

新TreeFoamの紹介

(TreeFoam ver 3.21.231130)

<変更内容>

1. 連成計算の高速化

FrontISTRの計算をProcess並列が可能になるよう修正。

(OpenFOAM、FrontISTRともprocess並列が可能)

FrontISTRのprocess並列について

FrontISTRの計算はthread並列を前提にしていたが、今回process並列（MPI並列）に対応させた。

<流体-構造の連成解析Dialog>

OpenFOAMの設定

連成するfield, patchを選択

p fieldの係数: 1.0

p fieldのshift量: -1e5

scalarField: p,T

pointField: pointDisplacement

patch名

- lower
- right
- left
- frontBack
- walls

並列計算する

FrontISTRの設定

連成するsolidDir, SGRPを選択

command「easyistr」「fistr1」のpath設定

easyistr: /home/caeuser/easyIstr 参照...

fistr1: /usr/bin/fistr1 参照...

solidDir: ../solid3 参照...

SGRP名

- pressWall
- otherS

並列計算する 全core数: 8

連成計算するタイミング (deltaTの倍数:step数を設定)

流体-構造連成計算用のstep数: nStepsFsi: 2

熱移動、熱ひずみ計算用のstep数: nStepsHeat: 10

各Jobの待ち時間

最大待ち時間: 60 sec

結果クリア 設定取得・保存 連成計算開始 Fistrのlog

EasyISTR起動 流体 folder開く 固体 folder開く 閉じる

process並列計算の為には、

「並列計算する」をチェックする。

FrontISTR側は、さらに「全core数」を設定する。



並列計算の為には、予め並列計算用にメッシュ分割しておく。

連成計算開始時に、メッシュ分割数を確認して、分割数分のprocess並列計算を開始する。

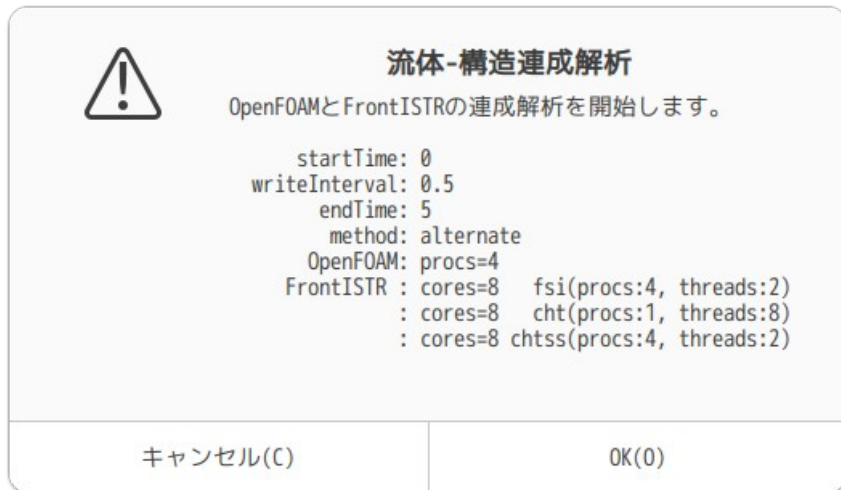
例) 全core数:8、メッシュを4並列で分割の場合
process並列数:4、thread並列数:2、合計8コア
で計算する。

FrontISTRのメッシュ分割について

FrontISTR側は、解析内容に応じたmshファイルを設定している。

解析内容	mshファイル名
fsi：流体-構造連成	FistrModel.msh
cht：流体-固体の熱連成	FistrModel_cht.msh
chtss：固体の熱ひずみ	FistrModel.msh（fsiと同じメッシュを使用）

