

OpenFOAMによる ターボポンプ流れ解析

CFXとの比較

(株) 荏原製作所 汎用水力技術開発室
大淵 真志

内 容

1. はじめに
2. ターボポンプ流れ解析の種類と必要な解析技術
3. OpenFOAMの対応状況
4. TurboMachinery SIG提供のターボ機械向け機能
5. 具体的な解析手順
6. 解析事例とCFXとの比較
7. まとめ

1. はじめに

2006年のANSYS社によるFluent買収によりCFD業界の寡占化が一気に進み、商用CFDコードのライセンス料金は値上がりし続けている。

マルチコア化で性能向上しているハードウェアに対し、ソフトウェアライセンス費用はコア課金のために急激に増大し、開発現場におけるCFDの普及を阻害している。

このような背景から、オープンソースで実用性の高いOpenFOAMが近年注目を集めている。

ここでは、ターボポンプの設計現場でOpenFOAMを用いた流れ解析の現状について報告する。

2. ターボポンプ流れ解析の種類と必要な技術

設計現場で実施するターボポンプ流れ解析の種類には以下の3種がある。

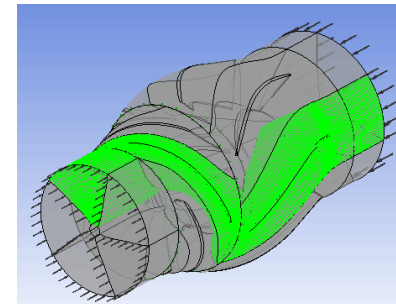
(1) 段落定常解析 (インペラ、ガイドの1ピッチを周期境界で計算する)

要素設計時に形状修正しながら繰り返し計算する場合に実施

メッシュ数が少ないので迅速に計算できる

設計流量付近でのみ有効

必要な解析技術: **General Grid Interface (GGI)**、**Multi-Reference Frame (MRF)**

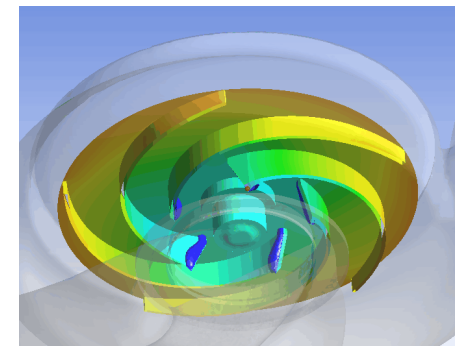


(2) 全機解析 (インペラ全周およびボリュート、ガイドなど静止系全体を扱う)

非設計流量での特性を評価するために実施する

メッシュ数が多く、非定常計算では特に計算負荷が大きい

必要な解析技術: **GGI**、**MRF**、**Sliding Mesh**



(3) 吸込性能解析 (キャビテーションモデルを組み込んだ計算)

ポンプの吸込性能を評価するために行う

吸込圧力を徐々に下げながらキャビテーションの成長による揚程低下を評価する

ターボポンプ流れ解析としては最も高度な解析

必要な解析技術: **Cavitation Model**

3. OpenFOAMの対応状況

1. Multi-Reference Frame . . . ◎

Ver1.4以降で非圧縮ソルバMRFSimpleFoamが提供されている

2. General Grid Interface . . . ○

OpenFOAM-1.5-dev版で提供されている

GGI, cyclic GGI, partial overlap GGI (mixing planeは未対応)

制約事項: overlap GGIはパッチのエッジを揃える必要がある
並列化性能があまり良くない(8並列程度まで)

3. Sliding Mesh . . . ○

OpenFOAM-1.5-devにmixerGgiFvMeshが提供されている

正規版では、mixerFvMesh (GGI非対応) が提供されている

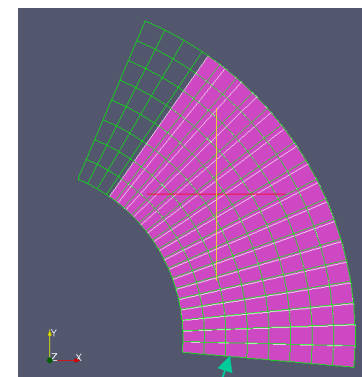
対応ソルバは、turboDyMFoamがある

制約事項: mixerGgiFvMeshは単一口ター領域のみ扱える →turboFvMesh(OpenFOAM-1.6-ext)

