中国科学院计算机网络信息中心
Computer Network Information Center,
Chinese Academy of Sciences

格点QCD的数结构与矢量化方法

中科院计算机网络信息中心
张克龙
klzhang@cnic.cn

2023年7月12日，青海·西宁

格点QCD简介

从第一性原理出发研究QCD的非微扰理论 + 计算方法

欧几里得空间作用量

$$S = \int d^4x \, \mathcal{L}_{QCD}$$

时空离散化

$$S_f[\bar{\psi}, \psi, U] = \sum_x \bar{\psi}\left(\gamma_\mu D_w^\mu[U_\mu(x)] + m_0\right)\psi = \overline{\psi_x} \, M_{xy} \psi_y$$

$$\sum_\mu (1 - \gamma_\mu) U_\mu(x) \delta_{x+\hat{\mu},y} + (1 + \gamma_\mu) U_\mu(x - \hat{\mu})^\dagger \delta_{x-\hat{\mu},y}$$

- 时空离散化、有限体积：有限自由度
- 计算机模拟：马尔科夫链蒙特卡洛重点抽样

LQCD模拟

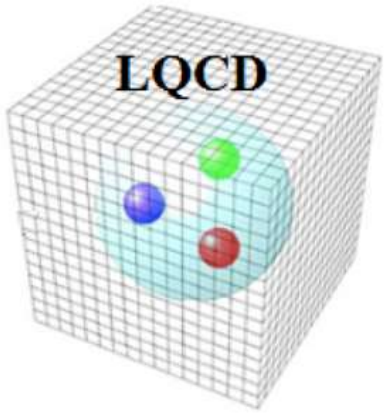
$$Z = \int \mathcal{D}[\bar{\psi}, \psi, U] e^{-S_f - S_g}$$

$$S_f = \overline{\psi_x} \, M_{xy} \psi_y$$

$$\mathcal{P}[U] = \frac{1}{Z} e^{-S_g} * \det M$$

LQCD测量

$$< O_n >= \int \mathcal{D} e^{-S_g} O_n \det M$$



基本的物理量

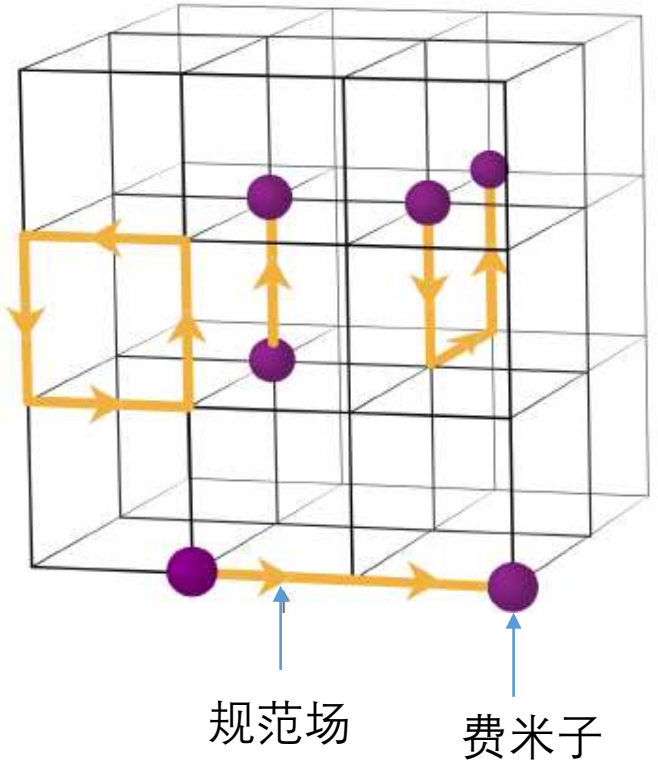
时空离散化

$$S_f[\bar{\psi},\psi,U]=\sum_x \bar{\psi}\left(\gamma_\mu D_w^\mu[U_\mu(x)]+m_0\right)=\overline{\psi_x}\, M_{xy}\psi_y$$

$$\sum_\mu (1-\gamma_\mu)U_\mu(x)\delta_{x+\hat{\mu},y}+(1+\gamma_\mu)U_\mu(x-\hat{\mu})^\dagger\delta_{x-\hat{\mu},y}$$

四维结构化网格

- 格点QCD对称性引进了新的自由度----Spin、Color
- 每个格点上不在是一个浮点数，而张成了3⊗4内部空间复向量、矩阵
- 基本运算 ---- 矩阵（张量）运算



