

自研多重网格算法小组

2025/07/14-2025/07/26

工作汇报

兰州大学

核科学与技术学院

2021级核物二班

张鑫

代码托管:

<https://github.com/zhangxin8069/qcu.git>; <https://gitee.com/zhangxin8069/qcu.git>;
<https://github.com/zhangxin8069/PyQCU.git>; <https://gitee.com/zhangxin8069/PyQCU.git>;

代码运行环境:

SNSC;COMPUTER-4060;X99;

电话:













15572348069;

邮箱:

zhangxin2021@lzu.edu.cn;

zhangxin9542@gmail.com;

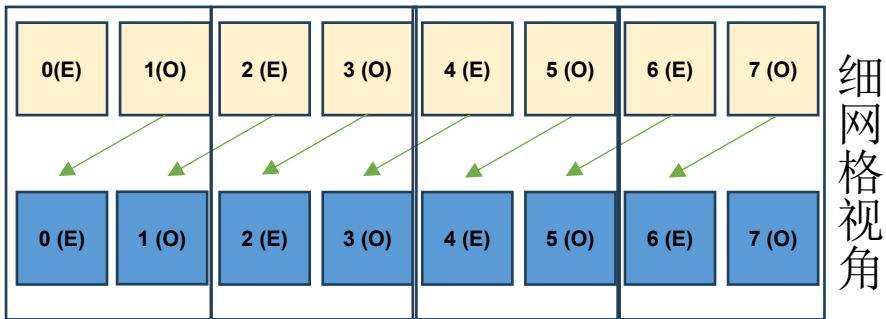
zhangxin8069@qq.com;

标签名	描述	提交信息
 bug20	follow dev51, remember that ###Schmitt orthogonalization of complex space###:"null_vecs[i] -= torch.vdot(null_vecs[j].flatten(), null_vecs[i].flatten())/torch.vdot(null_vecs[j].flatten(), null_vecs[j].flatten())*null_vecs[j]", not "null_vecs[i] -=	0dfdf7e  2025-07-25 22:50
 dev51	follow bug19, 1. rename some var in PyQCU/pyqcu/ascend/dslash_parity.py:eye -> I; 2. referring to PyQCU/test/test-dev51-ASCEND-*, PyQCU/pyqcu/ascend/inverse.py:cg & bicgstab & give_null_vecs done! ^_^	7f18ca4  2025-07-25 16:16
 bug19	follow stab18, remember that ###vdot(,)=dot(conj(),)###!!!,do not use vdot as same as dot!!!, debug PyQCU/pyqcu/ascend/inverse.py and some PyQCU/test/test-dev50-ASCEND-WILSON-* done. www~	d597c71  2025-07-24 23:20
 stab18	follow dev50, referring to PyQCU/extern/cuda/qcu/include/lattice_* and PyQCU/extern/cuda/qcu/src/apply_*, add "VERBOSE" to the params of qcu(cuda), some test files of test5 have been added and optimized. ^_^	7b22552  2025-07-18 20:42
 dev50	follow bug18, referring to PyQCU/test/test-dev50-ASCEND-* and PyQCU/pyqcu/ascend/dslash_parity.py, the bug mentioned earlier occurred due to the lack of details in parity handling: 1. The eye item should not appear in wilson_eo and wilson_oe, it should be	4ce92f2  2025-07-17 21:23
 bug18	follow dev49, there is a bug in PyQCU/pyqcu/ascend/dslash_parity.py(refer to PyQCU/test/test-dev50-ASCEND-WILSON-PARITY.py and /PyQCU/test/test-bug18-ASCEND-WILSON-PARITY.log).....	a12c8e4  2025-07-15 17:40