4. Cavitation Model . . . △

幾つかのCavitationモデルと、それを利用したソルバcavitatingFoamが提供されているが、MRFやDynamic Meshとの組合せて利用する標準ソルバがない

Overlap GGI



Start Edgeを揃える

3. OpenFOAMの対応状況

(1) 段落定常解析

○ インペラ単独

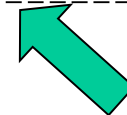
△ インペラ+ガイド (overlapGgiの適用性に依存)

(2) 全機解析

○ 定常、非定常ともに実用段階

(3) 吸込性能解析

× 標準ソルバなし



今回テスト実施

4. TurboMachinery SIG提供のターボ機械向け機能

turboPerformanceライブラリ

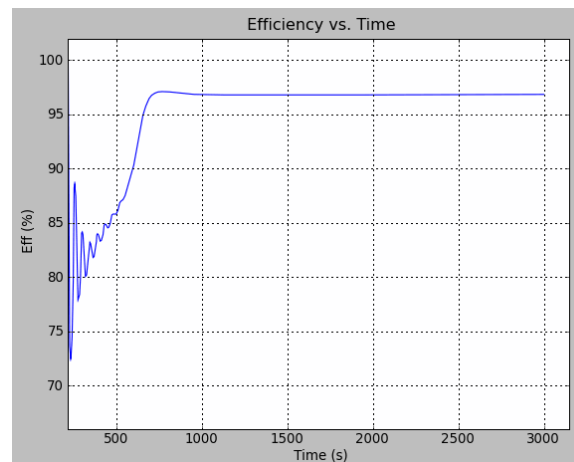
system/controlDictに下記の様なfunctionを定義することで、ターボ機械の特性
各計算ステップごとに出力する

```
functions
(
  turboPerformance
  {
    type turboPerformance;
    functionObjectLibs ("libturboPerformance.so");
    inletPatches (inflow);      入口境界パッチ
    outletPatches (outflow);    出口境界パッチ
    patches ( vane hub shroud ); インペラ壁面パッチ
    log true;
    rhoInf 998.0;               流体密度
    CofR (0 0 0);              回転軸
    omega (0 0 -307.3525);     角速度
  }
)
```

system/controlDict

turboPerformance.dat

#	Time	Head (m)	Axial Power (W)	Efficiency (%)	F_x	F_y	F_z
...							
597		63.625706	15154.741	90.179109	0.12569272	-0.030168457	-2983.2644
598		63.672381	15152.326	90.259651	0.12548831	-0.029208823	-2983.5379
599		63.719201	15149.835	90.340869	0.12506798	-0.028188633	-2983.7887
600		63.766278	15147.323	90.422614	0.1244332	-0.027167927	-2984.0229
601		63.81373	15144.838	90.504748	0.12359115	-0.026204495	-2984.2467
602		63.861667	15142.43	90.587139	0.12257444	-0.025362883	-2984.4659
...							



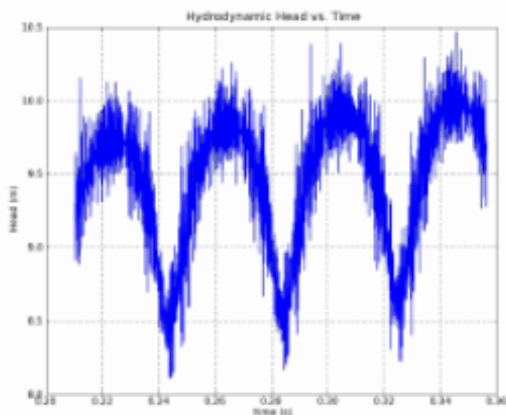
plotTurbo.pyで表示
した揚程の履歴

4. TurboMachinery SIG提供のターボ機械向け機能

transientSimpleDyMFoamソルバ

turbDyMFoamソルバはPISOベースでsliding Mesh解析を行う際に、1stepあたりの回転角度を0.5度以内に小さくとらないと数値振動が発生する