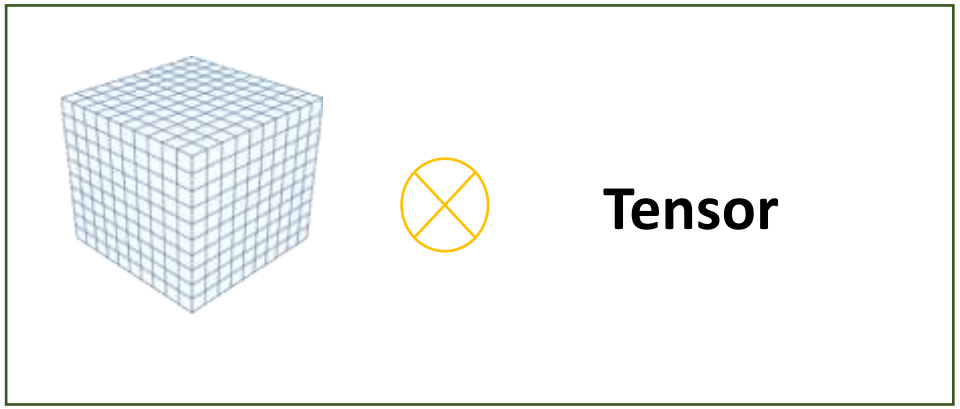
费米子场（Lattice Fermion） 每个网格点上3⊗4，4个3维复向量

规范场（Lattice Gauge）每个网格点上3⊗3⊗4，4个**SU(3)**矩阵

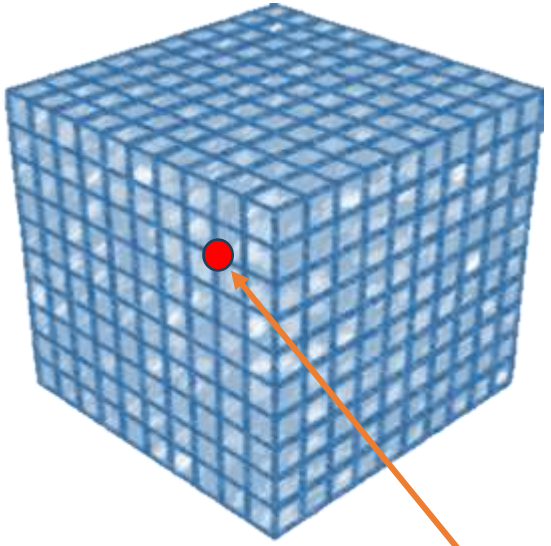
数据类型 Grid<Tensor>

中国科学院计算机网络信息中心
Computer Network Information Center,
Chinese Academy of Sciences

■ 格点QCD中的数据类型　　结构网格 + 张量



Grid<Tensor>



Tensor Point，张量点

费米子场（Lattice Fermion） 每个网格点上3⊗4，4个3维复向量

规范场（Lattice Gauge）每个网格点上3⊗3⊗4，4个**SU(3)**矩阵

实现方式示例

➢ **QDP++ Template Tensor Type Structure**

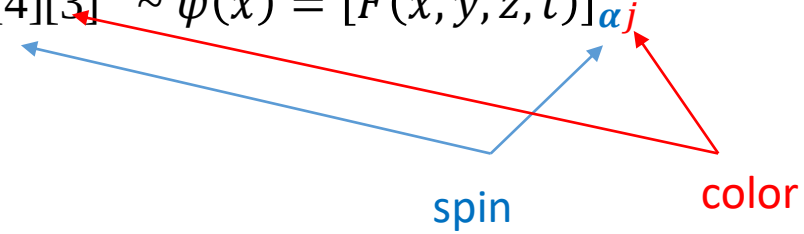
	Lattice	Spin	Colour	Reality	BaseType
Real	Scalar	Scalar	Scalar	Real	REAL
LatticeColorMatrix	Lattice	Scalar	Matrix(Nc,Nc)	Complex	REAL
LatticePropagator	Lattice	Matrix(Ns,Ns)	Matrix(Nc,Nc)	Complex	REAL
LatticeFermionF	Lattice	Vector(Ns)	Vector(Nc)	Complex	REAL32
DComplex	Scalar	Scalar	Scalar	Complex	REAL64

```
typedef OScalar < PScalar    < PScalar<      RScalar <REAL>    >    > > Real;
typedef OLattice< PScalar    < PColorMatrix< RComplex<REAL>, Nc>    > > LatticeColorMatrix;
typedef OLattice< PSpinMatrix< PColorMatrix< RComplex<REAL>, Nc>, Ns> > LatticePropagator;
```

➢ **C/C++ Array**

LatticeColorMatrix std::**complex**<FLOAT> U[T][X][Y][Z][3][3] $\sim U(\tilde{x}) = [U(x,y,x,t)]_{ij}$

LatticeFermion std::**complex**<FLOAT> U[T][X][Y][Z][4][3] $\sim \psi(\tilde{x}) = [F(x,y,z,t)]_{\alpha j}$

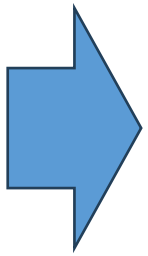


共性问题归纳

- 大规模、同构、小型**矩阵（张量）**的运算
- 结构网格

特点：

- **小型**：每个网格点上的数据结构
  一个小型矩阵或向量（张量）
- **同构**：每个网格点的数据结构和运算
  一致的；
- **大规模**：遍历所有网格点



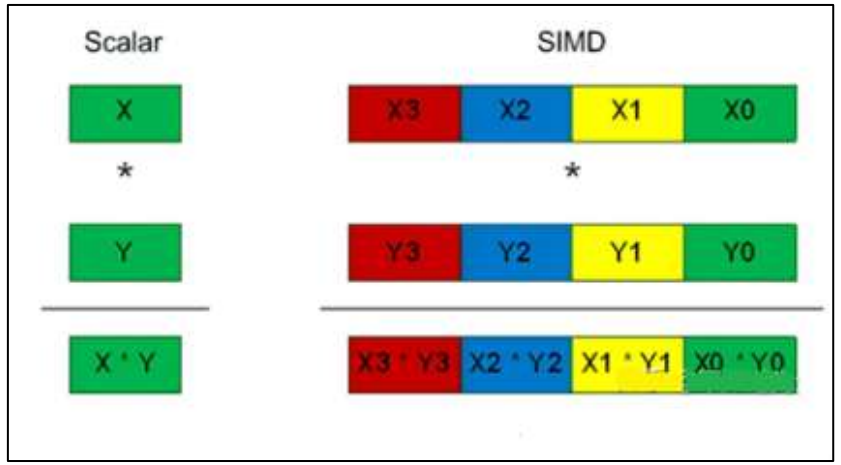
格点QCD

LatticeGauge、LatticeFermion

- SU(3)矩阵、向量
- 矩阵向量乘（Gemv）
- 网格 32x32x32x64 ~ 200万网格点

- 通常，数学库对小矩阵优化不好（C++ std::complex<Tp> 实现的Gemv 性能优于 cblas库zgemv()
- 矢量加速单元位宽越来越长，小矩阵不能充分其发挥
- 复数的向量化较复杂，且效果不好

