#### 数值算法代码的说明

编译环境：OpenFOAM-v2006

代码主要包含两部分：1）相变模型类库的代码；2）求解器的代码。

建议将自己写的类库放在路径：$WM\_PROJECT\_USER\_DIR/src，建议将求解器放在路径：$WM\_PROJECT\_USER\_DIR/applications，建议将算例放在：$WM\_PROJECT\_USER\_DIR/run，其中“$WM\_PROJECT\_USER\_DIR”为OpenFOAM的环境变量，表示一个路径。

相变模型类库的说明：代码主要包括相变潜热，饱和温度和相变模型三部分。

相变模型类库的编译方式：在phaseChangeModels文件夹下（该文件夹下包含一个Make文件夹，用于存在编译相关的指令）运行wmake。

求解器的说明：求解器的代码在applications下，主要实现了如下几个求解器。

|  |  |
| --- | --- |
| 可压缩+相变 | compressibleInterPhaseChangeFlow |
| 可压缩+相变+流固共轭传热 | multiRegionPhaseChangeFlow |
| 不可压缩+相变 | thermoPhaseChangeFlow |
| 不可压缩+相变，采用MULES代数重构算法，其他的求解器都采用isoAdvecor几何重构算法 | thermoPhaseChangeFoam |
| 可压缩+相变，同compressibleInterPhaseChangeFlow，通过定义旋转轴位置和旋转角速度对二维模型添加离心力。 | centrifugalForce/compEvapCondFlow |
| 可压缩+相变+流固共轭传热，同multiRegionPhaseChangeFlow，通过定义旋转轴位置和旋转角速度对二维模型添加离心力。 | centrifugalForce/chtCompEvapCondFlow |

所有求解器都支持旋转坐标系MRF，但是MRF需要采用三维模型进行仿真。所以如果采用二维模型的话，实现了两个额外的直接添加离心力的求解器。

求解器的编译方式：进入各个求解器所在的文件夹，运行wmake，完成编译。

编译求解器代码之前需要进行的其他修改：

1. 进入$WM\_PROJECT\_DIR/applications/solvers/multiphase/compressibleInterFoam/VoFphaseCompressibleTurbulenceModels，分别替换compressibleInterPhaseTransportModel.H和compressibleInterPhaseTransportModel.C，建议替换前先备份。这里的目的是添加了函数kappaEff()，返回mixture的导热系数。然后编译。
2. 进入$WM\_PROJECT\_DIR/src/transportModels/interfaceProperties，分别替换interfaceProperties.H和interfaceProperties.C，建议替换前先备份。这里的目的是采用smeared过后的相体积分数计算曲率半径，减小误差。然后编译。

数值算法代码的编译顺序：首先编译类库的代码（包括源代码修改的部分），然后编译求解器的代码。

#### 相应算例的说明

一个算例的文件结构如下：

<case>

├── 0.orig //初始条件和边界条件，其中0.orig运行时必须替换成0，0.orig是备份

│   ├── alpha.water

│   ├── p

│   ├── p\_rgh

│   ├── T

│   └── U

├── Allclean //脚本，清除所以

├── Allrun //脚本，运行算例

├── constant //物性参数，网格，重力加速度，相变模型等

│   ├── g

│   ├── polyMesh //网格

│   ├── thermophysicalProperties //表面张力，潜热，饱和温度，相变模型的相关参数

│   ├── thermophysicalProperties.steam //气相的物性参数

│   ├── thermophysicalProperties.water //液相的物性参数

│   └── turbulenceProperties

└── system

├── blockMeshDict //用于定义简单网格的字典文件

├── controlDict //计算控制条件，时间步长，读写等

├── decomposeParDict //并行计算

├── fvSchemes //离散格式

├── fvSolution //线性代数方程求解器参数设置，收敛残杀设置，迭代设置等

├── probes

├── setFieldsDict //数值仿真的Initial Condition

└── singleGraph

其中共轭传热求解器算例的文件结构有所不同，具体可参考一个相关算例。

在$WM\_PROJECT\_USER\_DIR/run文件夹下有许多算例，可以拷贝一份算例，然后重新运行一下，多运行几个算例能够快速了解求解器是怎么用的，建议手动在Terminal中输入Allrun中的指令，而不是直接运行Allrun脚本，同时多多主要日志文件。

运行算例是在controlDict中可以通过application对应的关键词知道该算例运行的是哪一个求解器。

在Best Practice Guideline文件夹中包含了相变模型，常见物性参数，fvSchemes和fvSolution应该如何设置参数的示例，新建算例，或遇到收敛问题时可以参考。

#### 后处理

1. 通过paraview进行后处理。1）在运行完成算例之后，在Terminal输入paraFaom直接打开该算例的后处理。2）在运行完成算例之后，添加一个.foam为后缀名的文件（如case.foam），通过paraview打开该文件进行后处理。
2. 通过OpenFOAM自带的后处理工具，提取数据，进行后处理，具体可参加OpenFOAM User Guide。
3. 通过写代码进行后处理，也可以进行初始化或者定义边界条件。
4. 如果遇到难以解决的复杂后处理问题，call me on WeChat.

同时存在一些其他很厉害的第三方工具，可以提高效率，如swak4Foam，gdbof (debug tool)，如果常用可以考虑了解。

#### 划分网格的工具

对于OpenFOAM的算例来说，如果是单区域网格（single region），可以采用多种网格划分工具，具体看个喜好。如果是多区域网格（multi-region），用于共轭传热求解器，snappyHexMesh和SALOME可以用来这类网格。两种网格划分工具都有点复杂，任选一种。具体可以google it.

#### 热管的数值模拟

关于热管的数值模拟，在路径$WM\_PROJECT\_USER\_DIR/run/study下有多个相关的算例，都可以参考。我这里提供了两个算例：1）WM\_PROJECT\_USER\_DIR/run/HP，参考文献CFD modelling of a two-phase closed thermosyphon charged with R134a and R404a，重力热管的数值模拟。2）WM\_PROJECT\_USER\_DIR/run/RHP，参考穆英杰微小旋转热管实验，旋转热管的数值模拟。在SALOME文件夹中提供了他们各自的网格是如何划分的。

注意：在WM\_PROJECT\_USER\_DIR/run/RHP/constant中有一个名为centrifugalForceProperties的文件，用于定义二维模型的旋转轴和离心力，如果有的话。

旋转热管采用三维模型数值模拟的算例也有，但是没有经过验证。三维模型通过MRFProperties定义旋转，其他相关参数设置没有什么区别。可参考算例WM\_PROJECT\_USER\_DIR/run/chtCylinder3D-salome或WM\_PROJECT\_USER\_DIR/run/chtCylinder3D-snappy设置新的三维模型旋转热管算例。

热管数值模拟的建议：

1. 采用湍流模型，数值算法支持湍流模型，可以考虑湍流。
2. 旋转热管采用Schrage相变模型，重力热管采用Lee相变模型。
3. 注意关注计算过程中的非连续性误差，不能太大了，容易导致发散。

“文档”文件夹包含了一些参考文档和文献，如果用得上的话，可以看一下。