the number of feedback bits for each measurement is <100, the full channel information recovery accuracy NMSE <-10dB.

• **Reference**

[1] Ahmed I, Khammari H, Shahid A, et al. A survey on hybrid beamforming techniques in 5G: Architecture and system model perspectives[J]. IEEE Communications Surveys & Tutorials, 2018, 20(4): 3060-3097.

12 Acceleration of Score-based diffusion process

1. Background

The score-based generative models have become a widely-used option for image/video generation. However, these models usually require multiple denoising steps (usually>50) to ensure the quality of the generated images, which inevitably introduces high inference cost. Therefore the acceleration of the diffusion process has become a critical research problem.

2. Current Result

The diffusion process can be expressed as:[1]

$$dx = f(x; t) - \eta_t \nabla_x \log p_t(x) \approx f(x; t) - \eta_t G_\theta(x; t) \quad (1)$$

Here $\nabla_x \log p_t(x)$ is the gradient of the log-likelihood function for the probability of the state in the diffusion process. Since we do not have the full access of the data distribute, usually people utilize a trained network $G_\theta(x; t)$ to approximate $\nabla_x \log p_t(x)$. Afterwards, the denoising process (1) can be used for the sampling from a Guassian distribution $p_1(x)$ $(t = 1)$ to real data distribution $p_0(x)$ $(t = 0)$.

Right now there are two major directions for accelerating the denoising process. One is the design of advanced sampler, such as iPNDM, DPM-Solver; Another direction is to optimize the sampling interval, such as AYS, AMED.

3. Challenge Description

Regarding (1), assuming $f(x; t) = \alpha_t x + \beta_t t$, $G_\theta(x; t)$ is a deep neural network, given the discrete sampling number N (usually between 2 to 100 steps), output the optimized sampling strategy, including the sampler and sampling intervals.

4. Demand

1) Sampler Optimization: Compared to the DPM-Solver series, the number of sampling steps should be reduced by more than 25%. The generated quality (referencing metrics like Clip Score, FID, Hpsv2, and human evaluation) should not deteriorate.

2) Sampling Interval Optimization: Output the optimized sampling interval strategy:

$$t_i = schedule(i)$$

to minimize the final diffusion error: $D := E_{x_0 \sim p_0} ||x_1^{discrete} - x_1^{continuous}||^2$ where

$x_1^{discrete}$ is the final result using the optimized sampling intervals, and $x_1^{continuous}$ is the result obtained by continuous sampling (usually simulated with *N*=1000).

Either of the two objectives can be achieved. For the second objective, the number of samples for $G_\theta(x; t)$ should be controlled within 10,000 samples. The specific network selection can refer to SD1.5-XL.

- **Reference**

[1] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano

Ermon, Ben Poole, Score-Based Generative Modeling through Stochastic Differential

Equations, https://arxiv.org/abs/2011.13456

13 Accelerated Iterative Algorithm for Solving Large Complex Sparse Matrix

1. Background

Many simulation problems need to solve large systems of linear algebraic equations. One class of linear systems is a 4D large sparse linear equations obtained by Wilson-Dirac discretization method $Ma=b$, where M is a large complex sparse symmetric matrix and each non-zero element in M is a small complex matrix rather than a floating-point number, that is, for a example, having 4D space (x, y, z, t), every location in n at $a(n)$ requires solution of an equation $M(n)a(n) = b(n)$, where $M(n)$ is built by the Kronecker product of the 2 directions u_μ and corresponding γ_μ for the all 4 dimensions. For details, see the formula shown in the Figure below.

The solution of such systems of linear algebraic equations is common for quantum chromodynamics (QCD), for example, let us take lattice gauge (32, 32, 32, 8), the size of the complex sub-matrix in each lattice point is 12×12. Solving such large systems of linear equations is a typical memory-constrained problem. We believe here is a huge room for optimization leveraging symmetry, cutting-edge numerical algorithms and hardware platform characteristics.

One of the common iterative methods used for solving the system is BiCGstab with multigrid preconditioning. In a typical case, the dimension of the coefficient matrix can be of size $O(10^9)$. In this case, the required computing resources are large, so it is urgent and critical to accelerate the solution of this type of systems.

$$u_\mu(n) = \begin{bmatrix} u_{\mu,1,1} & u_{\mu,1,2} & u_{\mu,1,3} \\ u_{\mu,2,1} & u_{\mu,2,2} & u_{\mu,2,3} \\ u_{\mu,3,1} & u_{\mu,3,2} & u_{\mu,3,3} \end{bmatrix} \in \mathbb{C}^{3\times3}, a(n) = \begin{bmatrix} a_{1,1} \\ \vdots \\ a_{12,1} \end{bmatrix}, b(n) = \begin{bmatrix} b_{1,1} \\ \vdots \\ b_{12,1} \end{bmatrix}$$

$$b(n) = M(n)a(n), M(n) \propto -\frac{1}{2}\sum_{\mu=\pm1}^{4} (I - \gamma_\mu) \otimes u_\mu a(n \pm \mu)$$

2. Current Result

The iterative solution algorithm for this scenario has three difficulties:

1) **Special structure:** matrix M is non-dominant, source term\ b has only one element with 1+0i，while the remaining ones are zeros;

2) **Long iteration time:** the smaller the eigenvalues of the matrix M, the more iterations are needed to convergence (critical challenge) of algorithms such as BiCGstab. As a result, the solution time is big being a performance bottleneck for QCD applications.