SIMD



矢量加速单元

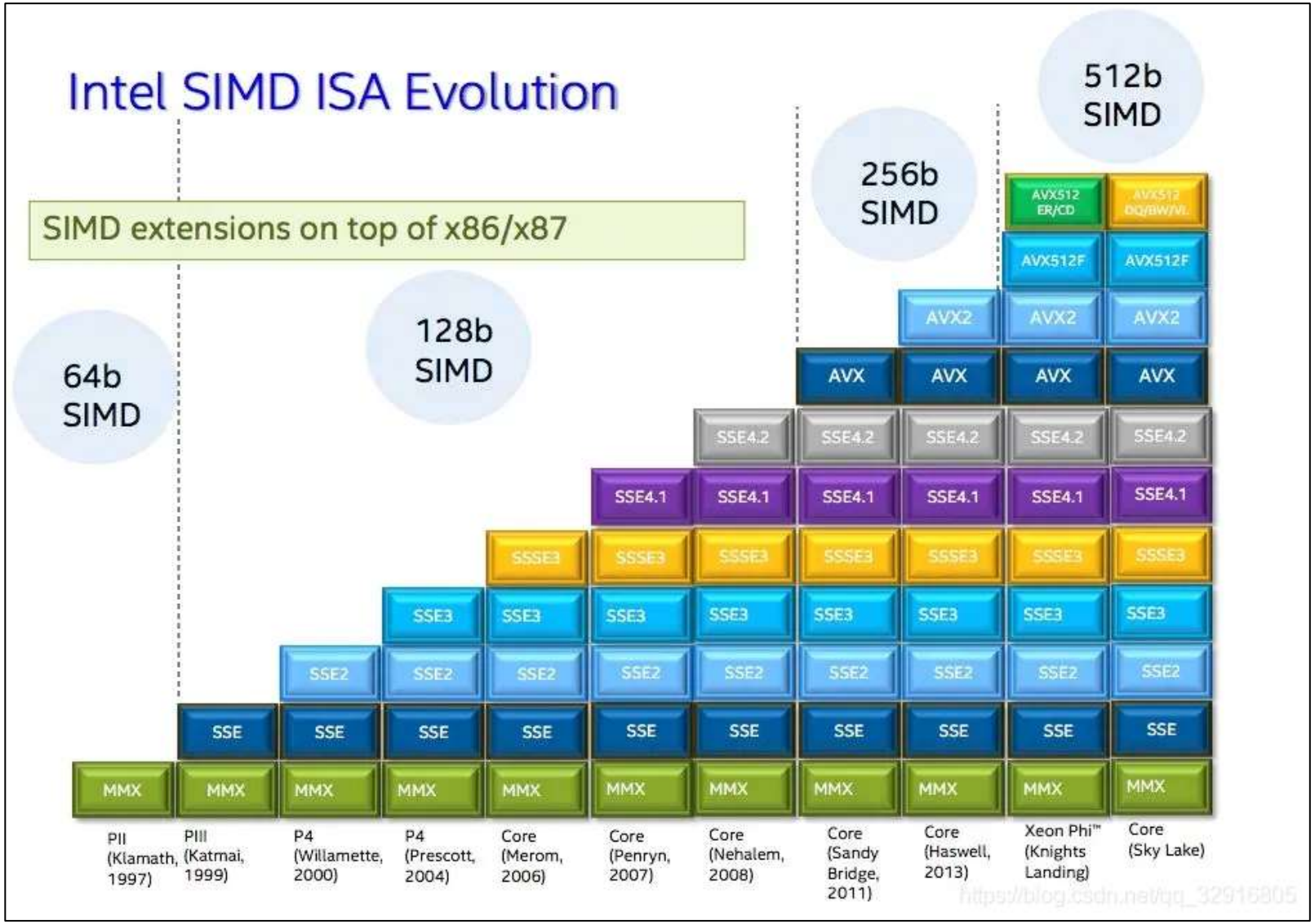
同构:

x86: 128bits、256bit、512bits

Arm: neon、SVE

异构:

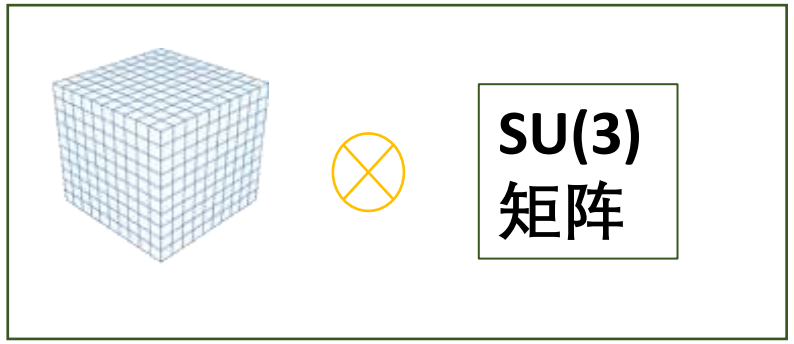
神威、天河....

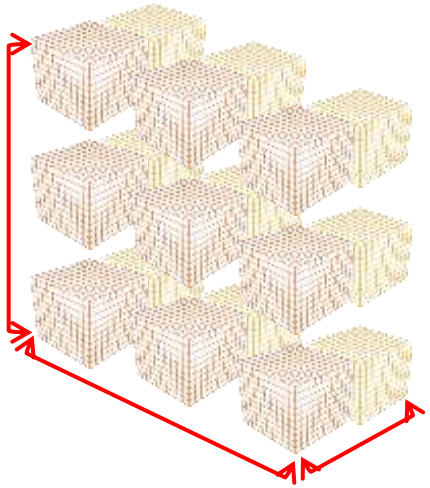


Tensor<Grid> & DLP

中国科学院计算机网络信息中心
Computer Network Information Center,
Chinese Academy of Sciences

DLP （Data Level Parallelism）





TensorGrid 方案：

- 将tensor指标提取分离到时空
- 数据并行度为 Grid，并且内存连续
- 向量化实现更容易

Grid<Tensor>

方法 Array of Struct (AOS)

```
LatticeFermion   std::complex<double> F[GridSize][4][3]
LatticeFermion   std::complex<double> F[GridSize][4][3]
```

SIMD

- 矢量长度受限网格点上Tensor结构，如4、3
- 复数矢量化涉及位变换
- 复数类的多精度计算性能受限

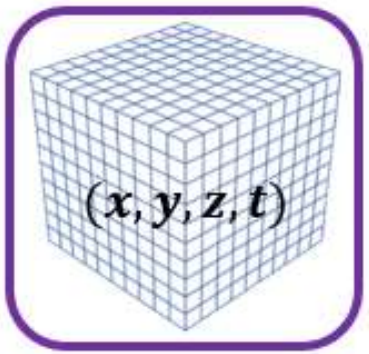
Tensor<Grid>

Struct of Array (SOA)

