TreeFoamによる OF ⇔Fistr 連成解析について

Ver 3.20からOF、Fistrの連成計算に関し、
　　OF, Fistr同時計算（高速）に加え、交互計算（安定）を追加。

現在、さらに高速化を図るため、Fistrの計算を
　　thread並列に加え、process並列（MPI並列）を追加。

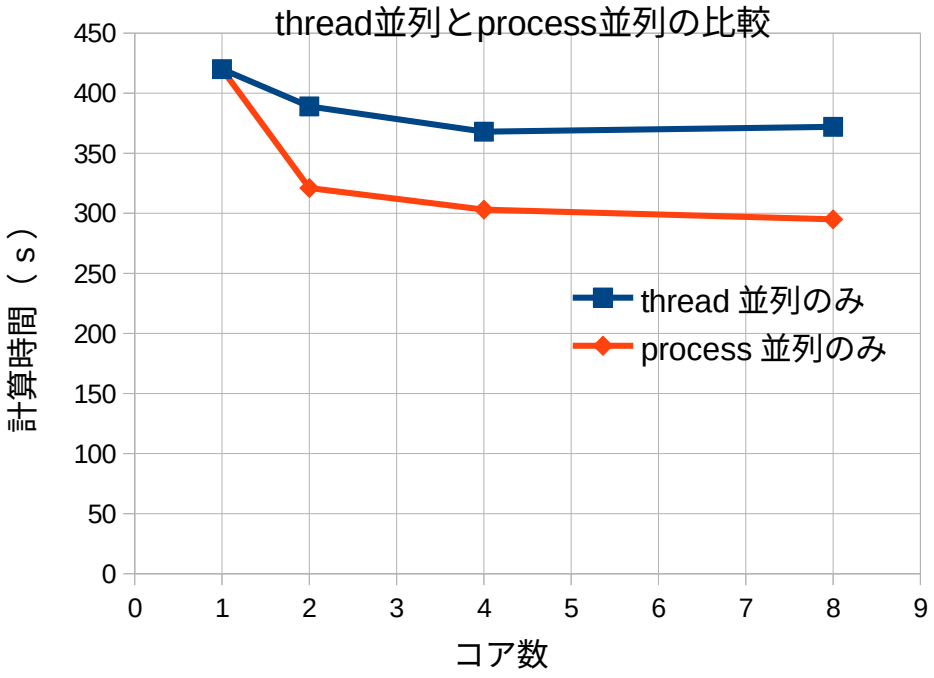
23/11/25　藤井

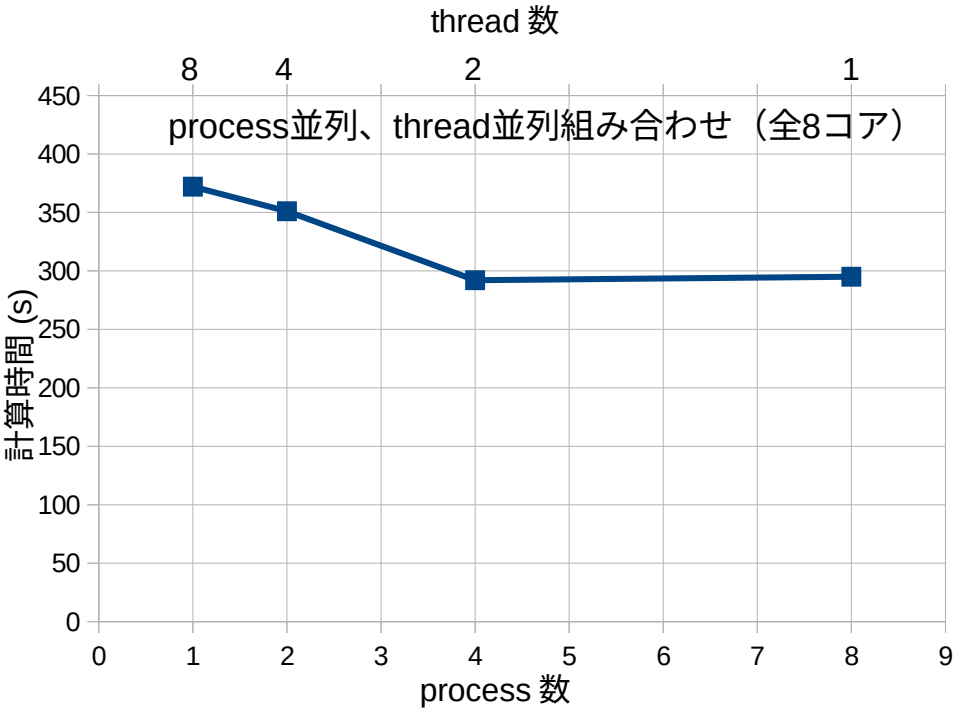
1.　Fistrのprocess並列の方法

1)　process並列の為には、予め並列数分メッシュ分割しておく。
2)　使用する全core数を指定して、計算を開始する。
　全8coreを指定して、4並列でメッシュ分割した場合は、
　　process数:4　thread数:2　で計算を開始する。
3)　連成計算の為に、分散しているFistrの計算結果を、連成面のみmergeしている。
　（mergeは、全processで担当させている。）

2.　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。