

# Weekly Research Progress Report

Student: Xing Cong

Date: 7/26/2023 - 8/8/2023

## List of accomplishments this week: (工作成果列表)

- 学习钱院士的宣讲 PPT，了解高性能计算的基本知识。
- 阅读最新的 HYCOM 官方介绍书籍--**Software Design Description for the HYbrid Coordinate Ocean Model (HYCOM) Version 2.2**，获得 HYCOM 的并行计算相关内容，渐渐懂得源码包中各个文件所具有的功能。
- 阅读 HYCOM 的运行指导书籍--**Hybrid Coordinate Ocean Model (HYCOM) Version 2.1 User's Guid**，知悉 HYCOM 运行的一般流程。
- 阅读 HYCOM 的介绍书籍-- **HYCOM Version 2.0 User's Manual (Details of the numerical code)**，了解数学公式向 Fortran 代码的转化过程。
- 阅读会议论文 **Cache-aware Sparse Patterns for the Factorized Sparse Approximate Inverse Preconditioner** 部分内容，回顾最优化部分相关知识。

## Paper summary: (文献总结)

Name : 中国的高性能计算: 挑战与进展

Introduction: 汇报内容主要分为三个部分，第一个部分是 2019 年超级计算机发展的最新动态分享和预测，第二部分提出了超级计算机发展需要重视的几个问题，其中包括 E 级超级计算机研制面临的瓶颈问题、我国超算部分技术仍被国外封锁问题、最新的体系结构的发展和创新问题以及超算硬件发展问题和软件应用等问题。第三部分主要介绍了我国的高性能计算重点专项活动的进展情况。

Motivation : 用于在 2019 年联想全球超算峰会上做报告使用，与参会者分享超算的最新动态和相关趋势预测，同时提出超算研究的重点方向和主要问题供大家讨论，也让参会者了解中国的超算发展与应用情况。

Name : Software Design Description for the HYbrid Coordinate Ocean Model (HYCOM)  
Version 2.2

Introduction: 本文档可以大致分为两个主要的部分。第一部分, 主要介绍了 HYCOM (HYbrid Coordinate Ocean Model) 2.2 版本的数学公式以及求解的过程, 它以比较大的篇幅对大气海洋学中的理论知识进行阐述和论证, 涉及了诸多理论的公式, 比如模型运行的边界条件、用于海洋中水柱进行垂直混合的多种模型、温度平衡、水盐平衡、动量模型和引入大气驱动因素的公式等。这部分更像是为大气海洋学相关领域的学者提供的本程序运行的理论依据。第二部分, 主要介绍了 HYCOM 模型中主要子程序的作用以及相关参数的含义, 这个部分对于阅读 HYCOM 2.3 版本的源码具有很重要的参考和指导作用。

Motivation: 这篇文档可以为使用该模型的研究人员提供模型运行的理论依据以及相关公式的推导过程, 同时还可以提供模型对应源代码中相关子程序的大致说明以及参数注释。

Name: Hybrid Coordinate Ocean Model (HYCOM) Version 2.1 User's Guide

Introduction: 本文档主要介绍如何将 HYCOM 程序部署到计算机上并运行。文档主要以大西洋的一小块区域为例, 详细介绍了要将程序在单个计算机上运行起来的主要步骤。随后, 文章补充了在新的研究区域上运行模型的方法和注意事项, 同时介绍了该程序在多个计算机上进行并行处理的四个可选选项 (omp、mpi、ompi 、shmem) 以及对输出的数据数据进行可视化展示的方法。

Motivation: 文档可以一步步指导初次使用该程序的人, 慢慢摸索该程序的使用流程以及相关注意事项。

Name: HYCOM Version 2.0 User's Manual (Details of the numerical code)

Introduction: 本文档和 Software Design Description for the HYbrid Coordinate Ocean Model (HYCOM) Version 2.2 所介绍的内容很类似。文档大部分也是在介绍理论公式和应用, 但文档的组织形式更加便于理解和接受。本文档中, 以章节对应相关理论和源程序文件, 并部分给出程序化的过程以及流程图, 更加便于理解。

Motivation: 这篇文档可以为初次接触该模型的研究人员提供模型运行的基础理论依据以及相关公式的推导过程, 同时提供将公式进行程序化的流程和步骤, 对入门友好。