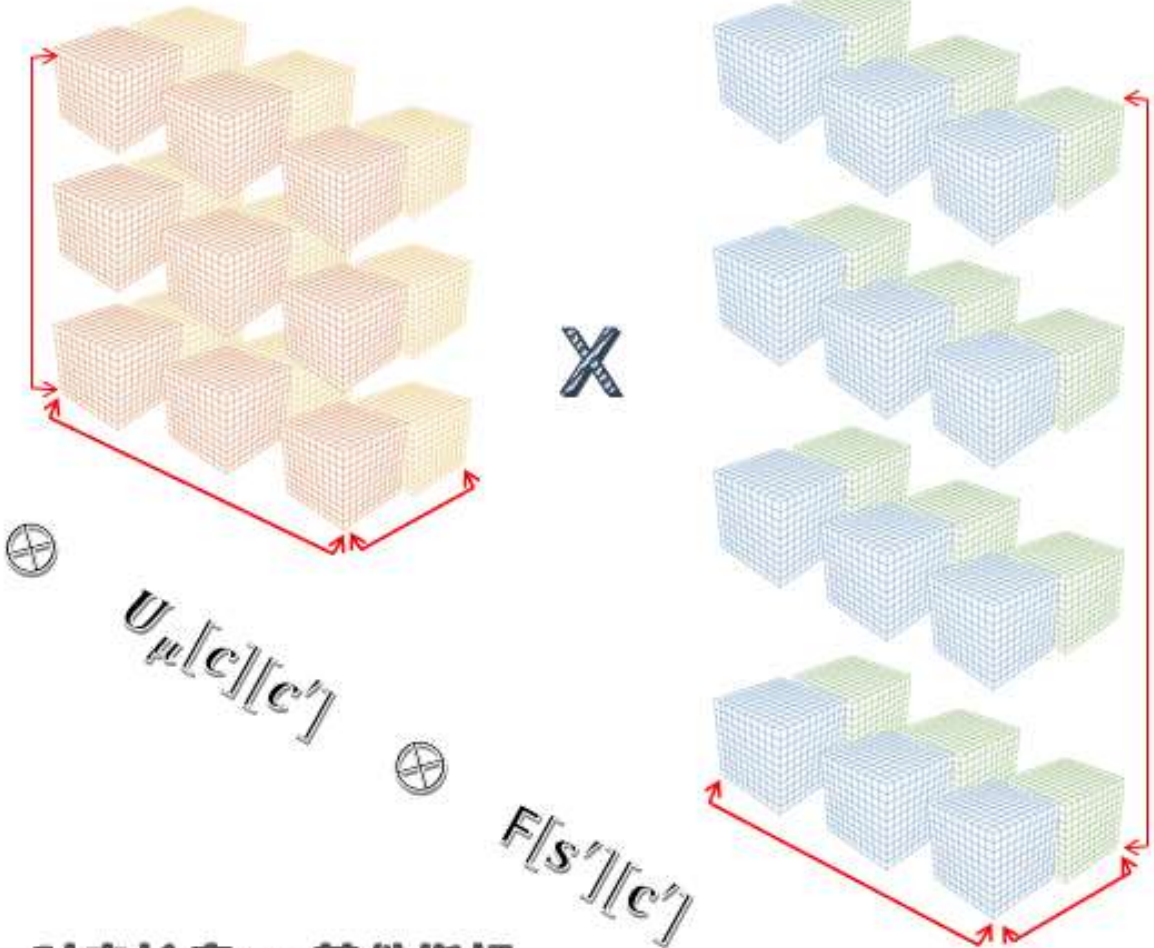
```
double F[4][3][2] [GridSize]
double F[4][3][2] [GridSize]
```

- 矢量长度位网格格点数，GridSize
- 计算转化为连续实数简单运算
- 多精度计算性能较好

$$[F_{sc}] = \Gamma_{ss'} \sum_{c'=012} [U_{cc'}][F_{sc'}]$$



$[(1-\gamma)]$ ⊗ $(1-\gamma_\mu)$ ⊗ $U_\mu[c][c']$ ⊗ $F[s'][c']$



基本数据单元

- 将各内部指标（包括复数）提取分离到时空外
- 每个格点位置一个数
- 以此来构建Gauge、Fermion
- 并行度为 Lattice Size，并且内存连续