工作一：引入奇偶分离处理，回顾如下：

约定：
实际使用时为了粗化过程更加符合Dslash的数学特性
(实际引入了GMG的研究方法)，将CLOVER项与
HOPPING项分开处理。

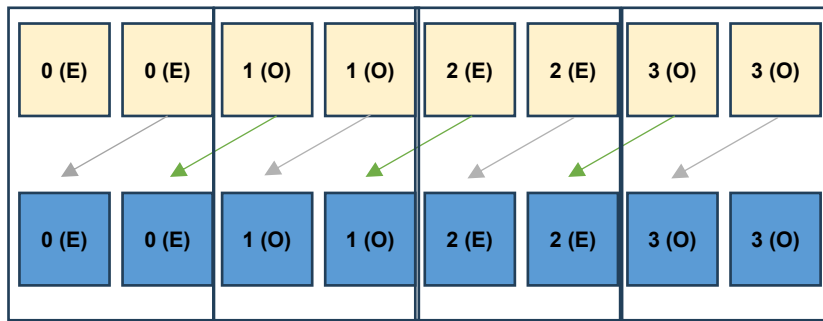
外禀非对角（称为Hopping项），以+1项为例，-1项同理



细
网
格
视
角



增加了外禀对角（称为Clover项），灰色标记的箭头



粗
网
格
视
角

为了使粗化过程中细Hopping项产生的粗Hopping项与粗Clover项分离，在此引入奇偶分离处理。



$D_c \cdot src_c = R \cdot D_f \cdot P \cdot src_c = dest_c$

反复代入内禀单位的src_c,

即for (xyzt) :