## Name : Cache-aware Sparse Patterns for the Factorized Sparse Approximate Inverse Preconditioner

Introduction: 本篇文章主要介绍了研究人员面向共轭梯度算法预处理器开发的面向架构的新标准，它可以在不增加时间开销的情况下，增加预条件器的数值效果。实验结果表明，在三种不同的架构上的 72 个测试矩阵上，平均解决时间的显著减少，减少的范围在 12.94%至 22.85%之间。

Motivation: 提出一种面向架构的新的标准，以提高预处理器在不同架构的计算机上运行的数值效果。

## Work summary (工作总结)

### 钱院士 PPT 汇报部分总结和补充

#### ● 超级计算机概述

超级计算机通常由数以千计的处理器核心组成，并具备大规模的内存和高速的存储系统，它被设计用于处理极其复杂和计算密集的任务，这些任务在传统计算机上可能需要花费数天或数周的时间来完成。

根据最新的 Top500 的数据显示，超级计算机的运算速度发展趋势如下：

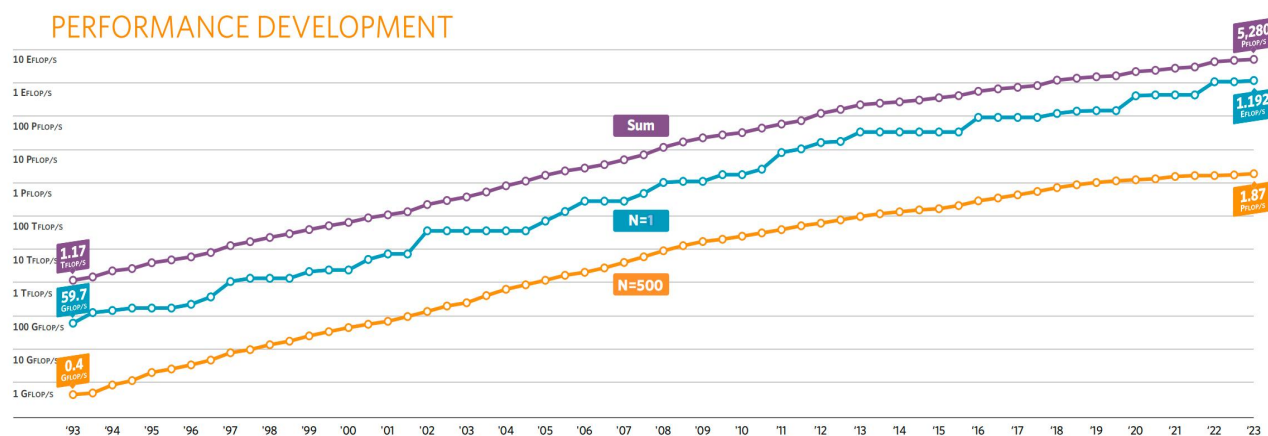


Figure 1 超级计算的发展趋势（截止 2023 年）

由此可见，超级计算机性能排行榜第一名的算力已经达到每秒钟 1.192E 次浮点运算，这和钱院士在 2019 年做的报告中的预测不谋而合。

超级计算机具有如下几个特点：

**处理能力：**超级计算机拥有极高的处理能力，通常以浮点运算速度来衡量。其性能通常以每秒浮点运算次数（FLOPS）来衡量，可以达到百万亿级（petaflops）或者更高。现代的超级计算机甚至可以达到百万亿亿级（exaflops）的性能。比如美国的 **Frontier** 超级计算机。如下图 2 所示。

JUNE 2023			SITE	COUNTRY	CORES	R <sub>MAX</sub> PFLOP/S	POWER MW
1	Frontier	HPE Cray EX235a, AMD Opt 3rd Gen EPYC (64C 2GHz), AMD Instinct MI250X, Slingshot-11	DOE/SC/ORNL	USA	8,699,904	1,194.0	22.7
2	Fugaku	Fujitsu A64FX (48C, 2.2GHz), Tofu Interconnect D	RIKEN R-CCS	Japan	7,630,848	442.0	29.9
3	LUMI	HPE Cray EX235a, AMD Opt 3rd Gen EPYC (64C 2GHz), AMD Instinct MI250X, Slingshot-11	EuroHPC/CSC	Finland	2,220,288	309.0	6.01
4	Leonardo	Atos Bullsequana intelXeon (32C, 2.6 GHz), NVIDIA A100 quad-rail NVIDIA HDR100 Infiniband	EuroHPC/CINEC	Italy	1,824,768	238.7	7.40
5	Summit	IBM POWER9 (22C, 3.07GHz), NVIDIA Volta GV100 (80C), Dual-Rail Mellanox EDR Infiniband	DOE/SC/ORNL	USA	2,414,592	148.6	10.1