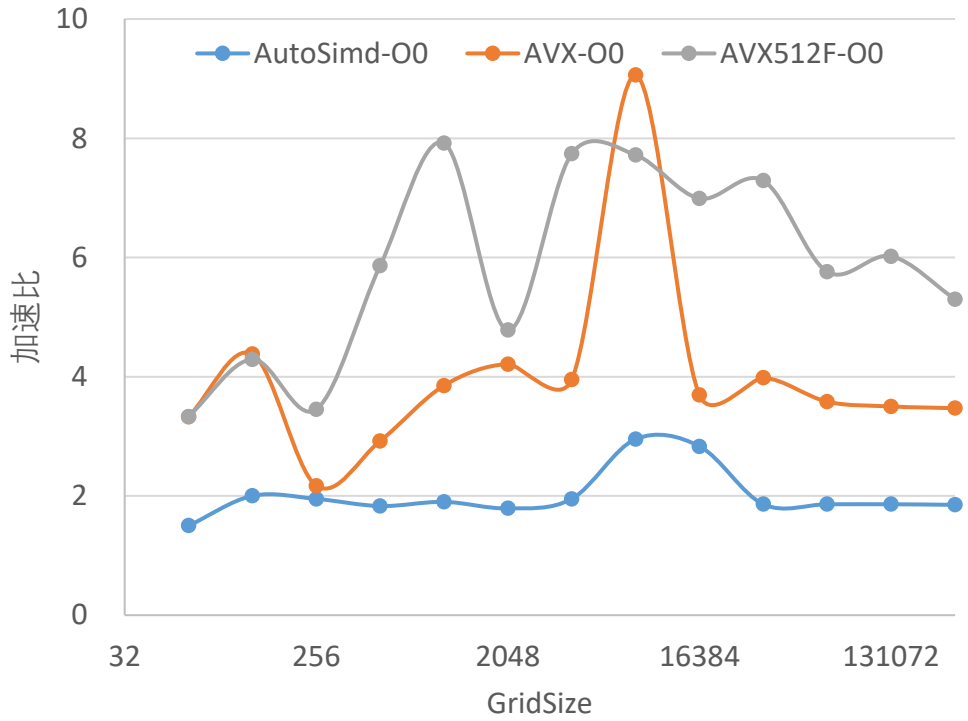
时空长度 >> 其他指标

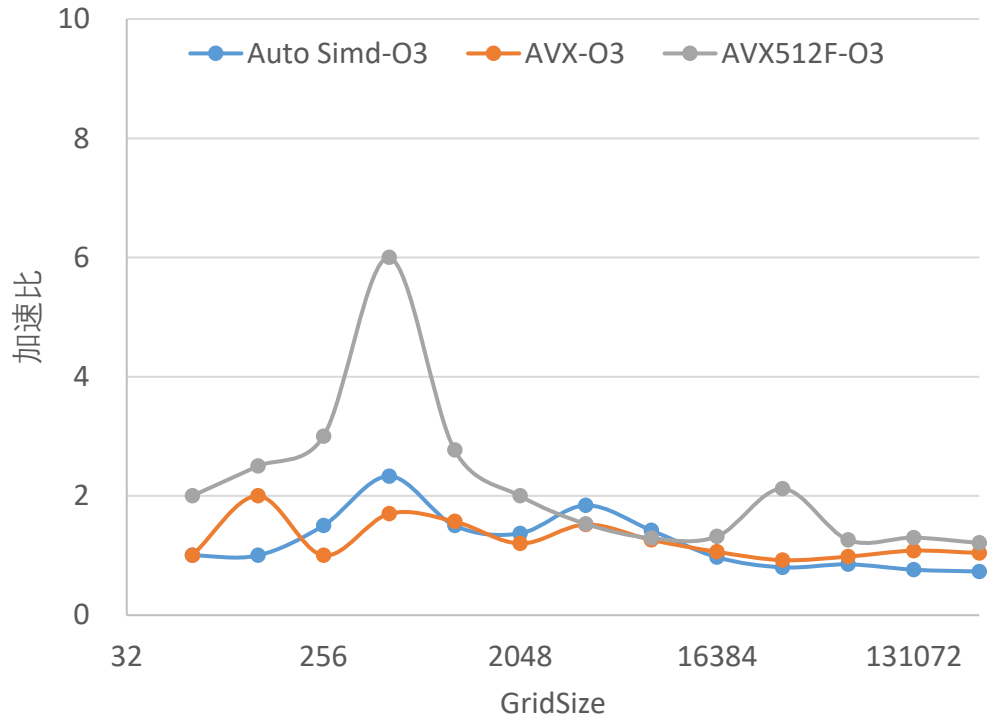
Tensor<Grid> SU(3) Gemv，Double Prec

中国科学院计算机网络信息中心
Computer Network Information Center,
Chinese Academy of Sciences

$$[V(x)]_i = [U(x)]_{ij}\,[V(x)]_j$$

i,j 为 color 空间指标



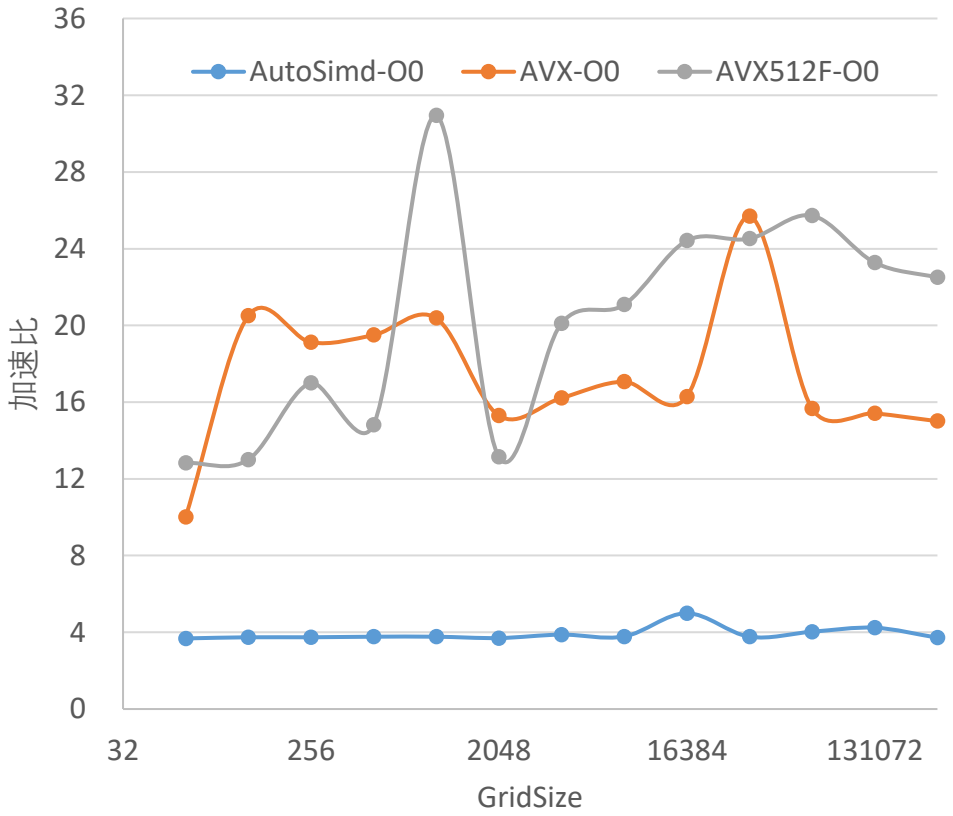


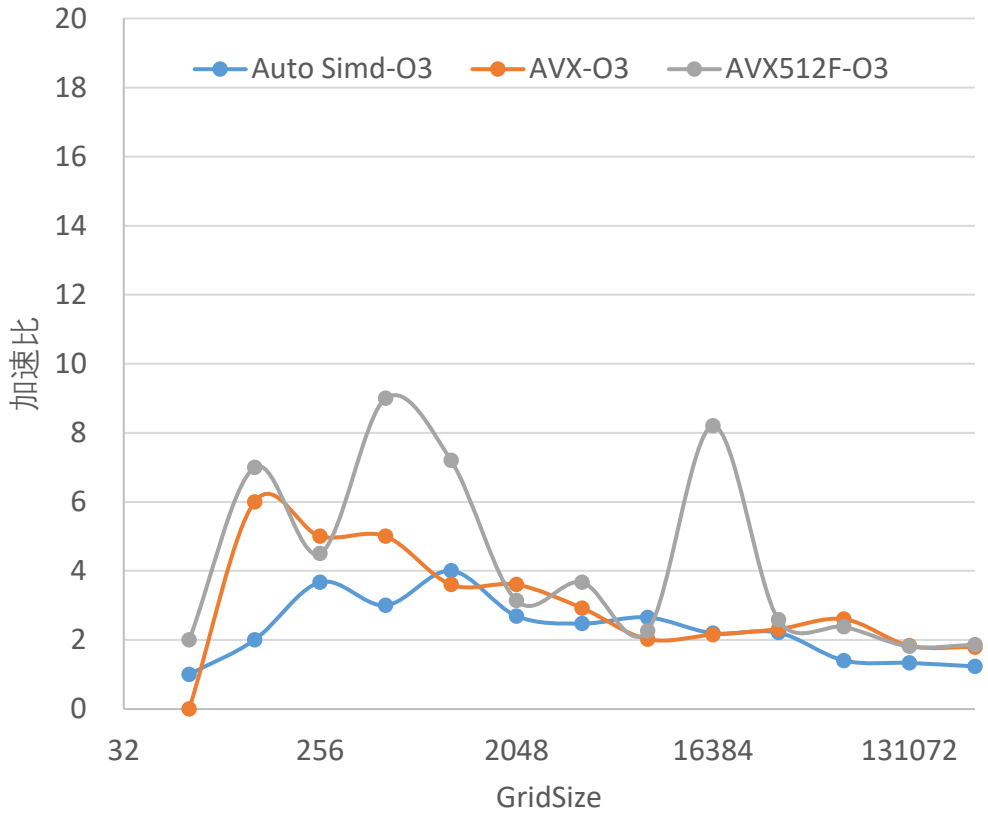
Tensor<Grid> 相对于 Grid<Tensor>的双精度SU(3) Gemv 加速比