transientSimpleDyMFoamは非定常版のSIMPLEソルバであり、turbDyMFoamよりも大きな回転角度（2度程度）でも安定に計算することができる



turbDyMFoamによる不自然な数値振動の例

turbDyMFoam

Time Loop :

p.storePrevIter();

runTime++;

mesh.moving();

PISO Outer Loop:

predictor -U eq

PISO Inner Loop :

rAU = 1.0/UEqn.A();

U = rAU*UEqn.H();

phi = (fvc::interpolate(U) & mesh.Sf());

adjustPhi(phi, U, p);

non Orthogonal Correction :

p.relax();

fvc::makeRelative(phi, U);

U -= rAU*fvc::grad(p);

U.correctBoundaryConditions();

turbulence->correct();

transientSimpleDyMFoam

Time Loop :

runTime++;

bool meshChanged = mesh.update();

SIMPLE loop:

predictor - U eq

rAU = 1.0/UEqn.A();

U = rAU*UEqn.H();

phi = (fvc::interpolate(U) & mesh.Sf());

adjustPhi(phi, U, p);

p.storePrevIter(); Loopの最新値を使用

non Orthogonal Correction :

p.relax();

fvc::makeRelative(phi, U);

U -= rAU*fvc::grad(p);

U.correctBoundaryConditions();

turbulence->correct();

5. 具体的な解析手順

①メッシュの作成 (商用メッシャを利用)

ICEM CFDやTurboGridなど

②メッシュの変換

fluentMeshToFoamなど

③GGIの設定

constant/polyMesh/faceZone

④境界パッチの設定

constant/polyMesh/boundary

⑤回転領域の設定

constant/MRFZones, constant/dynamicMeshDict

⑥初期、境界条件の設定

0/U, p, k, epsilon

⑦計算方法、条件の設定

system/controlDict, fvScheme, fvSolution

⑧物性値、乱流パラメータ設定

constant/transportProperties, RASProperties

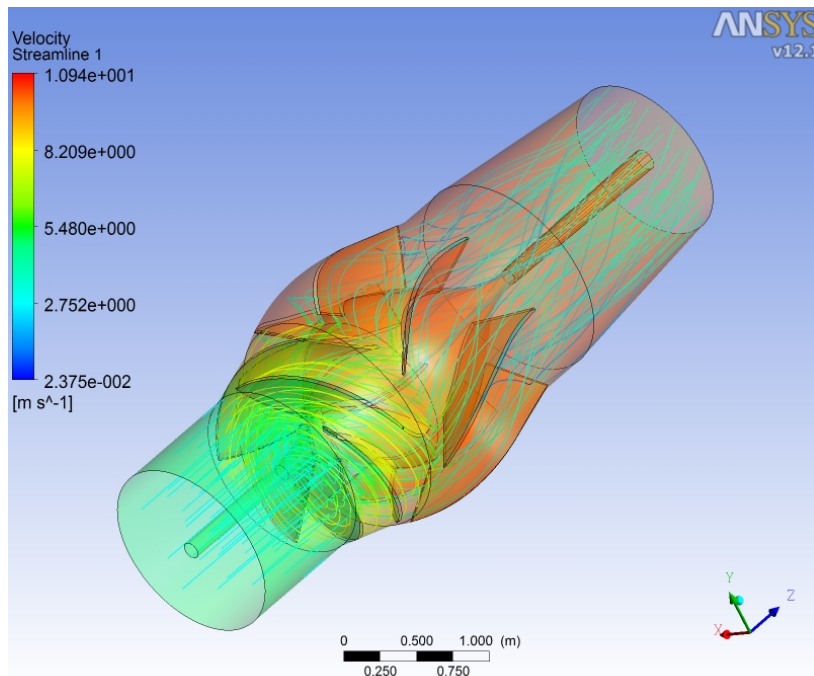
⑨計算の実行

⑩後処理

Paraviewまたはデータ変換してFieldViewなどの商用ポストを利用

6. 解析事例とCFXとの比較

1) 斜流ポンプの全体解析 (MRFSimpleFoam)