Figure 2 世界超级计算机排行榜前五名（数据截止到 2023 年 6 月）

**并行处理：**超级计算机通过将任务划分为多个并行的子任务，并同时在多个处理器核心上执行这些子任务，从而实现高效的并行处理。

**大规模存储：**超级计算机通常配备大规模的存储系统，用于存储海量的数据和计算结果。这些存储系统可以包括高速磁盘阵列等技术，以确保计算过程中的数据读取和写入能够高效进行。

**能耗：**由于超级计算机的巨大处理能力和复杂性，其能耗也非常高。因此，研究人员和工程师在设计超级计算机时也会关注能效问题，力求在性能和能耗之间取得平衡。

● 流行的体系结构

**片内结构：**基于众核，一个主核带众多小核，这是神威太湖之光的体系结构。

**节点内异构：**CPU+加速器结构，一个节点上有一个 CPU，另外会有一个或几个加速器。

● 并行框架

**并行编程框架：**并行计算是高性能计算下的一个细分领域，其主要思想是将复杂问题分解成若干个部分，将每一个部分交给独立的处理器（计算资源）进行计算，以提高效率。

并行编程框架是用于在计算机系统中实现并行计算的软件库、工具集或编程模型。针对不同的问题，并行计算需要专用的并行架构，架构既可以是专门设计的，含有多个处理器的单一硬件或超级计算机，也可以是以某种方式互连的若干台的独立计算机构成的集群。

这些框架旨在简化并行编程的复杂性，并允许程序员更轻松地利用多核处理器、分布式系统或超级计算机等高性能计算平台的潜力。

常见的并行编程框架：

**OpenMP (Open Multi-Processing) :** OpenMP 是一种针对共享内存系统的并行编程模型。它使用基于指令的编译器指示来指定并行化的区域，允许程序员将串行代码转换为并行代码。OpenMP 适用于多核处理器，并且在许多编程语言（如 C, Fortran）中都有支持。

**MPI (Message Passing Interface) :** MPI 是一种用于在分布式内存系统中进行消息传递的并行编程模型。它适用于构建在多个计算节点上运行的并行应用程序，允许节点之间通过消息交换进行通信和同步。

**CUDA (Compute Unified Device Architecture) :** CUDA 是由 NVIDIA 开发的并行编程框架，针对 NVIDIA 的 GPU（图形处理单元）进行并行计算。它允许程序员在 GPU 上执行大规模的并行计算，特别适用于处理图形、深度学习和科学计算等任务。

**OpenCL (Open Computing Language) :** OpenCL 是一种开放标准的并行编程框架，可用于异构计算环境中的并行计算，包括 GPU、CPU 和其他加速器。它提供了跨不同设备的通用并行计算支持。

## ● E 级计算机研制

适用于 E 级计算机的可计算物理建模与新型计算方法

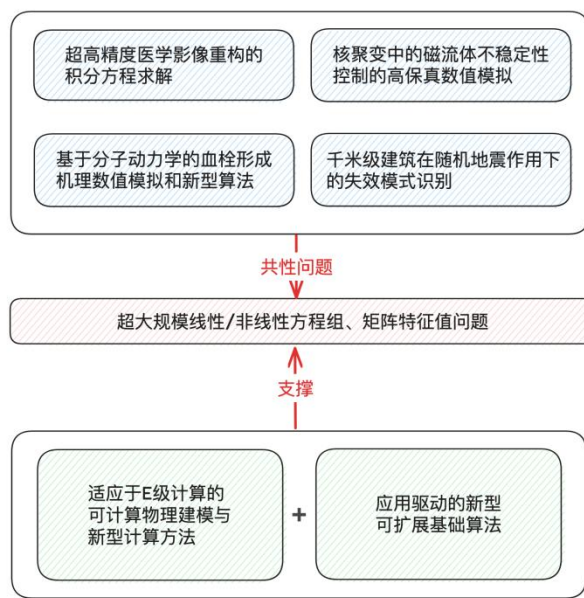


Figure 3 E 级计算机可支撑计算问题

Software Design Description for HYCOM Version 2.2 总结

该文档分为九个章节，前四章以非常简短的语句，对 HYCOM 的背景、目的、程序特点和目录进行了简要的说明。

第五章是本文档比较重要的内容，主要介绍了 HYCOM (HYbrid Coordinate Ocean Model) 2.2 版本的数学公式以及求解的过程，它以比较大的篇幅对大气海洋学中的理论知识进行阐述和论证，涉及了诸多理论的公式。在这一章中，也对程序的部分思想进行了简要说明，即说明了运算过程，从运算过程中可以看到部分需要用到迭代和进行重复计算的部分。