Tensor<Grid> SU(3) Gemv，Single Prec

中国科学院计算机网络信息中心
Computer Network Information Center,
Chinese Academy of Sciences

$$[V(x)]_i = [U(x)]_{ij}\,[V(x)]_j$$ ，*i,j 为 color 空间指标*





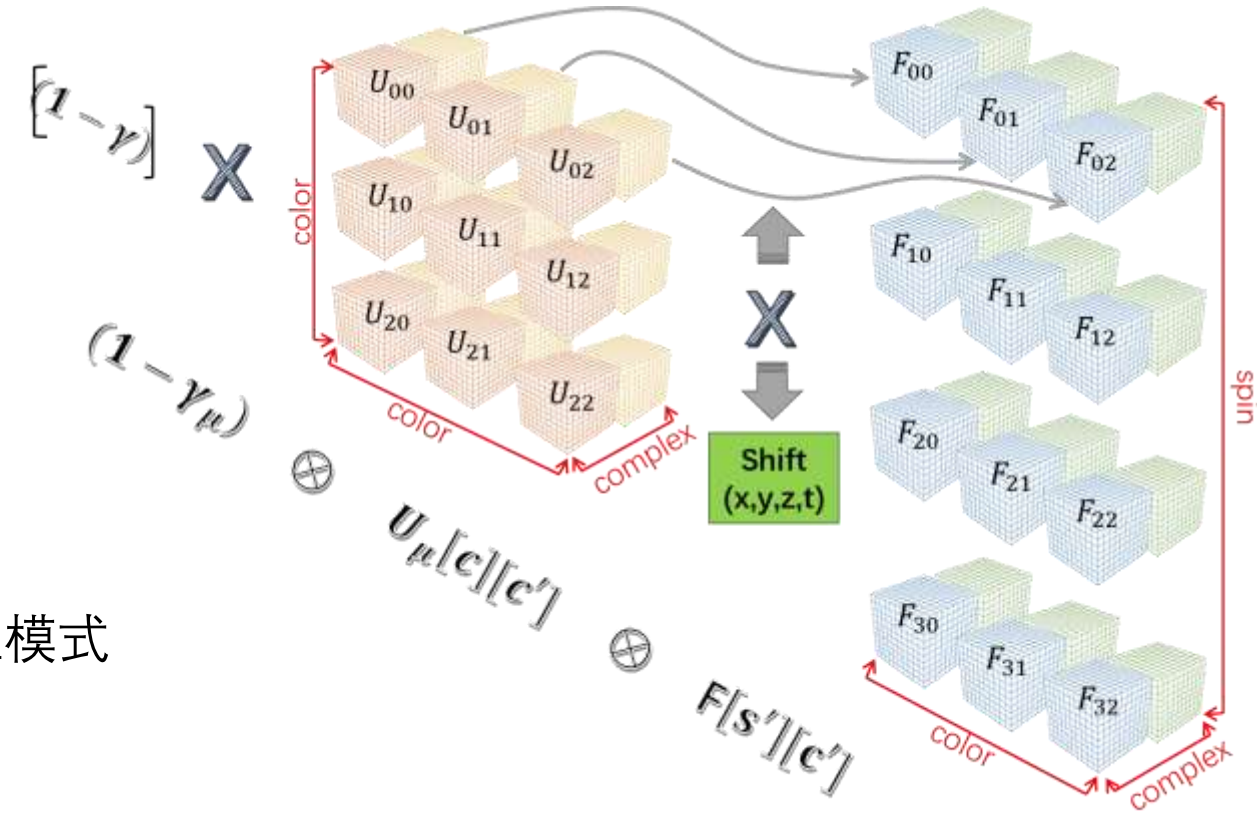
Tensor<Grid> 相对于 Grid<Tensor>的双精度SU(3) Gemv 加速比

总结与展望

中国科学院计算机网络信息中心
Computer Network Information Center,
Chinese Academy of Sciences

总结

- 格点QCD计算中规范场、费米子场数据类型
- 大规模、同构、小规模矩阵运算的共性问题
- SU(3)空间的Tensor<Grid>数据结构及矢量化方案



展望

- 面向数据（DOP）结合 面向对象（OOP）的编程模式
- 基于Tensor<Grid> 数据结构，构建格点QCD算符
- 关于结构网格并行计算与通信的处理
- 共性算法：面向结构化网格；同构、小型的张量运算

致谢

请各位专家批评指正！

谢谢！

表1. Gemv算子在O0编译选项下不同矢量指令的双精度加速比

GridSize	clang auto-O0	AutoSimd-O0	AVX-O0	AVX512F-O0	clang 512-O0	1
64	1.81	1.50	3.33	3.33	2.73	2.83
128	2.00	2.00	4.38	4.29	3.33	3.36
256	1.90	1.95	2.17	3.45	4.76	4.00
512	1.91	1.83	2.92	5.86	5.92	4.76
1024	1.90	1.90	3.85	7.92	6.76	5.66
2048	1.82	1.79	4.21	4.78	6.49	6.73
4096	1.90	1.95	3.95	7.74	7.40	8.00
8192	1.91	2.95	9.06	7.72	9.04	9.51
16384	2.38	2.83	3.69	6.99	8.12	11.31
32768	2.19	1.86	3.98	7.29	5.77	13.45
65536	1.92	1.86	3.58	5.76	4.54	16.00
131072	1.87	1.86	3.50	6.02	4.66	19.03
262144	1.87	1.85	3.47	5.30	4.42	22.63

