به نام خدا

انتقال حرارت جابجایی طبیعی در یک شکاف عمودی با استفاده از نرم افزار اپن فوم

بر روی اپن فوم نسخه ۴

حسام رمضان زاده

DO NOT COPY (www.novin-eng.ir)

هدف از یک توتوریال یا کیس آموزشی اپن فوم این است که چهار چیز عمده را به کاربر آموزش دهد: اول،اینکه چطور از نرم افزار استفاده کنیم، دوم، اینکه تئوری پشت مساله و حلگر پیش رو چیست، سوم، اینکه چطور تغییرات در نرم افزار اعمال میشوند و چهارم اینکه چطور مساله را تغییر دهیم تا به خواسته خود برسیم.

خواننده این آموزش موارد زیر را فرا خواهد گرفت:

چگونگی استفاده:

- چگونه از حلگرهای موجود در اپن فوم برای شبیه سازی انتقال حرارت جابجایی در یک شکاف عمودی استفاده کند.
- چگونه از مدل های ویسکوزیته در اپن فوم استفاده کند.
- چگونه از ابزار شار حرارتی دیواره (wall heat flux) برای محاسبه شار حرارتی کل یک مرز استفاده کند.

تئوری مساله و حلگر:

- تئوری پشت حلگرهای buoyantBossinesqSimpleFoam و buoyantSimpleFoam.
- چگونه مدل های ویسکوزیته، مدل های انتقال و مدل های توربولانسی در اپن فوم با هم لینک شده اند.

چگونگی اعمال کیس:

- چگونه یک کیس را با استفاده از حلگرهای buoyantBossinesqSimpleFoam و buoyantSimpleFoam تنظیم کنیم.
- چگونه مدل ویسکوزیته را در کیس خود تنظیم کنیم.

چگونه کیس و حلگر خود را تغییر دهیم:

- چگونه مدلهای ویسکوزیته موجود را تغییر دهیم و یک مدل ویسکوزیته وابسته به دما ایجاد کنیم.

DO NOT COPY (www.novin-eng.ir)

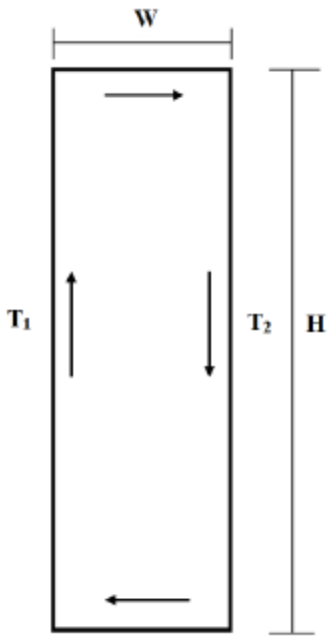
۱- معرفی

۱-۱- انتقال حرارت جابجایی در یک شکاف عمودی

انتقال حرارت طبیعی در یک محفظه عمودی با دو دیواره با دماهای متفاوت شامل تعاملات پیچیده ای مابین سیال و دیواره می‌شود که منجر به تشکیل الگوهای جریان مختلفی خواهد شد. شکل ۱-۱ تصویر سه بعدی شکاف عمودی را نشان میدهد. بیایید تا انتقال حرارت جابجایی طبیعی را به صورت دو بعدی در یک شکاف عمودی در نظر بگیریم که در آن دو مرز دیواره های افقی دارای شرط مرزی دما ثابت هستند. در داخل شکاف، سیال در طول دیواره گرم بالا میرود، در بالا میچرخد و در طول دیواره سرد فرو میریزد و دوباره بر میگردد. این قضیه حرکتی از سیال را شکل میدهد که در شکل ۱-۱ با پیکان نشان داده شده است. پارامترهای اصلی که جریان را تعریف میکنند عدد رایلی(Ra)، عدد پرانتل (Pr) و نسبت منظری (ارتفاع/عرض) هستند. این آموزش شامل تحلیل عددی دوبعدی از انتقال حرارت جابجایی طبیعی در یک شکاف عمودی با استفاده از نرم افزار اپن فوم است. این آموزش یک آموزش نتیجه محور و نمودار محور نیست بلکه بیشتر بر روی حلگر و کتابخانه های اپن فوم تاکید دارد.