后面三章主要是以表格的形式，对程序各个文件中出现的子程序、子程序中的参数以及公共块进行了简要介绍，如下图 4 这个介绍初始化 kpp 垂直混合算法的子程序 inikpp 的介绍说明。

Subroutine	Description
<i>inikpp</i>	Initializes the Large, McWilliams, and Doney KPP vertical mixing scheme (see Section 5.2.7.1). No arguments are called. <b>Common Blocks:</b> common/kppltr/ <b>I/O:</b> None <b>Uses:</b> mod_xc

Figure 4 inikpp 的介绍

学生认为，后面的部分结合源代码进行理解会更好一些，直接阅读可能并无法获得更加有效的信息，所以，本部分将主要介绍第五章的内容 ----HYCOM Logic and Basic Equations。

● 第五章整体框架

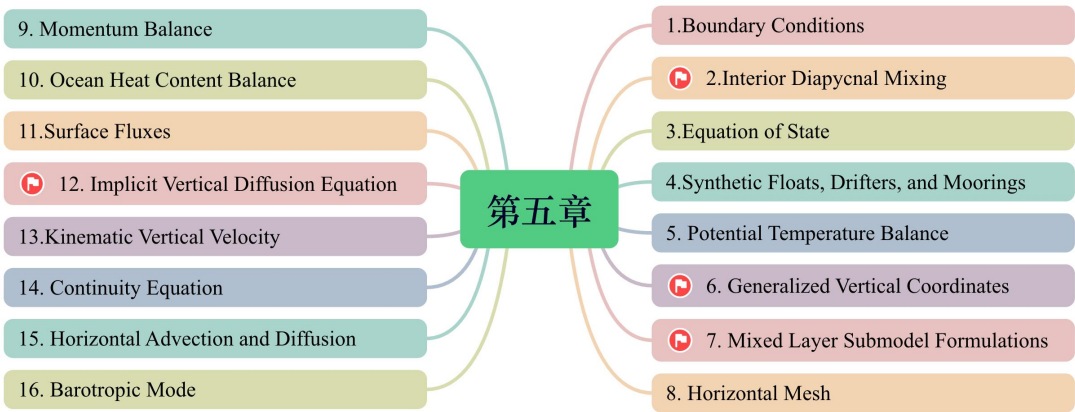


Figure 5 第五章整体框架图

其中, 在 2、6、7、12 等方程的思想介绍中提到循环求解或迭代求解。

- Boundary Condition 边界条件

HYCOM 2.2 模型配置了两种类型的边界条件

1. Newtonian relaxation in sponge layers
2. full open-ocean boundary conditions

该节描述了 HYCOM 处理边界条件的思路,既可用于限制性的边界,也可用于开放性的交互边界,保证了数值稳定性。

- Interior Diapycnal Mixing 内部层间混合

Diapycnal mixing 的作用是在相邻层之间交换质量、盐量和热量, 满足质量、盐量和热量守恒, 维持密度结构的稳定性。在 HYCOM 模型中, 参数化内部海洋垂向混合主要涉及三个算法。如下所示:

1. Hybrid Coordinate Implicit Algorithm (Interior Mixing From KPP) KPP 隐式算法:

该算法需要采用**隐式数值算法**进行求解, 隐式算法通过**构建方程组**来表达变量之间的关系,并通过**迭代求解方程组**来推进计算过程。

2. Hybrid Coordinate Explicit Algorithm 显式混合坐标算法:

计算层间热量、盐量和质量通量, 根据通量更新层厚度、温盐, 每个变量的值直接依据前一时刻变量的值计算得到, 迭代次数较少, **也需要进行迭代求解**,直到密度收敛。

3. Isopycnic Coordinate Explicit Algorithm 显式等密度坐标算法:

不常用。

- Equation of State 状态方程

本节主要介绍使用多项式逼近状态方程以高效计算海水密度, 并提供了参数选项来适应不同的模拟需求, 在混合坐标系统中尽可能维持层间的稳定密度坐标。

- Synthetic Floats, Drifters, and Moorings

本节主要介绍了 HYCOM 中用于模拟和追踪各类浮标的算法思路、空间插值方法、时间推进方案等, 可以模拟浮标在海洋中的运动过程并得到高时间分辨率的数值结果。

模型支持计算和跟踪 4 种类型的浮标: 3D 拉格朗日浮标、等密度浮标、等压力浮标、固定浮标(系留装置)。

- Generalized Vertical Coordinates 广义的垂直坐标

HYCOM 使用混合坐标,允许不同区域使用不同类型的垂直坐标,如压力坐标、等密度坐标和 Sigma 坐标。

模型为每个层指定一个最小层厚,坐标调整算法将尽量保持层厚超过该最小值。在开放海洋区,层厚最小值按照海底深度的函数指定,用于压力-等密度坐标的转变。在近海区,层厚最小值采用不同的规定,用于压力-Sigma 坐标的转变。