表2. Gemv算子在O3编译选项下不同矢量指令的双精度加速比

GridSize	clang auto-O3	Auto Simd-O3	AVX-O3	AVX512F-O3	clang 512-O3	SVE-O3
64	1	1.00	1.00	2.00	1.00	
128	1.5	1.00	2.00	2.50	2.50	
256	1.5	1.50	1.00	3.00	3.33	
512	1.64	2.33	1.70	6.00	3.80	
1024	1.44	1.50	1.57	2.77	5.45	
2048	1.35	1.37	1.20	2.00	2.30	
4096	1.86	1.84	1.52	1.53	2.29	
8192	1.76	1.42	1.26	1.29	1.49	
16384	1.61	0.97	1.06	1.32	1.42	
32768	1.27	0.80	0.92	2.12	2.66	
65536	0.87	0.85	0.98	1.26	1.66	
131072	0.79	0.76	1.08	1.30	1.39	
262144	0.71	0.73	1.04	1.21	1.28	

表3. Gemv算子在O0编译选项下不同矢量指令的单精度加速比

GridSize	clang auto-O0	AutoSimd-O0	AVX-O0	AVX512F-O0	clang 512-O0	SVE-O0
64	3.73	3.67	10.00	12.83	9.50	
128	3.62	3.73	20.50	13.00	9.67	
256	4.04	3.73	19.12	17.00	12.50	
512	3.96	3.76	19.50	14.82	17.96	
1024	3.88	3.76	20.38	30.95	21.95	
2048	3.91	3.69	15.30	13.14	23.17	
4096	3.89	3.87	16.22	20.11	25.92	
8192	4.82	3.77	17.07	21.08	22.98	
16384	4.11	4.99	16.28	24.42	20.51	
32768	5.43	3.76	25.69	24.52	17.44	
65536	4.04	4.02	15.67	25.73	18.88	
131072	3.80	4.23	15.42	23.28	18.05	
262144	3.84	3.71	15.01	22.50	17.45	

表4. Gemv算子在O0编译选项下不同矢量指令的单精度加速比

GridSize	clang auto-O3	Auto Simd-O3	AVX-O3	AVX512F-O3	clang 512-O3	SVE-O3
64	1.50	1.00	inf	2.00	3.00	
128	1.00	2.00	6.00	7.00	2.50	
256	2.20	3.67	5.00	4.50	4.50	
512	3.00	3.00	5.00	9.00	6.33	
1024	3.42	4.00	3.60	7.20	7.20	
2048	2.56	2.69	3.60	3.14	3.83	
4096	2.46	2.47	2.92	3.67	3.54	
8192	3.85	2.65	2.02	2.26	4.82	
16384	3.29	2.19	2.15	8.20	3.08	
32768	4.06	2.21	2.31	2.58	3.19	
65536	1.59	1.40	2.60	2.38	2.93	
131072	1.21	1.33	1.83	1.81	2.27	
262144	1.22	1.23	1.79	1.86	2.10	

automatically perform four, eight or sixteen matrix vector products in parallel. There is no need to horizontally add across SIMD lanes, and no impact of pipeline latency beyond any seen by scalar code. The vector code is essentially identical to scalar code, except that *sizeof(double)* appears to have grown to 128/256/512 bits. As a result the routine is automatically 100% SIMD efficient.

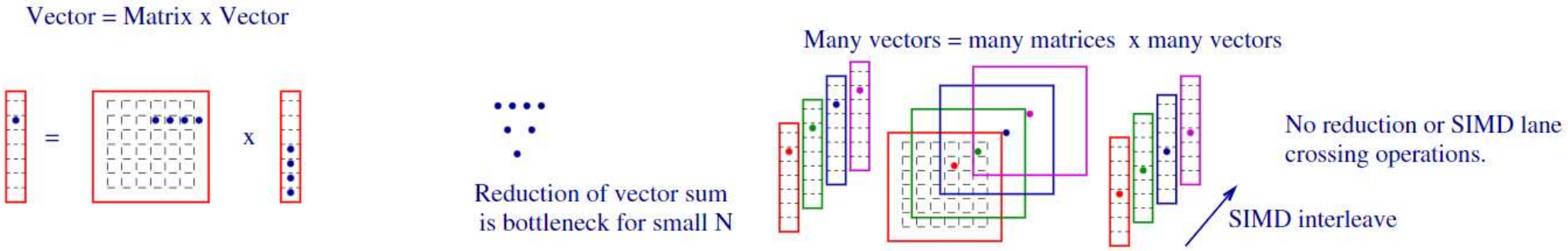


Figure 1: Two different approaches to SIMD accelerating matrix-vector products. The left approach suffers from long latency operations when horizonally adding a summation register (in addition to the fact that increasing vector lengths do not typically neatly divide the index ranks of QCD fields), while the right hand approach to performing many of these operations does not.

QCD Wide SIMD Library (QWS)

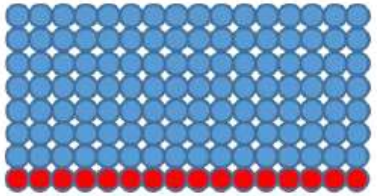
- **Lattice QCD simulation library for Fugaku and computers with wide SIMD, in C/C++**
- **Development with Fujitsu**
  - Has been started by Y. Nakamura since 2014
  - Y. Mukai (Fujitsu) joined since 2015
  - K.-I. Ishikawa (Hiroshima) joined since 2015
  - I. Kanamori (Hiroshima -> RIKEN) joined since 2018
- **Download : https://github.com/RIKEN-LQCD/qws**
- **High performance on Fugaku for Wilson-clover**
  - **SIMD vectorization for 512 bits SIMD**
    - FMA computational latency (Real : 9, Complex : 15 or 16)
    - It is easier to achieve performance with a data layout that separates real and imaginary parts (**RRII layout**)

    Re Re Re Re Re Re Re Re 512 bits SIMD register
    Im Im Im Im Im Im Im Im 512 bits SIMD register