در این آموزش، شبیه سازی ها برای یک شکاف عمودی با ارتفاع ۳۰۰mm (H) و عرض ۱۵mm (W) انجام شده است که این ابعاد منجر به نسبت منظری ۲۰ میشود (شکل ۱-۱ را ببینید). دمای دیواره گرم ۳۲۳K (T1) و دمای دیواره سرد ۲۷۳K (T2) در نظر گرفته شده اند. در واقع یک اختلاف دمای ۵۰ درجه ای بین دو دیواره عمودی وجود دارد. سیال درون شیار روغن سیلیکون است. عدد پرانتل روغن سیلیکون ۵۰ در نظر گرفته شده است.

DO NOT COPY (www.novin-eng.ir)



شکل ۱-۱: تصویر دو بعدی شکاف عمودی

۱-۲- حلگر های اپن فوم در زمینه انتقال حرارت

حلگرهایی که در اپن فوم برای شبیه سازی انتقال حرارت موجود هستند از قرار زیر اند:

- buoyantBoussinesqSimpleFoam : حلگر حالت پایا برای جریان های توربولانسی و شناوری سیال های غیر قابل تراکم.
- buoyantBoussinesqPimpleFoam : حلگر حالت گذرا برای جریان های شناوری و توربولانسی سیال های غیر قابل تراکم.
- buoyantSimpleFoam : حلگر حالت پایا برای جریان های شناوری و توربولانسی سیال های قابل تراکم.
- buoyantPimpleFoam: حلگر حالت گذرا برای جریان های شناوری و توربولانسی سیال های قابل تراکم.
- chtMultiRegionSimpleFoam: حلگر حالت پایا برای جریان های شناوری و توربولانسی به همراه انتقال حرارت کانجوگیت ما بین سیال و جامد.

DO NOT COPY (www.ovin-eng.ir)

- chtMultiRegionFoam: حلگر حالت گذرا برای جریان های شناوری و توربولانسی به همراه انتقال حرارت کانجوگیت ما بین سیال و جامد.

چون کیس ارائه شده شامل جریان های بویانسی و توربولانسی میشود، حلگر هایوپ buoyantSimpleFoam buoyantBoussinesqSimpleFoam در نظر گرفته شده اند. حلگرهای گذرای buoyantBoussinesqPimpleFoam و buoyantPimpleFoamبه دلیل محدودیت های زمانی در نظر گرفته نشدند.

۱-۳- دید کلی ای از کیس آموزشی

این آموزش به طور کلی به دو بخش تقسیم میشود:

- استفاده کردن از حلگر buoyantBoussinesqSimpleFoam و اعمال یک مدل ویسکوزیته جدید وابسته به دما برای کیس شکاف عمودی.
- استفاده کردن از حلگر buoyantSimpleFoam برای همان کیس و محاسبه کردن شار حرارتی دیواره در مرزها با استفاده از ابزار موجود در نرم افزار اپن فوم.

۲- توضیح حلگر buoyantBoussinesqSimpleFoam

این بخش به تئوری های اساسی و معادلات حاکم بر حلگر که به وسیله حلگر buoyantBoussinesqSimpleFoam مورد استفاده قرار میگیرند، میپردازد.