因此,该节主要介绍了 HYCOM 模型中的混合垂直坐标的调整算法,介绍了 HYCOM 模型混合坐标系统的设计思想和实现方法,既适合开放海洋,也适合近海和边界层,实现了通用的海洋坐标系统。

- Mixed Layer Submodel Formulations

该节主要介绍了 HYCOM 模型中用于混合层物理过程的参数化的几个子模型。

分别是 KPP 模型、Mellor-Yamada 模型、Price-Weller-Pinkel 模型、Kraus-Turner 模型、GISS 模型。

HYCOM 提供了多种混合层模型以供选择,用户可以根据具体的应用需求选择合适的模式。这些模式有不同的理论基础,数值算法也各不相同。

1. KPP Vertical Mixing Algorithm (kpp 垂直混合算法)

**KPP 算法需要进行多次迭代求解。**

第一次迭代:第一次迭代是在初始模型变量剖面中计算的模型界面处的扩散/粘度系数的垂向剖面。然后通过求解每个网格点上的一维垂向扩散方程来混合模型变量。**求解过程涉及到问题矩阵的形成和三对角矩阵的求逆。**

第二次迭代,在第二轮迭代中,模型变量的垂直混合剖面估算出新的扩散率/粘度剖面,然后将模型变量的原始剖面进行混合。

重复执行此过程,直到模型变量的混合剖面与上一次迭代的混合剖面相差不大为止。

2. The Mellor-Yamada Level 2.5 Turbulence Closure Model (MY 2.5)

3. Price-Weller-Pinkel Dynamical Instability Vertical Mixing Algorithm (PWP)

4. Krauss-Turner Mixed Layer Models (KT)

5. GISS Mixed Layer Model



## ● Implicit Vertical Diffusion Equation 隐性垂直扩散方程

本节主要介绍了一种统一高效的隐式数值框架来解决垂直扩散方程。基本思想是，在每格点解决一维垂直扩散方程。随后，分解变量为平均和波动成分，将垂向扩散表达为梯度和扩散系数的函数，参数化各向异性扩散系数。然后，构建对角矩阵表示各层的耦合关系。之后，求解矩阵方程组得到更新的变量向量。最后，迭代求解到收敛。

## Hybrid Coordinate Ocean Model (HYCOM) Version 2.1 User's Guid 总结

本篇文档主要介绍如何将 HYCOM 程序部署到计算机上并运行。

文档主要以大西洋的一小块区域为例，详细介绍了要将程序在单个计算机上运行起来的主要步骤。随后，文章补充了在新的研究区域上运行模型的方法和注意事项，同时介绍了该程序在多个计算机上进行并行处理的四个可选选项（omp、mpi、ompi、shmem）以及对输出的数据数据进行可视化展示的方法。