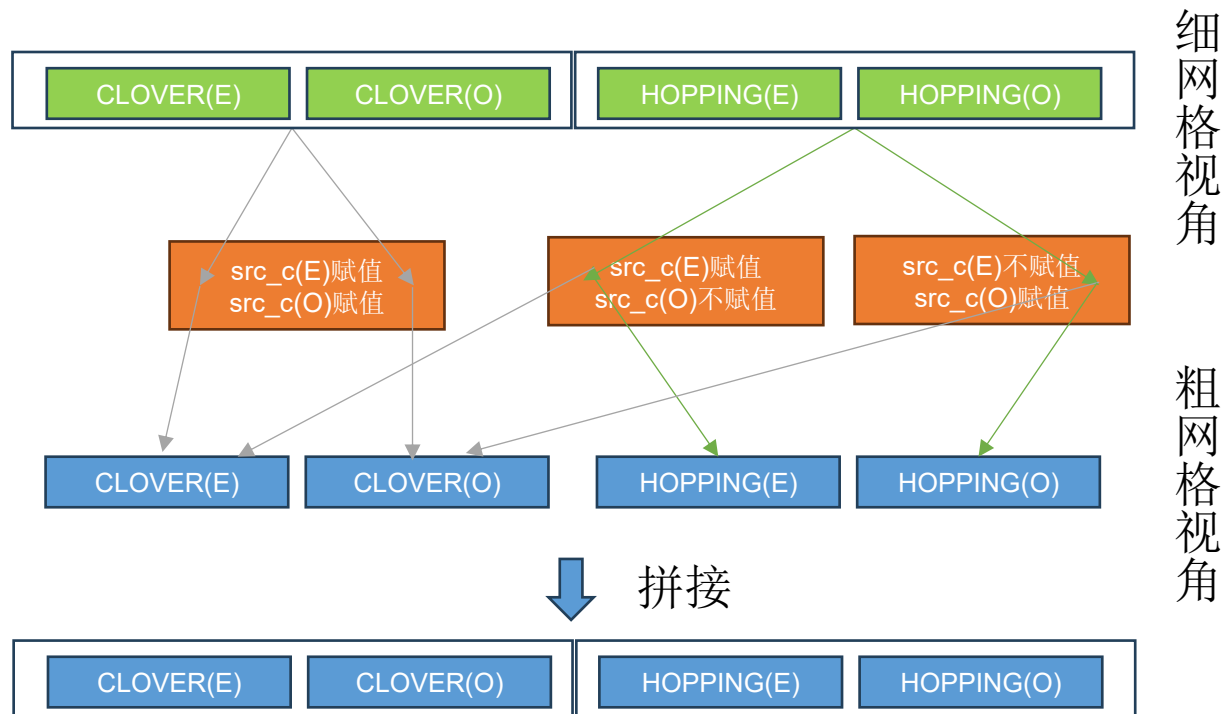
[1,0,0,0,0,0,0,0,0,0,0,0],

[0,1,0,0,0,0,0,0,0,0,0,0],