۲-۱- توضیح حلگر

کد منبع حلگر و معادلات حاکم حل شده به وسیله حلگر در این بخش شرح داده شده اند. کد منبع (سورس کد) حلگر در مسیر زیر واقع شده است:

DO NOT COPY (www.novin-eng.ir)

```
$FOAM_SOLVERS/heatTransfer/buoyantBoussinesqSimpleFoam
```

```
buoyantBoussinesqSimpleFoam
├── buoyantBoussinesqSimpleFoam.C
├── createFields.H
├── Make
│   ├── files
│   └── options
├── readTransportProperties.H
├── UEqn.H
├── TEqn.H
└── pEqn.H
```

سورس کد اصلی حلگر buoyantBoussinesqSimpleFoam.C است.که شامل فایل های پیش فرض H. میشود که در آن قرار گرفته اند.

```
#include "fvCFD.H"
#include "singlePhaseTransportModel.H"
#include "turbulentTransportModel.H"
#include "radiationModel.H"
#include "fvOptions.H"
#include "simpleControl.H"


int main(int argc, char *argv[])
{
    #include "postProcess.H"

    #include "setRootCase.H"
    #include "createTime.H"
    #include "createMesh.H"
```

```
    #include "createControl.H"
    #include "createFields.H"
    #include "createFvOptions.H"
    #include "initContinuityErrs.H"
```

هدف فایل های H. به طور خلاصه در زیر توضیح داده شده است:

- **#include "fvCFD.H"**: این هدر، فایل هدر استاندارد برای متد حجم سیال در اپن فوم است. این عبارت به نوبه خود شامل تعداد زیادی فایل های هدر از کلاسی میشود که در حلگرهای حجم محدود مورد استفاده قرار گرفته اند.
- **#include "singlePhaseTransportModel.H"**: این کلاس برای مدل انتقال بر پایه‌ی ویسکوزیته برای جریان های تراکم ناپذیر تکفاز است.
- **#include "turbulentTransportModel.H"** : این کلاس برای انتخاب مدل توربولانسی و ضرایب آن است.
- **#include "radiationModel.H"** : این کلاس برای مدلسازی انتقال حرارت تشعشعی است.
- **#include "fvOptions.H"** : این کلاس برای fvOptions در اپن فوم است.
- **#include "simpleControl.H"** : کلاس کنترل SIMPLE برای عرضه اطلاعات مربوط به همگرایی و چک کردن آن در حلقه SIMPLE است.

فایل های H. که در زیر می آیند، در حلقه اصلی قرار دارند. این کد ها، کدهای خاصی هستند که برای عملکردهای خاصی قرار داده شده اند.

- **#include "postProcess.H"** : برنامه های FunctionObjects را اجرا میکند تا نتایج موجود را پس پردازش کند.
- #include **"setRootCase.H"** : ساختار پوشه کیس را چک میکند.
- **#include "createTime.H"** : runtime را مطابق با controlDict چک میکند و متغیرهای زمانی را مقدار دهی اولیه میکند.
- **#include "createMesh.H"**: مش را برای runtime می سازد.
- **#include "createControl.H"** : الگوریتم کنترل حل را تعریف میکند.
- **#include "createFields.H"** : میدان ها را برایدامنه می سازد، به عنوان مثال U، p، T، DT، phi.

DO NOT COPY (www.novin-eng.ir)

- **#include "createFvOptions.H"** : گزینه های FV (finite volume) را تعریف میکند.
- **#include "initContinuityErrs.H"** : خطاهای پیوستگی را اعلان و مقدار دهی اولیه میکند.

بخش بعدی کد، شامل لوپ های (چرخه های) حلگر میشود که میدان ها را محاسبه میکند. میدان runtime تنظیم میشود و معادلات برای سرعت (UEqn)، فشار (pEqn) و انرژی (TEqn) بارگذاری و حل میشوند.

```
Info<< "\nStarting time loop\n" << endl;

while (simple.loop())
{
    Info<< "Time = " << runTime.timeName() << nl << endl;

    // Pressure-velocity SIMPLE corrector


    {
        #include "UEqn.H"
        #include "TEqn.H"
        #include "pEqn.H"
    }

    laminarTransport.correct();
    turbulence->correct();

    runTime.write();

    Info<< "ExecutionTime = " << runTime.elapsedCpuTime() << " s"
        << "  ClockTime = " << runTime.elapsedClockTime() << " s"
        << nl << endl;
}

Info<< "End\n" << endl;

return 0;
```