3) **Preconditioning selection:** Choose an appropriate preconditioning technique to accelerate the iterative algorithm in different use cases. Preconditioning itself takes a lot

of time, so special research and design is required to address this challenge as well.

There are two solutions: numerical solution and AI model.

- **Preconditioned BiCGstab:** In the model example with lattice (32, 32, 32, 8) it takes about 216 seconds on a single ARM-based server with 188 GB/s memory bandwidth and 1 TFlops peak performance. to calculate 12 groups of sources in about 240 iteration steps.

- **Using AI to define the initial value for iteration:** Usage of the convolutional AI model with inputs b and u provides predictions in a range (1e-3, 1) for a with a relative error for a trace <1e-2. The predicted result is used as the initial value for BiCGStab iteration. The number of iteration steps decreases by 9% that is still far from the speed-up needed.

3. Challenge Description

Provide an efficient and strongly generalized iteration solution based on the calculation example, including but not limited to numerical algorithms and AI technologies. The performance is better than that of the existing solution.

4. Demand

- **Iterative method:** Research and develop an efficient and general iterative method for the solution of lattice QCD problems adjusting numerical algorithm, applying AI-DNN technology, etc. with the goal to have the relative error in the calculated trace is less than 1e-2.

- **Reduce solution time:** Achieve the 30% iteration time reduction on a given ARM architecture compared against preconditioned BiCGstab method,

- **High-energy physics scenario verification:** Apply the proposed method in QCD and verify its consistency with the anticipated modeling results.
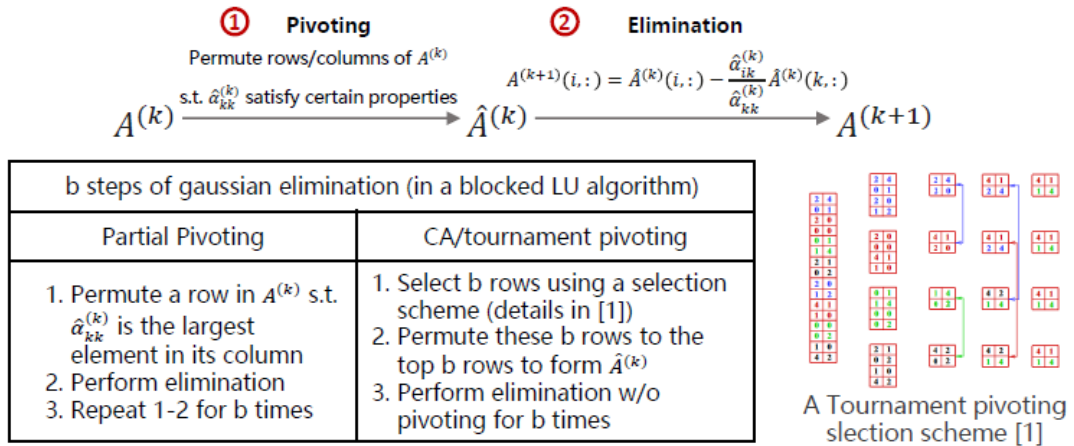
- **Reference**

[1]   Sun W., Bi Y., Cheng Y.. Lattice QCD Calculation and Optimization on ARM Processor, J. Computer Science, 2023, 50(6): 52-57. https://doi.org/10.11896/jsjkx.230200159

[2] Chen Y., Ding H.T., Feng X., et al.   Lattice quantum chromodynamics in China. Knowledge of Modern Physics, 2020, 32 (1):36-44.