並列計算の為には、EasyISTR上で、fsi,chtのメッシュを並列計算用に分割しておく必要がある。



連成計算開始時のdialog

全core数:8、メッシュ分割は、

fsi, chtss用のFistrModel.mshを4分割

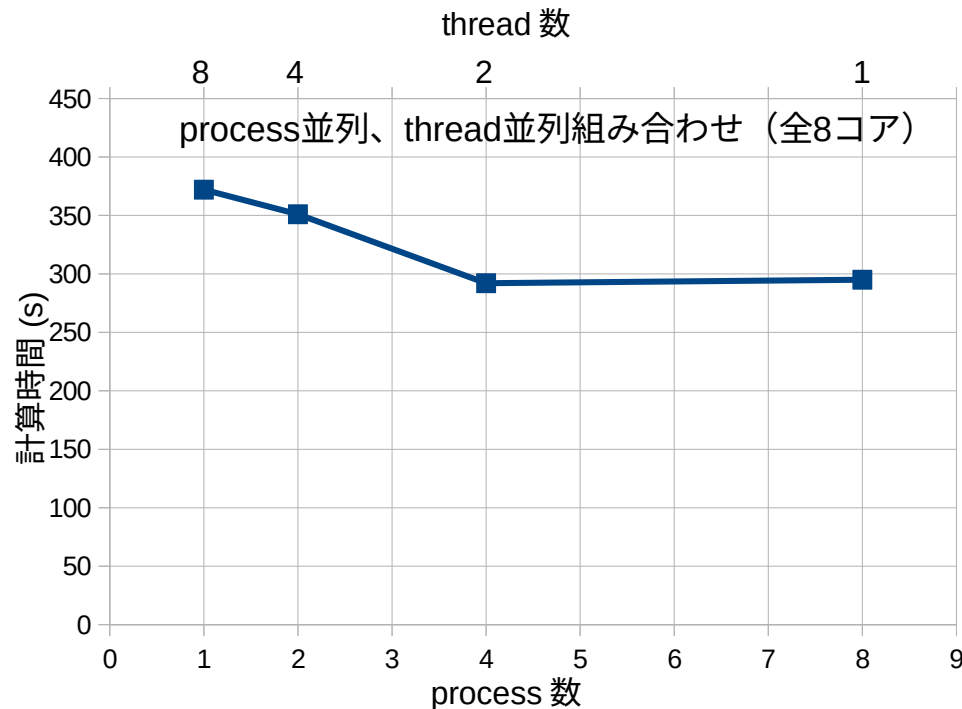
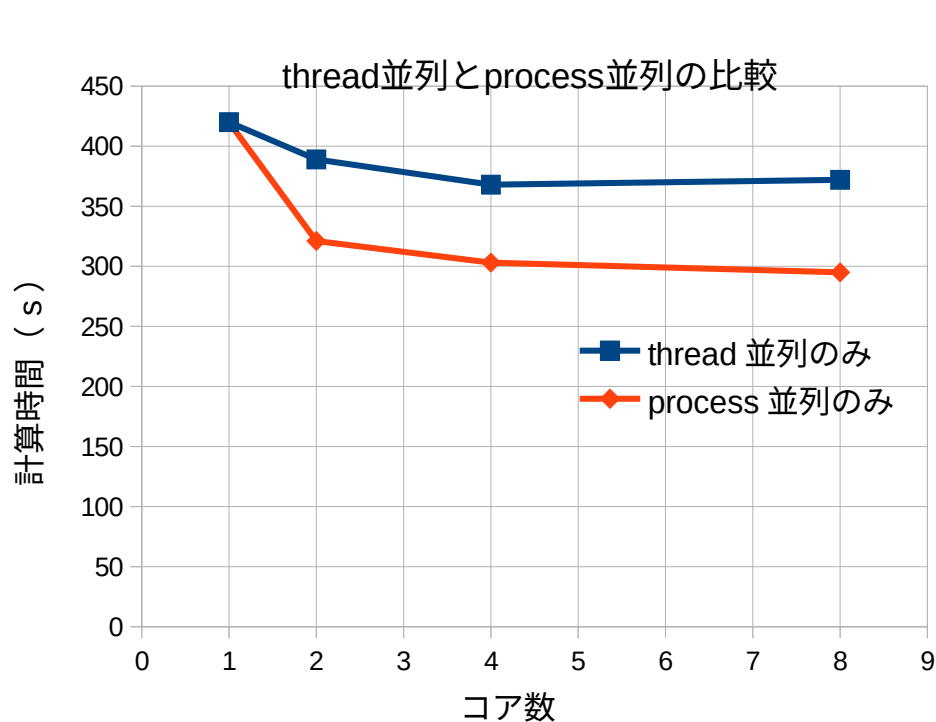
cht用のFstrModel_cht.mshを分割せず

の内容で連成解析を開始した場合。

chtは、8 threadで計算する。

process並列の速度比較

tutorialsの「softPipe_OpenFOAM-FrontISTR」の交互計算で確認。OF側は、1コアで計算。



process並列の方がthread並列よりも早い。