ノード数 : 467,232、セル数 : 1,611,312

<MRFSimpleFoam>

移流項 : limitedLinear 1

乱流モデル : k- Ω SST

メモリ : 1.3GB (倍精度)

計算時間 : 3000step

12時間 (Opteron2220 4並列, GAMG)

28時間 (Opteron2220 4並列, PCG)

<CFX>

移流項 : Blended Factor 0.75

乱流モデル : k- Ω SST

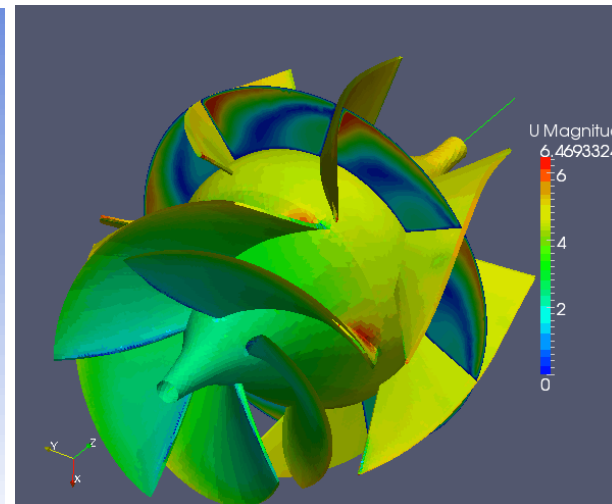
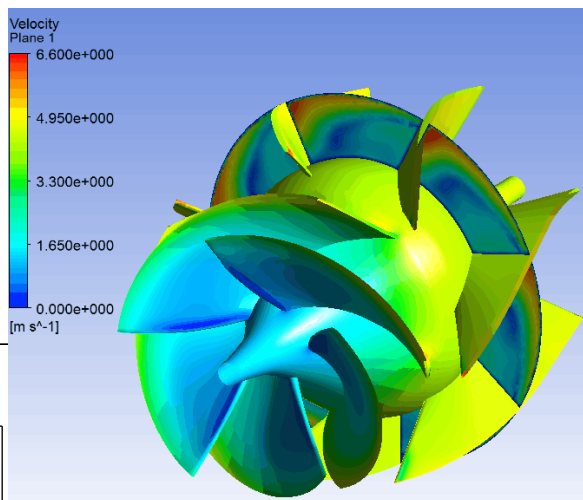
メモリ : 960MB (単精度)

計算時間 : 429stepで収束

4.95時間 (Core2Quad 9650)

6. 解析事例とCFXとの比較

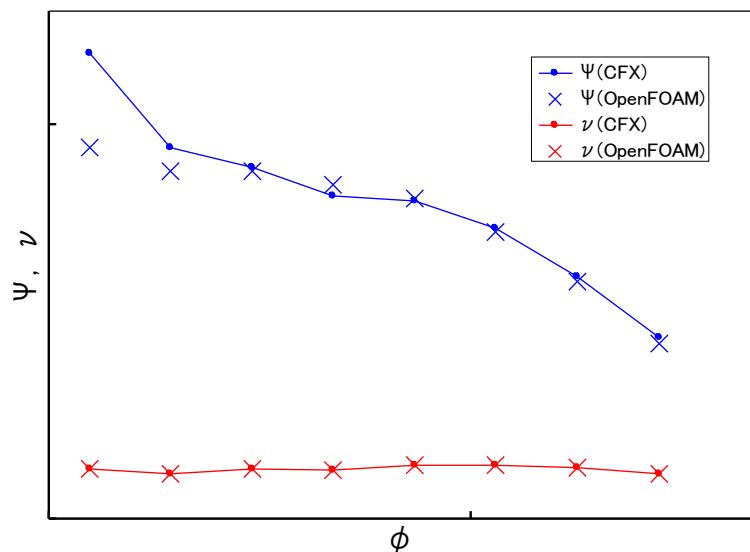
1) 斜流ポンプの全体解析 (MRF SimpleFoam)