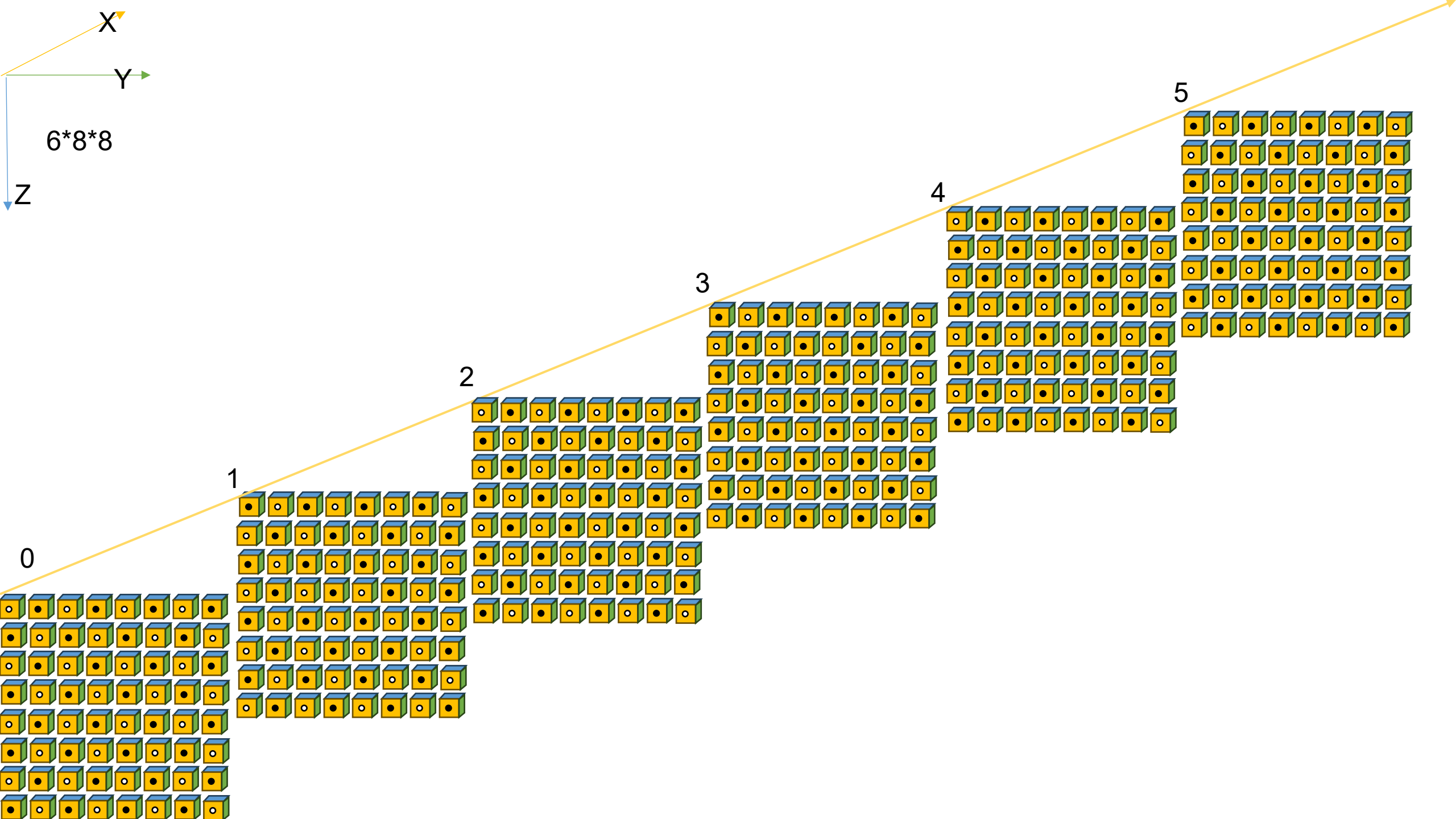
[0,0,1,0,0,0,0,0,0,0,0,0],

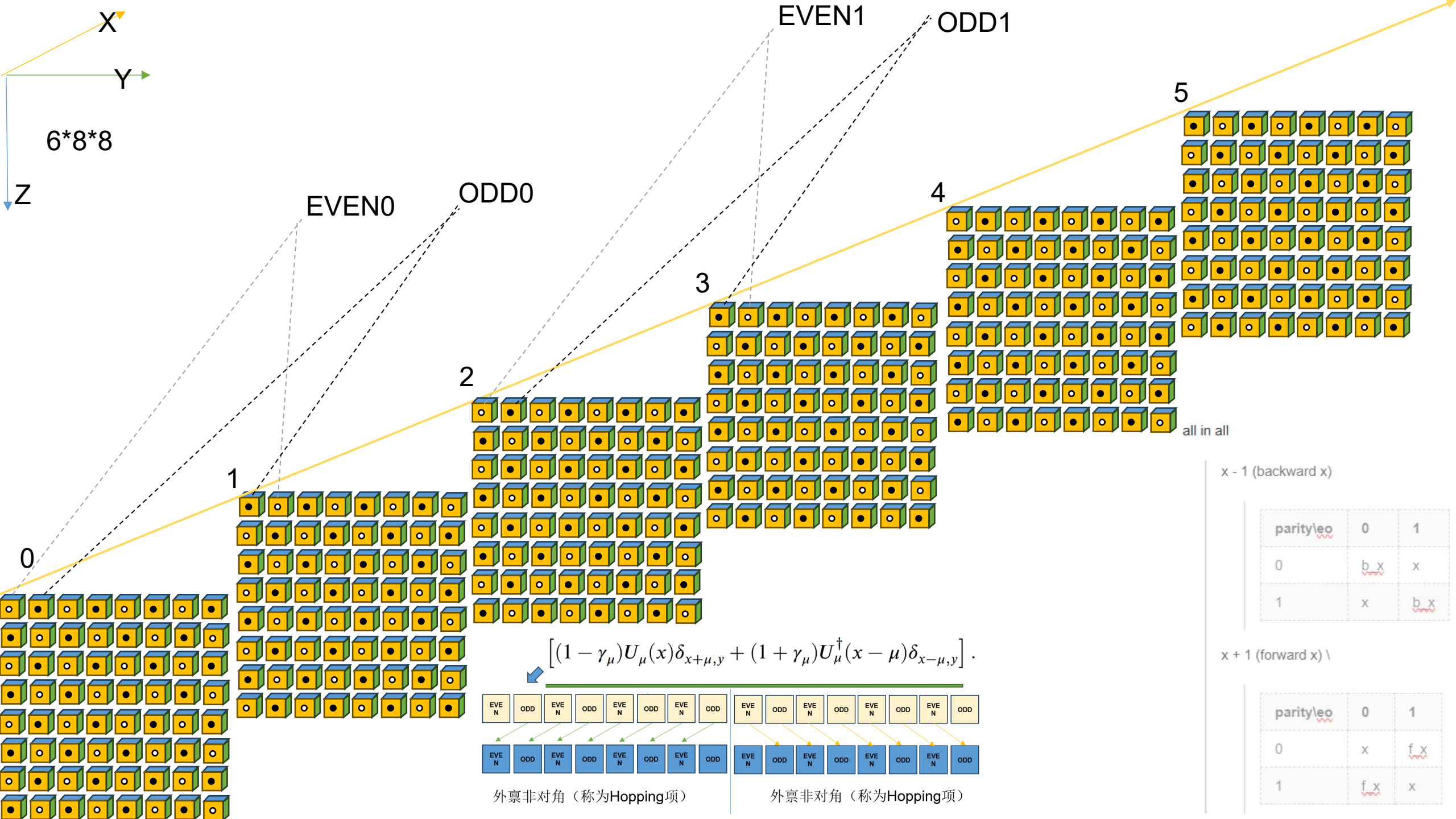
...