14 Numerical Stability Analysis of Communication Avoiding LU (CALU) factorization
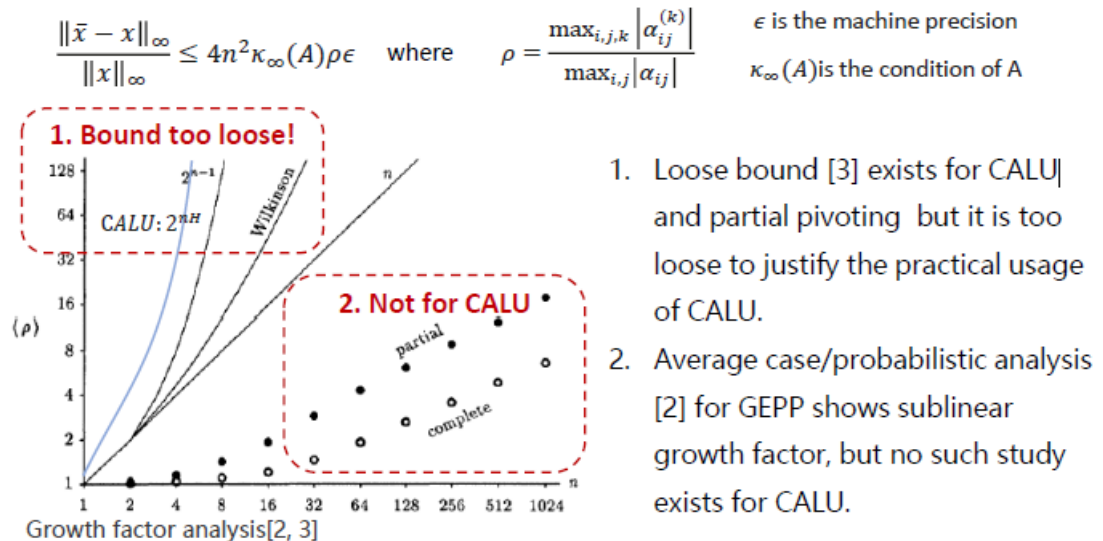
1. Background

Solving system of linear equations are at the heart of numerical linear algebra and dense solvers are widely used in industrial applications. Pivoting is one of the key steps for numerical stable dense linear algebra solvers. Gaussian elimination with partial pivoting (GEPP) is one of the most stable and widely used algorithm for LU factorization. However, for large scale application that uses cluster wide solvers, communication cost associated with partial pivoting is an increasing bottleneck. Communication Avoiding LU factorization (CALU) is thus proposed to alleviate the communication bottleneck. It mainly uses a ca pivoting scheme to reduce comm. overhead.

At the k-th step, Gaussian Elimination with a pivoting scheme can be expressed as follows:

① Pivoting

Permute rows/columns of $A^{(k)}$

s.t. $\hat{a}_{kk}^{(k)}$ satisfy certain properties

② Elimination

$$A^{(k+1)}(i,:) = \hat{A}^{(k)}(i,:) - \frac{\hat{a}_{ik}^{(k)}}{\hat{a}_{kk}^{(k)}} \hat{A}^{(k)}(k,:)$$

$$A^{(k)} \longrightarrow \hat{A}^{(k)} \longrightarrow A^{(k+1)}$$

b steps of gaussian elimination (in a blocked LU algorithm)	
Partial Pivoting	CA/tournament pivoting
1. Permute a row in $A^{(k)}$ s.t. $\hat{a}_{kk}^{(k)}$ is the largest element in its column 2. Perform elimination 3. Repeat 1-2 for b times	1. Select b rows using a selection scheme (details in [1]) 2. Permute these b rows to the top b rows to form $\hat{A}^{(k)}$ 3. Perform elimination w/o pivoting for b times

A Tournament pivoting slection scheme [1]

2. Current Result

Wilkinson proved that the relative error of solving $Ax=b$, where A is factorized with LU, is related to the growth factor ρ[2]:

$$\frac{\|\bar{x} - x\|_\infty}{\|x\|_\infty} \leq 4n^2 \kappa_\infty(A) \rho \epsilon \quad \text{where} \quad \rho = \frac{\max_{i,j,k} \left| \alpha_{ij}^{(k)} \right|}{\max_{i,j} \left| \alpha_{ij} \right|}$$

ϵ is the machine precision

$\kappa_\infty(A)$ is the condition of A

Growth factor analysis[2, 3]

1. Loose bound [3] exists for CALU| and partial pivoting but it is too loose to justify the practical usage of CALU.

2. Average case/probabilistic analysis [2] for GEPP shows sublinear growth factor, but no such study exists for CALU.

3. Challenge Description

Numerical Stability Analysis for CALU is immature Cannot use CALU safely as an alternative to GEPP, i.e. GETRF, in all scenarios.

4. Demand

■ Theoretical Stability Analysis for CALU Prove CALU have sublinear growth factor in most cases and show the relationship with GEPP. Derive growth factor relation with the algorithm, such as param. H and the tournament scheme in [1].

■ Adaptive Algorithm: Provide an algorithm that can adapt between CALU and GEPP , i.e., have stability and comm. cost both between those CALU & GEPP

● **Reference**

[1] Grigori, Laura, James W. Demmel, and Hua Xiang. "Communication avoiding Gaussian elimination."

SC'08: Proceedings of the 2008 ACM/IEEE Conference on Supercomputing. IEEE, 2008.

[2] Trefethen, Lloyd N., and Robert S. Schreiber. "Average case stability of Gaussian elimination." SIAM

Journal on Matrix Analysis and Applications 11.3 (1990): 335 360.

[3] Grigori, Laura, James W. Demmel, and Hua Xiang. "CALU: a communication optimal LU factorization

algorithm." SIAM Journal on Matrix Analysis and Applications 32.4 (2011): 1317 1350.

Acknowledgements