CFX

OpenFOAM

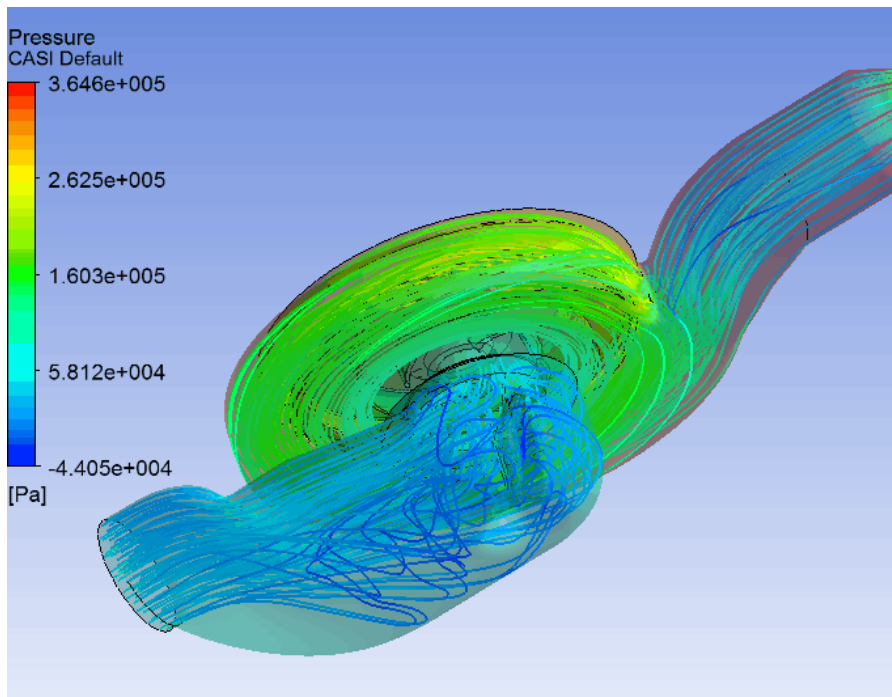
CFX vs. OpenFOAM (斜流ポンプ)



締切り付近を除き、良好に一致
CFXの結果は流れを考慮して補正すると実測
と良く一致するため、OpenFOAMも
精度的に十分実用的であると言える

6. 解析事例とCFXとの比較

2) ラインポンプの全体解析 (MRFSimpleFoam)



ノード数 : 584,717、セル数 : 2,203,907

<MRFSimpleFoam>

移流項 : limitedLinear 1

乱流モデル : k- Ω SST

メモリ : 1.9GB (倍精度)

計算時間 : 3000step

16時間 (Opteron2220 4並列, GAMG)

36時間 (Opteron2220 4並列, PCG)

