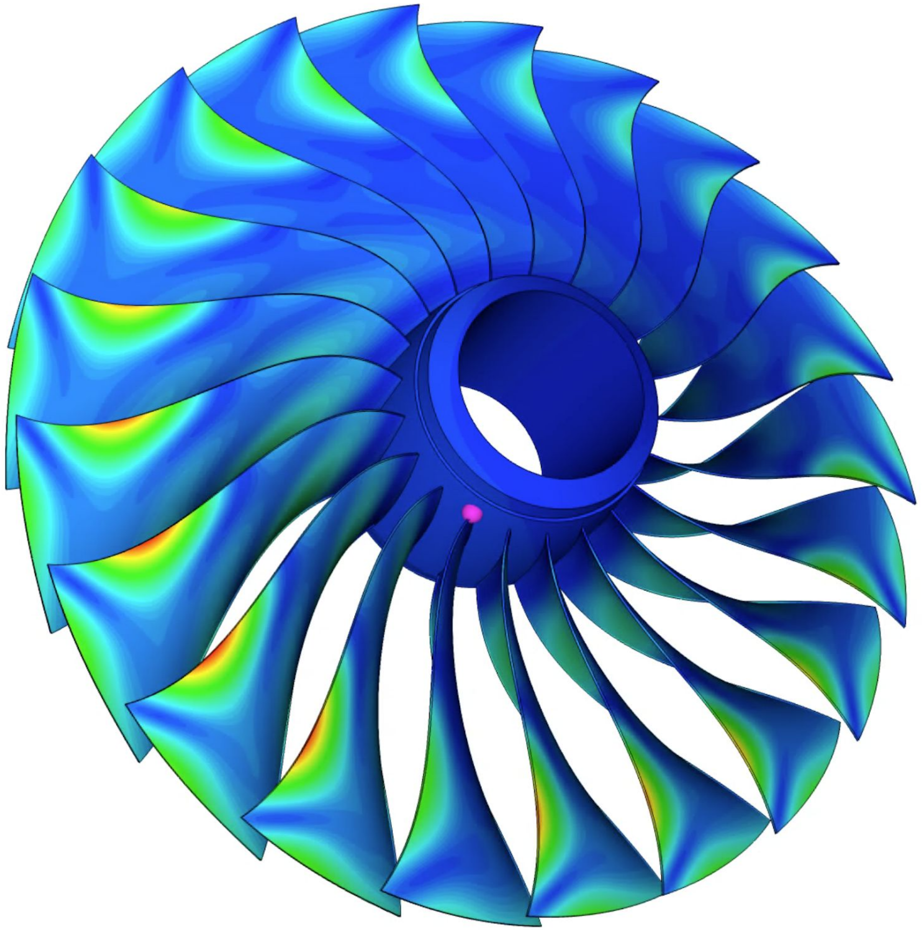
GPU accelerated computing for Finite Element Method



Introduction to GPU

GPU vs CPU

Matrix – Vector Multiplication

- GPUs for calculations involving dense matrices
- For example:
  - ❑ Compute explicit inverse which is full matrix
  - ❑ For different righthand side vectors
  - ❑ linear systems with Schur complements in FETI are also full

Introduction to GPU

Iterative solvers

Iterative solver for $Ax = b$

- $\boxed{\text{Conjugate Gradient}}$
- GMRES
- Orthomin
- Orthores
- Orthodir
- Bi Conjugate Gradient

Conjugate Gradient Method

$r_0 = b - Ax_0$
$p_0 = r_0$
$k = 0$
if $r^{\mathrm{T}}r < \text{tol}$ **then**
$\quad \alpha_k = \frac{r_k^{\mathrm{T}} r_k}{p_k^{\mathrm{T}} A p_k}$
$\quad x_{k+1} = x_k + \alpha_k p_k$
$\quad r_{k+1} = r_k + \alpha_k A p_k$
$\quad \beta_k = \frac{r_{k+1}^{\mathrm{T}} r_{k+1}}{p_k^{\mathrm{T}} A p_k}$
$\quad p_{k+1} = r_{k+1} + \beta_k p_k$
$\quad k = k + 1$
end if
return x_{k+1} as the result

- ❑ One Matrix - Vector Product
- ❑ Two vector dot product per iteration
- ❑ Four vectors of working stage

GPU vs CPU

One matrix vector multiplication in each iteration.