فایل createFields.H در دایرکتوری واقع شده است که خواص ترمو فیزیکی (thermo-physical) و انتقالی (transport) را به عنوان میدان هایی که به وسیله حلگر استفاده خواهند شد، ایجاد میکند. خواص ترموفیزیکی شامل

دما (T)، سرعت (U) و فشار (p_{rgh}) هستند. فشار کل به صورت $\rho \cdot g + p_{rgh}$ بیان میشود. ویژگی های انتقال شامل چگالی (ρ) و ویسکوزیته مطلق (ν) میشود. چگالی به وسیله فرض بوسینسک بیان یمشود که به صورت زیر داده میشود:

$$\rho = 1 - \beta(T - T_{ref})$$

در اینجا β ضریب انبساط دمایی (1/K)، T(K) و T_{ref}(K) به ترتیب دما و دمای مرجع هستند. ویسکوزیته مطلق به وسیله singlePhaseTransportModel تعریف شده است که یک مدل انتقال بر پایه ویسکوزیته است که برای جریان های تکفاز به کار میرود.

معادلات حاکم برای سرعت در UEqn.H حل شده است.

```
tmp<fvVectorMatrix> tUEqn
(
    fvm::div(phi, U)
  + MRF.DDt(U)
  + turbulence->divDevReff(U)
 ==
    fvOptions(U)
);
fvVectorMatrix& UEqn = tUEqn.ref();

UEqn.relax();

fvOptions.constrain(UEqn);

if (simple.momentumPredictor())
{
    solve
    (
        UEqn
```

novin-eng.ir

```
            ==
              fvc::reconstruct
              (
                  (
                    - ghf*fvc::snGrad(rhok)
                    - fvc::snGrad(p_rgh)
                  )*mesh.magSf()
              )
        );

        fvOptions.correct(U);
    }
```

معادله مومنتوم که در UEqn.H حل شده است به صورت زیر است:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla(\rho \mathbf{u}\mathbf{u}) + \nabla.(\nu_{eff}\nabla \mathbf{u}) + \nabla.\left(\nu_{eff}(\nabla \mathbf{u})^T - \nu_{eff}\frac{2}{3}tr(\nabla \mathbf{u})^T I\right) = -(\nabla\rho)g.h.f - \nabla p_{rgh}$$

در اینجا u بیان کننده بردار سرعت،ترم $(\nabla\rho)g\cdot h\cdot f$ در سمت راست بیانگر نیروهای بدنی است که بر روی المان سیال عمل می کند و $\nu_{eff} = \nu + \nu_t$ است.

با مقایسه کردن معادله ۲-۲ با UEqn.H، واضح است که ترم fvm::div(phi,U) ترم کانوکتیو یا جابجایی را بیان میکند و MRF.DDt(U) مشتق زمانی سرعت را بیان میکند. تابع turbulence−>divDevReff(U) در داخل کد به تنش برشی ویسکوز در معادله مومنتوم بر میگردد و به صورت زیر داده میشود:

$$divDevReff(U) = \nabla.(\nu_{eff}\nabla \mathbf{u}) + \nabla.\left(\nu_{eff}(\nabla \mathbf{u})^T - \nu_{eff}\frac{2}{3}tr(\nabla \mathbf{u})^T I\right)$$

ترم های ghf*fvc::snGrad(rhok) و fvc::snGrad(p_rgh) به ترتیب بیان کننده ترم های نیروهای بدنی و گرادیان فشار هستند.

معادله انرژی TEqn.H به صورت زیر حل شده است:

DO NOT COPY (www.novin-eli