<CFX>

移流項 : Blended Factor 0.75

乱流モデル : k- Ω SST

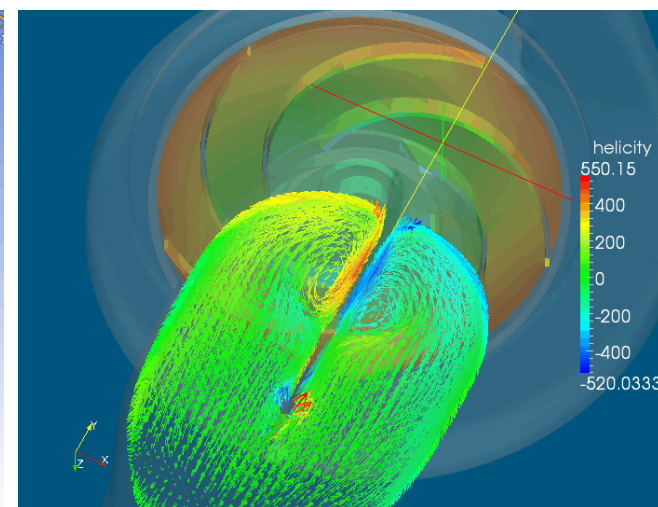
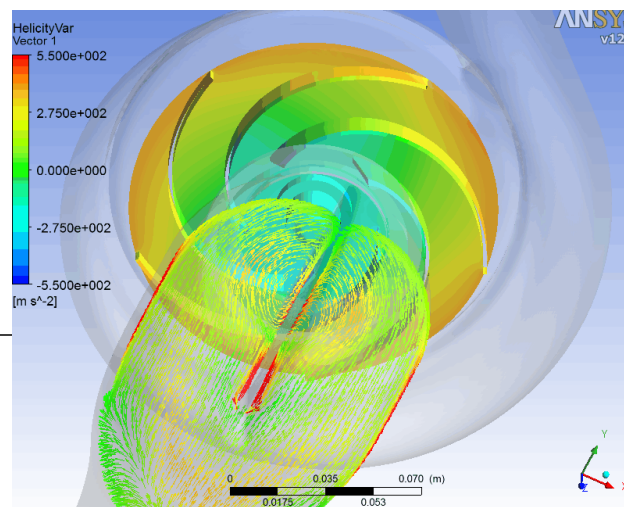
メモリ : 1.2GB (単精度)

計算時間 : 1000stepでほぼ収束

13時間 (Core2Quad 9650)

6. 解析事例とCFXとの比較

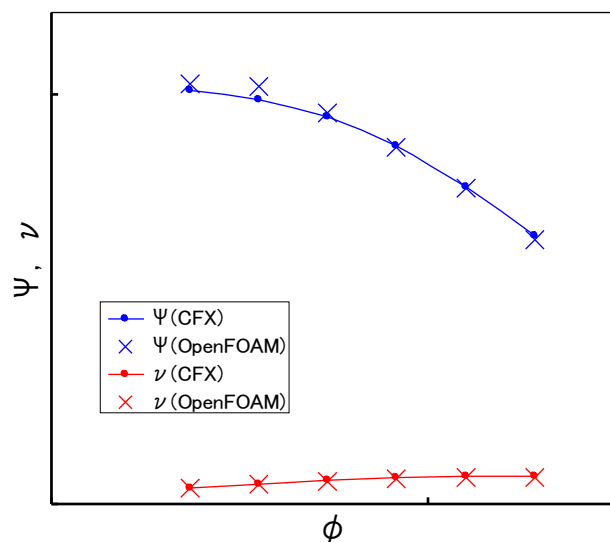
2) ラインポンプの全体解析 (MRFSimpleFoam)



CFX

OpenFOAM

CFX vs. OpenFOAM (Line Pump)