## ● 程序运行的流程图

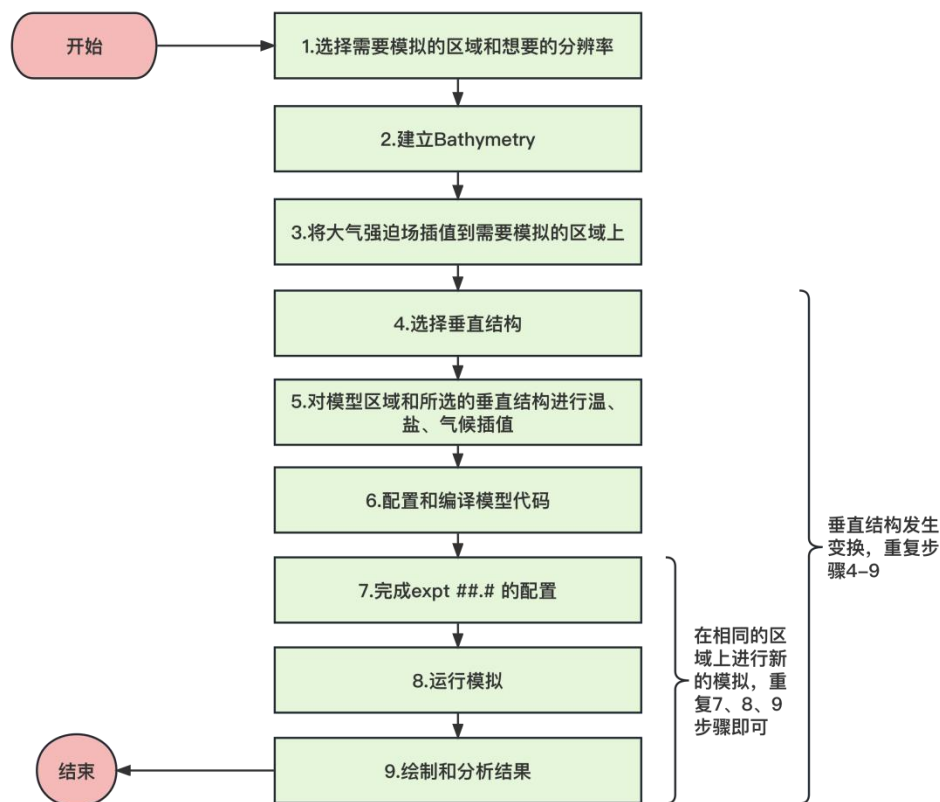


Figure 6 HYCOM 2.1 程序运行的流程图

大致步骤如下:

1. 选择一个要模拟的区域和想要的分辨率。

2. HYCOM 模型建立水深场 (Build a bathymetry)

水深场是描述海洋水体厚度或海底地形的一个三维数据集。

3. 将大气强迫场插值到需要模拟的区域上 (Interpolate atmospheric forcing to the domain)  
需要提供大气强迫作为模型的驱动力

4. 选择垂直结构 (在 `expt###` 或 `blkdat.input` 文件中)。(Choose vertical (isopycnal) structure)

用户可以通过在 `blkdat.input` 文件中设置参数来选择使用三种坐标的组合,以及具体的层厚度/层数等设置,从而定义模型的垂直结构。

5. 对模型领域和所选的垂直结构进行温盐气候插值。(Interpolate T or S climatology to the model domain and the chosen vertical structure)

在海洋模型的初始化过程中,需要提供温度和盐度场作为模型的初始状态。这些初始场通常来自气候学数据集,如 Levitus。气候数据集的格点结构与模型使用的格点结构不一致,因此需要进行插值。

6. 配置和编译模型代码。(Configure and compile the model code)

配置和编译对每个模拟系统都需要进行一次。配置 `makefile` 后直接调用 `make` 命令即可自动完成编译。

7. 完成 `expt###` 的配置。(complete configuration of `expt`)

在 HYCOM 中,每个模拟实验都有一个对应的目录,例如 `expt 01.0`。在这个目录下有多个配置文件,需要进行设置才能运行该实验的模拟。

8. 运行模拟。(Run the simulation)

在完成了模拟实验 `expt###` 的所有配置和编译准备工作后,就可以运行这个模拟案例了

9. 绘制和分析结果。(Plot and analyze results)

## ● HYCOM 通信程序

大多数通信例程充当将处理器状态同步的隐式屏障。在 HYCOM 2.1 版本中,为每个子程序提供了两个版本的通信程序包,其一是, `mod_xc_mp.F` 用于消息传递,另一个是 `mod_xc_sm.F` 用于单个处理器。

该例程被配置为一个模块，所有 HYCOM 例程应该以 “use mod\_xc” 开头，以便在需要时调用 HYCOM 通信例程。通信子程序如下图所示：

Table 21: HYCOM Communication Routines

Routine	Description
XCAGET	Converts an entire 2-D array from tiled to non-tiled layout.
XCAPUT	Converts an entire 2-D array from non-tiled to tiled layout.
XCEGET	Finds the value of $a(ia, ja)$ on the non-tiled 2-D grid.
XCEPUT	Fills a single element in the non-tiled 2-D grid.
XCHALT	Emergency stops all processes, called by one process.
XCLGET	Extracts a line of elements from the non-tiled 2-D grid.
XCLPUT	Fills a line of elements in the non-tiled 2-D grid.
XCMAXR	Replaces array $a$ with its element-wise maximum over all tiles.
XCMINR	Replaces array $a$ with its element-wise minimum over all tiles.
XCSPMD	Initializes processor data structures, called once.
XCSTOP	Stops all processes, called by all processes.
XCSUM	Sum of a 2-D array.
XCSUMJ	Row-sum of a 2-D array.
XCSYNC	Barrier, no processor exits until all arrive (flush STDOUT).
XCTBAR	Syncs with processors ipe1 and ipe2 (internal use only).
XCTILR	Updates the tile overlap halo of a 3-D real array.
XCTMRI	Initializes timers.
XCTMR0	Starts timer.
XCTMR1	Adds time since call to XCTIM0 to timer.
XCTMRN	Registers name of timer.
XCTMRP	Prints all active timers.