来得到数值的显式的D_c (SCSCTZYX)



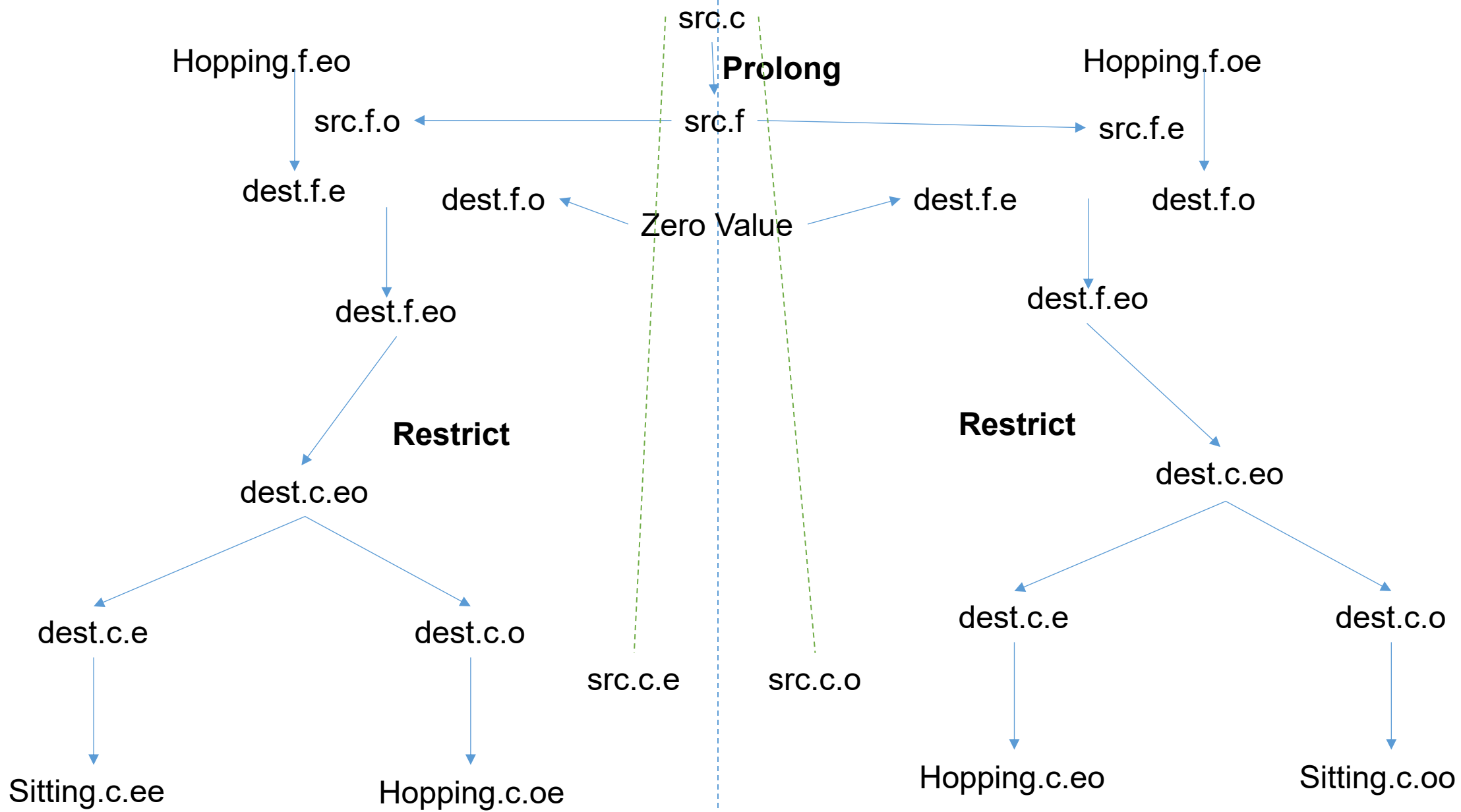
工作一：引入奇偶分离处理，思路如下：





工作一：引入奇偶分离处理，结果参考：
bug18;dev50;