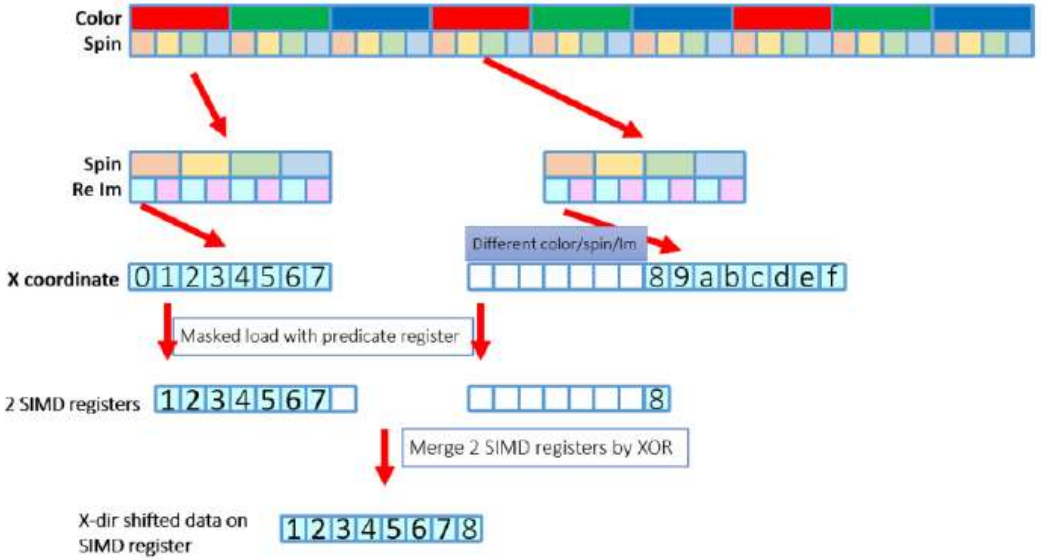
  - Removing temporal arrays, unnecessary copies
  - Manual prefetch for all arrays
  - OMP Parallel region expansion
  - Mixed precision Krylov subspace method
  - Minimizing communication overhead by process mapping and double buffering

Data layout example for Fugaku

Continuous access and 100% SIMD vectorization rate, except for X-direction difference calculation
Fugaku(FP64):[nt][nz][ny][nx/8][3][4][2][8]
Fugaku(FP32):[nt][nz][ny][nx/16][3][4][2][16]
Fugaku(FP16):[nt][nz][ny][nx/32][3][4][2][32]



X-direction shift with Arm C Language Extensions (ACLE)



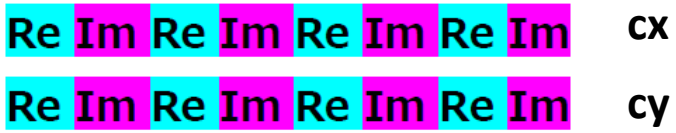
复向量乘 cx[i] *= cy[i]

① 复数数组

complex [N]

② **double 数组**

double [N][2]

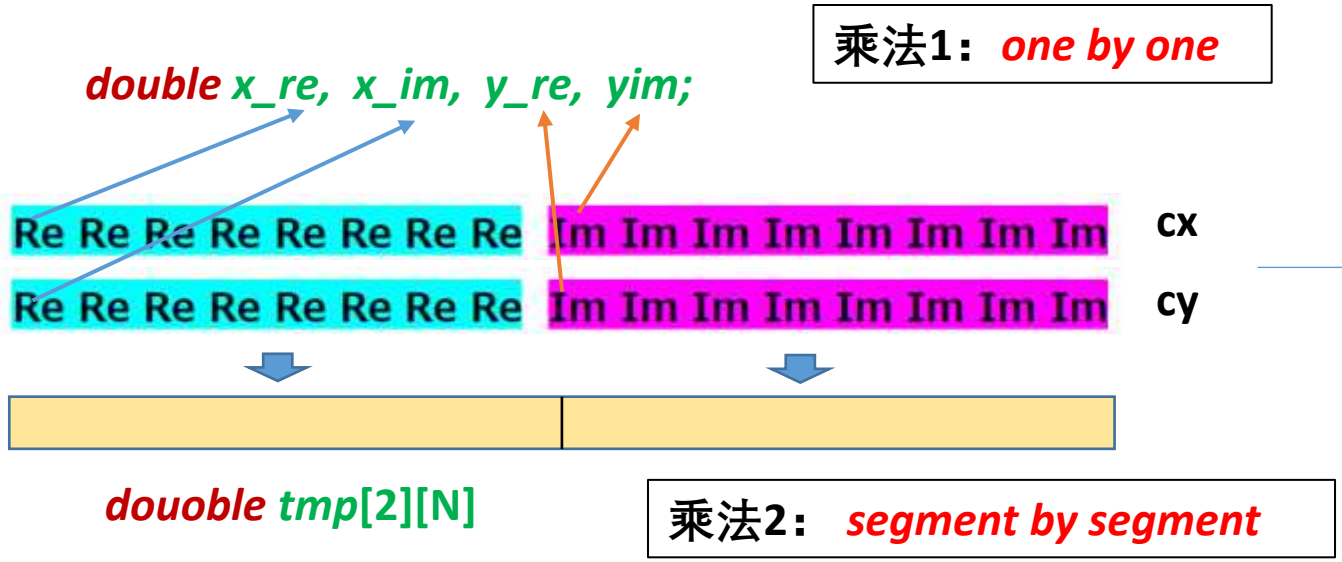


③ 实部虚部分离，乘法1

double [2][N]

④ 实部虚部分离，乘法2

double [2][N]



格点QCD的PDE与泊松方程

中国科学院计算机网络信息中心
Computer Network Information Center,
Chinese Academy of Sciences

格点的PDE与泊松方程

泊松方程 $-\Delta S(x) = f$

Wilson-Dirac方程 $\gamma D^{(U(x))} S(x) = f$

差分形式：

$$2S(x) - [\quad S(x+1) + \quad S(x-1)] = f$$

$$2S(x) - [\Gamma_{s's} U_{c'c}(x) S_{sc}(x+1) + \Gamma'_{s's} U^{+}_{c'c}(x-1) S_{sc}(x-1)\,] = f_{s'c'}$$

Stencil计算
矢量化SIMD

Stencil计算
线程级并行STMD
矢量化困难

格点QCD规范协变性引进了新的自由度----Spin、Color

每个格点上不在是一个浮点数，而张成了3⊗4内部空间复向量（矩阵）

Stencil计算 浮点数运算 ---- 矩阵（张量）运算

稀疏矩阵向量乘更加复杂