Figure 7 HYCOM 中负责通信的子程序及其介绍

- HYCOM 中的并行处理操作

HYCOM 可以通过以下几种方式在多个处理器上运行：

在共享内存多处理器上使用 OpenMP 线程（例如，Compaq ES40，Sun E10000）。

在具有全局共享内存的机器上使用 SHMEM（例如，Cray T3E，SGI Origin 2800/3800）。

在任何均匀的“连接”处理器集上使用 MPI（例如，IBM SP，集群，上述所有机器）。

在共享内存多处理器集群上同时使用 MPI 和 OpenMP（例如，IBM SP Power 3，Compaq AlphaServer SC）。

- 运行比较 Pipe.f

源代码在 pipe.f 中控制使用命名管道来比较两个相同的运行，即“主”和“从”。

它们通常只在使用处理器的数量上有所区别（即 NOMP 和 NMPI 的值）。

通常，主节点位于单个处理器上（即  $NMPI = 0$  且  $NOMP = 0$  或  $1$ ）。对于从节点发送给主节点的每个数组元素，都进行比较。默认情况下，在每个时间步的每个主要阶段之后，会比较大多数重要的模型数组。如果检测到错误（即主节点与从节点的差异），则可以在引入错误的子程序中添加附加的 `COMPARALL` 或 `COMPARE` 调用，以确定需要修改哪个 OpenMP 循环。每次运行的第一个任务必须在同一节点上（即相同的操作系统映像上），才能使管道正常工作。

## **HYCOM Version 2.0 User's Manual (Details of the numerical code) 总结**

本文档和 Software Design Description for the HYbrid Coordinate Ocean Model (HYCOM) Version 2.2 所介绍的内容很类似。文档大部分也是在介绍理论公式和应用，但文档的组织形式更加便于理解和接受。本文档中，以章节对应相关理论和源程序文件，并部分给出程序化的过程以及流程图，更加便于理解。

### ● 提到的迭代计算

迭代计算主要基于十个主要过程子程序，具体在程序中迭代哪个子程序，取决于选择哪个模型选项

注意：KT 模型只控制混合层内的混合，而 KPP 模型提供从表面到底部的混合。

#### 1. the non-slab K-Profile Parameterization model (KPP)

subroutine cnuity: continuity equation  
subroutine tsadv: advection equation  
subroutine momtum: momentum equations  
subroutine barotp: dynamic barotropic mode  
subroutine thermf : ocean-atmosphere exchanges  
subroutine icloan : ocean-ice exchanges  
subroutine mxkpp : k-profile vertical mixing  
subroutine hybgen: vertical coordinate remapping.

#### 2. Kraus-Turner (KT) model

subroutine cnuity: continuity equation  
subroutine tsadv: advection equation  
subroutine momtum: momentum equations

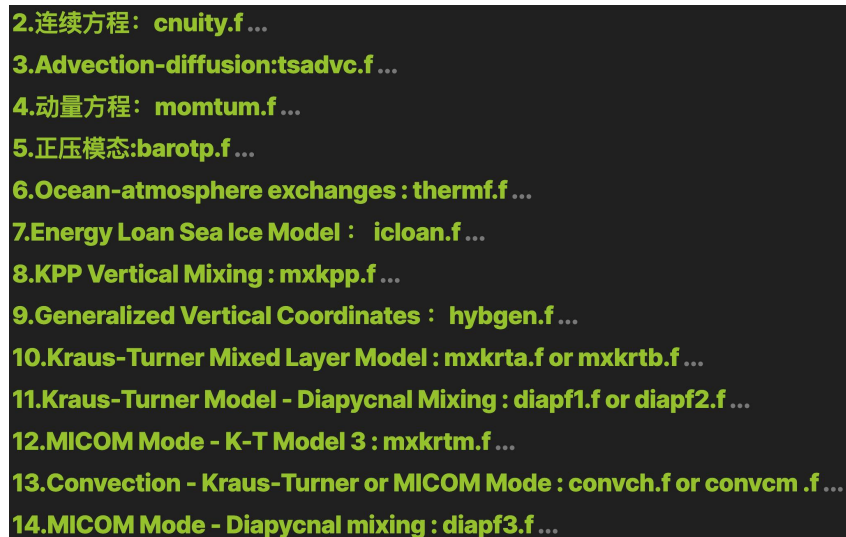
subroutine barotp: dynamic barotropic mode  
subroutine thermf : ocean-atmosphere exchanges  
subroutine icloan : ocean-ice exchange  
subroutine mxkrta or mxkrb: bulk surface mixed layer  
subroutine convch: vertical cvection  
subroutine diapf1 or diapf2: diapycnal mixing (explicit, implicit)  
subroutine hybgen: vertical coordinate remapping.