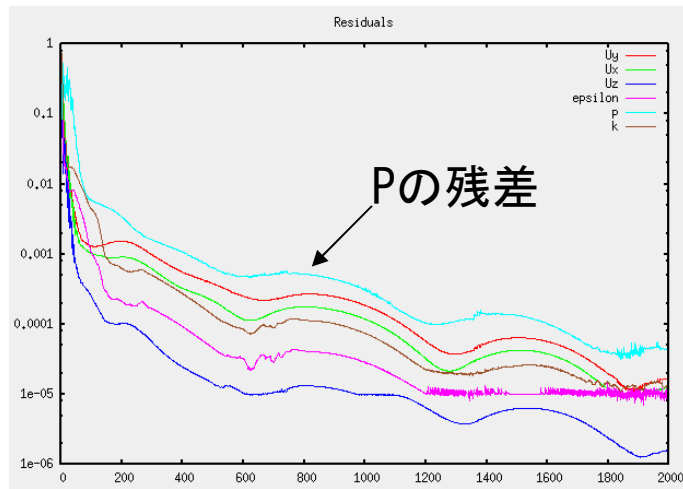


この場合でも、細かな二次流れパターンを除いて、OpenFOAMとCFXは概ね良好に一致

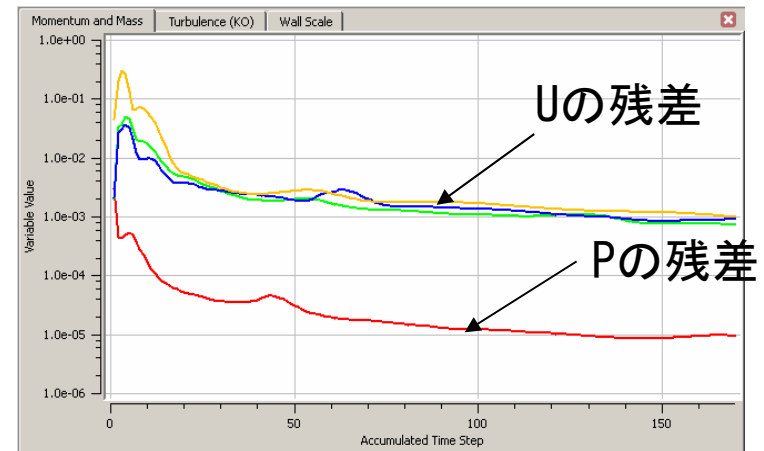
6. 解析事例とCFXとの比較

<CFXの特徴>

- ・ FEMベースのFVMソルバ（内挿関数）
二次精度スキームが標準
- ・ 非圧縮－圧縮を単一Coupledソルバ
ロバストかつ収束が早い（圧力の残差が速度より数桁小さく、少ない反復数で収束）
- ・ TASCFlowの特徴を引継ぎ、ターボ機械向け機能が豊富
- ・ 最新のGUI（Workbench）によるツール連携
殆ど標準設定で良いため操作が簡単
- ・ 高品位グラフィックスによるポスト処理



simpleFoam



CFX

6. 解析事例とCFXとの比較

<CFXと比較してOpenFOAMに不足している点>

・段落解析機能

overlap GGIやmixing Planeが不完全なため、段落解析の実用性が低い

ミュンヘン大学のOliver Boamらが独自のoverlapGgiを開発中 →OpenFOAM-1.6-ext

カナダHydro-QuebecのM. Beaudoinらがmixing Planeを開発中

・Coupledソルバ

SIMPLE系ソルバは収束が遅い

スイス連邦工科大学ローザンヌ校のL. MangamiらがCoupledソルバを開発中

・Pre/Post機能

使いやすいメッシュがない

blockMesh, snappyHexMeshではターボ機械向けには役不足

ターボ機械向けのポスト処理機能がない

ParaViewでターボ機械の後処理をするのは大変

条件設定や後処理の手間がかかり過ぎる

設計者向けのツールには必須

7. まとめ

OpenFOAMによるターボポンプ流れ解析への対応状況と具体的手順を説明し、適用事例として斜流ポンプとラインポンプの全機定常解析を実行し、CFXと比較した。

ターボポンプの解析には正規版ではなく、コミュニティにより開発されているOpenFOAM-dev版が適していることが解った。

OpenFOAMの解析結果はCFXと良好に一致し、実用段階に達していることが確認できたが、ターボ機械への適用性の高いCFXと比較すると、収束性や計算速度、条件設定や後処理の作業性などで、まだ大きな差があることが確認できた。

つい先日多くの新機能を備えた新バージョンOpenFOAM-1.6-extがリリースされた。その内容評価を含め、引き続き開発動向を注視していく予定である。