工作二：完成近核向量（零空间向量）构建，
回顾参考“工作一：回顾”，思路如下：



即寻找此Python实例没有加速比的原因



工作三： **DEBUG**以及参考wilson-mg（yht python版），
详情如下：

PyQCU/test/mg_example/bicgstab.py

📄 bicgstab.py 2.60 KB

ZhangXin8069 提交于 21天前 . 2025-07-04-09-31-44

```
44      # 计算步长 alpha
45      alpha = cp.vdot(cp.conj(r0), r) / cp.vdot(cp.conj(r0), Ap)
46      # print("alpha = ", alpha)
47      x += alpha * p
```

vdot(,)=dot(conj(),)

设 $\{v_1, v_2, \dots, v_n\}$ 是复数向量组，目标是构造正交组 $\{u_1, u_2, \dots, u_n\}$ ，使得：

$$u_k = v_k - \sum_{j=1}^{k-1} \text{proj}_{u_j}(v_k), \quad \text{其中 } \text{proj}_{u_j}(v_k) = \frac{\langle u_j, v_k \rangle}{\langle u_j, u_j \rangle} u_j$$

其中内积定义为（复共轭）：

$$\langle x, y \rangle = x^\dagger y = \sum_i \bar{x}_i y_i$$

PyQCU/test/mg_example/mg.py

📄 mg.py 24.26 KB

一键复制

编辑

原始数据

按行查看

历史

ZhangXin8069 提交于 21天前 . 2025-07-04-09-31-44

```
14      # 生成近零空间向量
15      def near_null_vec(self, P_null_vec_coarse, coarse_dof, coarse_op):
16          for i in range(0, coarse_dof):
17              # 施密特正交化
18              for k in range(0, i):
19                  P_null_vec_coarse[i, :, :, :] -= cp.vdot(cp.conj(P_null_vec_coarse[i, :, :, :]), P_null_vec_coarse[k, :, :, :])/cp.vdot(
20                      cp.conj(P_null_vec_coarse[k, :, :, :]), P_null_vec_coarse[k, :, :, :])*P_null_vec_coarse[k, :, :, :]
```

工作三：DEBUG以及参考wilson-mg（yht python版），
debug后完成了Null Space Vector的局域正交化，
结果参考：
bug20；

补充问题:

PyQCU/pyqcu/ascend/inverse.py

inverse.py 26.48 KB

一键复制

编辑

原始数据

按行查看

历史

ZhangXin8069 提交于 32分钟前 . 2025-07-25-14-50-54

```
149     Orthonormal near-null space vectors
150     """
151     dof = null_vecs.shape[0] # Number of null space vectors
152     null_vecs = torch.rand_like(null_vecs)
153     for i in range(dof):
154         # The orthogonalization of r
155         if normalize:
156             null_vecs[i] /= torch.norm(null_vecs[i]).item()
157         for j in range(0, i):
158             null_vecs[i] -= torch.vdot(null_vecs[j].flatten(), null_vecs[i].flatten())/torch.vdot(
159                 null_vecs[j].flatten(), null_vecs[j].flatten())*null_vecs[j]
160         if normalize:
161             null_vecs[i] /= torch.norm(null_vecs[i]).item()
162         #  $v=r-A^{-1}Ar$ 
163         null_vecs[i] -= bicgstab(b=matvec(null_vecs[i]), matvec=matvec, tol=tol*1000, max_iter=max_iter, x0=torch.zeros_like(null_vecs[i]),
164                                 verbose=verbose) # tol needs to be bigger...
```

当使用此算法求解Null Space Vector时，tol的选择？

太小时null_vec趋近于零向量，
太大时于随机向量无异（ $Av/v \sim 0.1$ ，显然不近零）

Just pass.