### 3. HYCOM run in MICOM 的兼容模式

subroutine cnuity: continuity equation  
subroutine tsadv: advection equation  
subroutine momtum: momentum equations  
subroutine barotp: dynamic barotropic mode  
subroutine thermf : ocean-atmosphere exchanges  
subroutine icloan : ocean-ice exchanges  
subroutine mxkrm : bulk surface mixed layer  
subroutine convcm: vertical convection  
subroutine diapf3: diapycnal mixing (explicit MICOM mode).

每次迭代，输出文本文件和图像文件以及一次对照测试数据。这些文档中所说的需要迭代计算的子程序，往往都是在源文件中，以单独的.f90 文件存在。

#### ● 文档结构



2.连续方程: **cnuity.f** ...  
3.Advection-diffusion:**tsadv.f** ...  
4.动量方程: **momtum.f** ...  
5.正压模态:**barotp.f** ...  
6.Ocean-atmosphere exchanges : **thermf.f** ...  
7.Energy Loan Sea Ice Model : **icloan.f** ...  
8.KPP Vertical Mixing : **mxkpp.f** ...  
9.Generalized Vertical Coordinates : **hybgen.f** ...  
10.Kraus-Turner Mixed Layer Model : **mxkrta.f** or **mxkrb.f** ...  
11.Kraus-Turner Model - Diapycnal Mixing : **diapf1.f** or **diapf2.f** ...  
12.MICOM Mode - K-T Model 3 : **mxkrm.f** ...  
13.Convection - Kraus-Turner or MICOM Mode : **convch.f** or **convcm.f** ...  
14.MICOM Mode - Diapycnal mixing : **diapf3.f** ...

Figure 8 HYCOM v2.0 文档结构

文档的章节结构如上图 8 所示。上面的几个章节所介绍的文件中包含的程序，正是文章在开头中介绍的迭代计算的十个过程子程序。

每一章都具有固定的格式来进行阐述，首先是公式定理推导和运算以及转化为程序的过程参数，其次是运行过程介绍，最后给出程序的参数含义和公式与程序中参数的对照表。

总体而言，介绍的内容基本上和 SDD for HYCOM v2.2 是一致的。

## **Cache-aware Sparse Patterns for the Factorized Sparse Approximate Inverse Preconditioner 总结**

该论文目前还没阅读完，最近去学习了最优化的经典算法。目前大致明白了利用共轭梯度法来快速求解  $Ax = b$  的原理（其中  $A$  必须是对称且正定的矩阵）。

假如求解的  $x$  变量是  $N$  维向量的话，利用共轭梯度法来进行求解，则可以在最多经历  $N$  次迭代的情况下，找到  $x^*$  使得构造的函数具有最小值，即为  $Ax = b$  的解。而经过证明，可以得出当特征值存在重叠时，只需要更少的迭代次数就可以收敛。也就是说，当特征值分布越集中时，收敛的速度就越快。因此，就引入了预处理策略来改善那些特征值分布不佳的情况，从而加快迭代的速率。而最佳的预处理策略就是对  $A$  直接做个变换，让变换后的矩阵具有更为集中的特征值，这个参与变换的矩阵就是稀疏矩阵  $C$ 。

使用稀疏矩阵的好处在于，不需要矩阵分解和求逆，只需要上次迭代的步长和当前的梯度，即可计算下次的步长。计算量低且对内存十分友好。

为了进一步加快该算法的效率，该论文从该算法的预处理策略入手，引入一个关于预处理器开发的面向架构的新标准，它可以在不增加时间开销的情况下，增加预处理器的数值效果，进一步提高该算法在超级计算机上的执行效率。

### **Next (下一步)**

- 结合源码和文献，跟着老师的指导，对 HYCOM 项目进行更近一步的学习。
- 学习有关最优化算法相关的知识