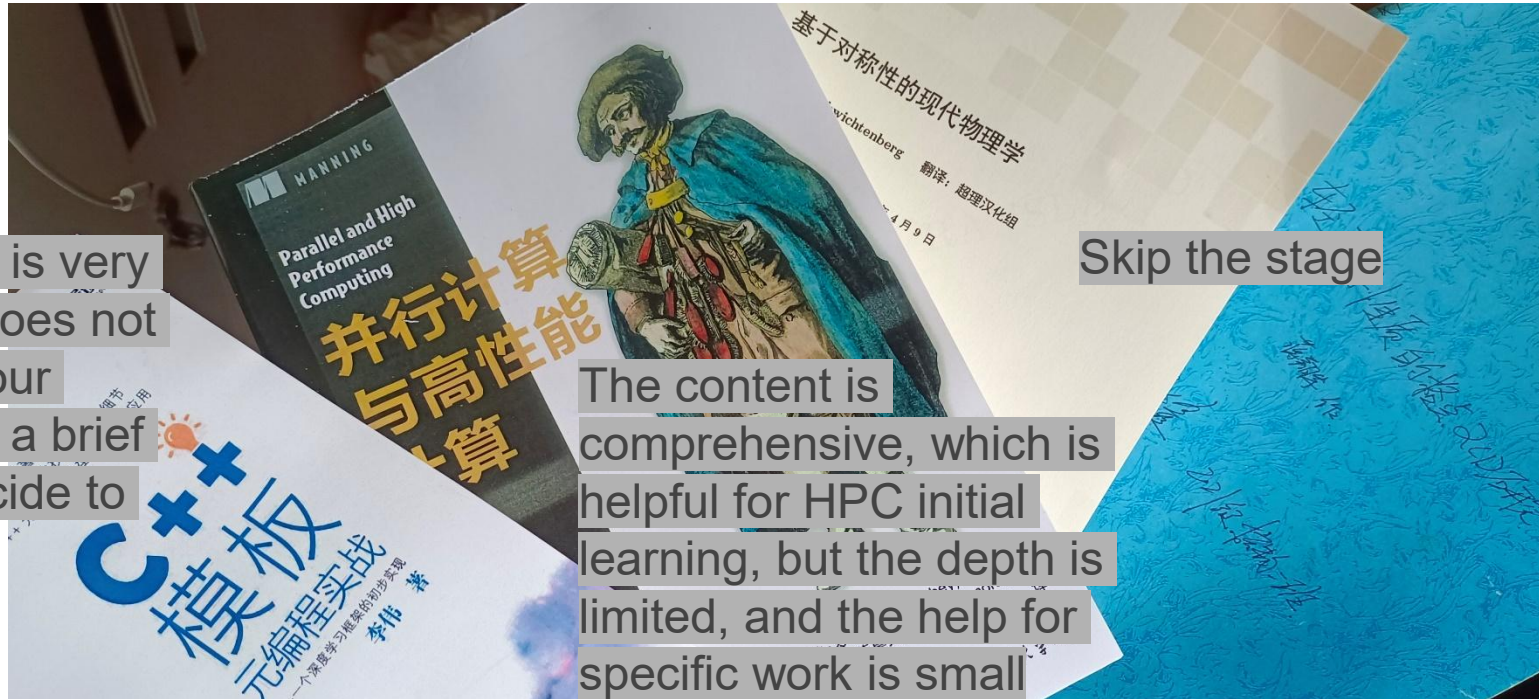
Done!!!

Still unknown

1. ~~Debug clover (multi version).~~----- a long time.
2. Test Performance.----- only a partial analysis of a single threaded program has been completed.
3. Improve wilson dslash.----- may more than a month.

A bit harder, in the process

The content is very tricky, and does not quite meet our needs, after a brief reading, decide to discard



The content is comprehensive, which is helpful for HPC initial learning, but the depth is limited, and the help for specific work is small

Skip the stage

希望大家一起讨论，指正

**我的汇报到此结束
谢谢大家!**