Matrix – Vector Multiplication

$$A_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix} \qquad x_n = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

Sequential algorithm for Matrix - Vector product

1: **for** $i = 1, 2, ..., m$ **do**
2:     out[i] = 0
3:     **for** $j = 1, 2, ..., n$ **do**
4:         out[i]+ = mat[i][j]* x[j]
5:     **end for**
6: **end for**

$$Ax = \begin{pmatrix} a_{1,1}x_1 + a_{1,2}x_2 + \cdots & a_{1,n}x_n \\ a_{2,1}x_1 + a_{2,2}x_2 + \cdots & a_{2,n}x_n \\ \vdots & \\ a_{m,1}x_1 + a_{m,2}x_2 + \cdots & a_{m,n}x_n \end{pmatrix}$$

Introduction to GPU

GPU programming syntax

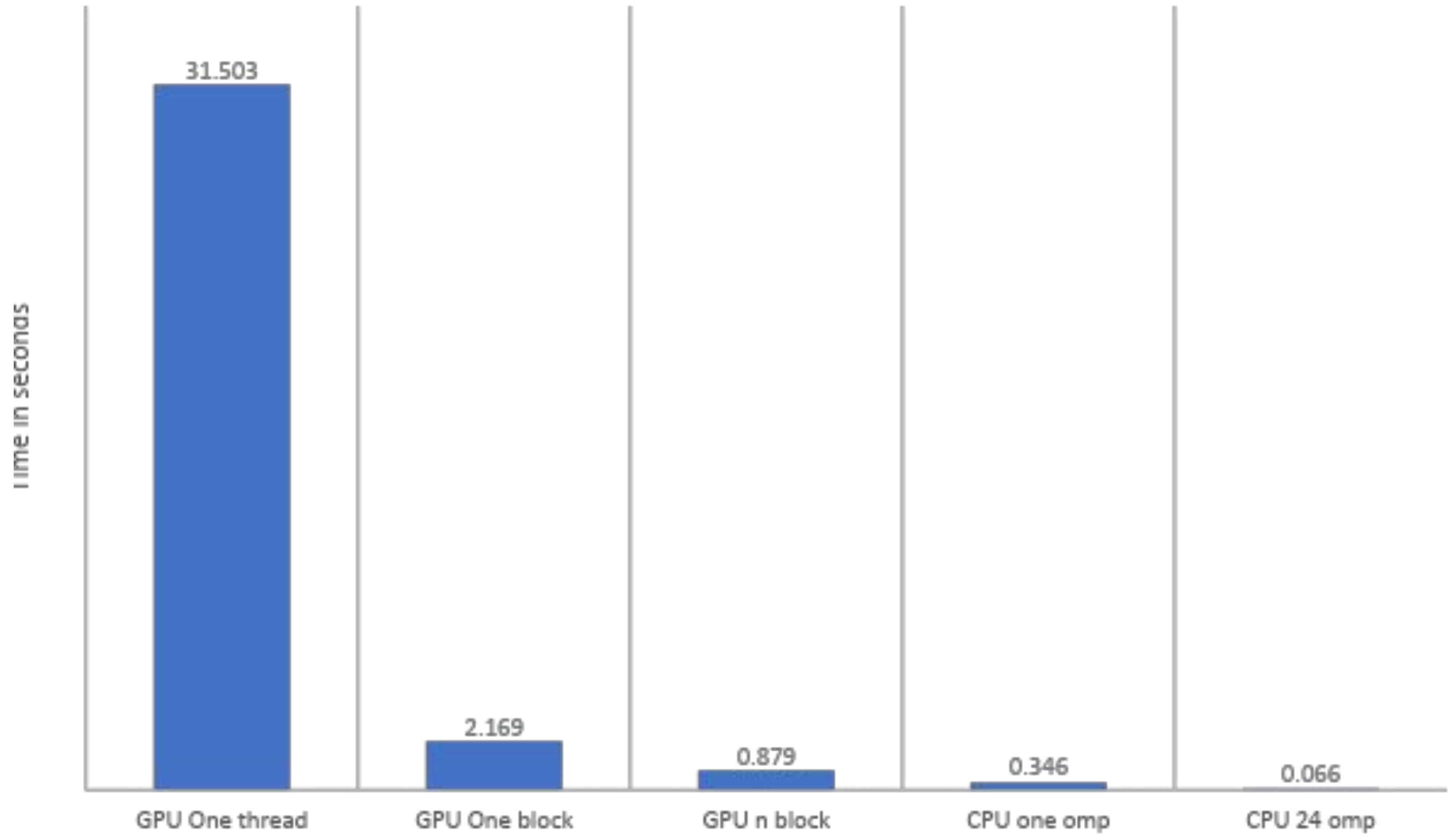
EXERCISE 2 : Matrix Vector product using GPU program

Source code:
https://github.com/sivasanarul/FEMwithGPU/tree/master/EX2_matrixvectmul

Introduction to GPU

Introduction of GPU programming

Vector Addition Example : vector_addition()



Introduction to GPU

GPU programming syntax

EXERCISE 3 : Matrix Vector product using GPU program

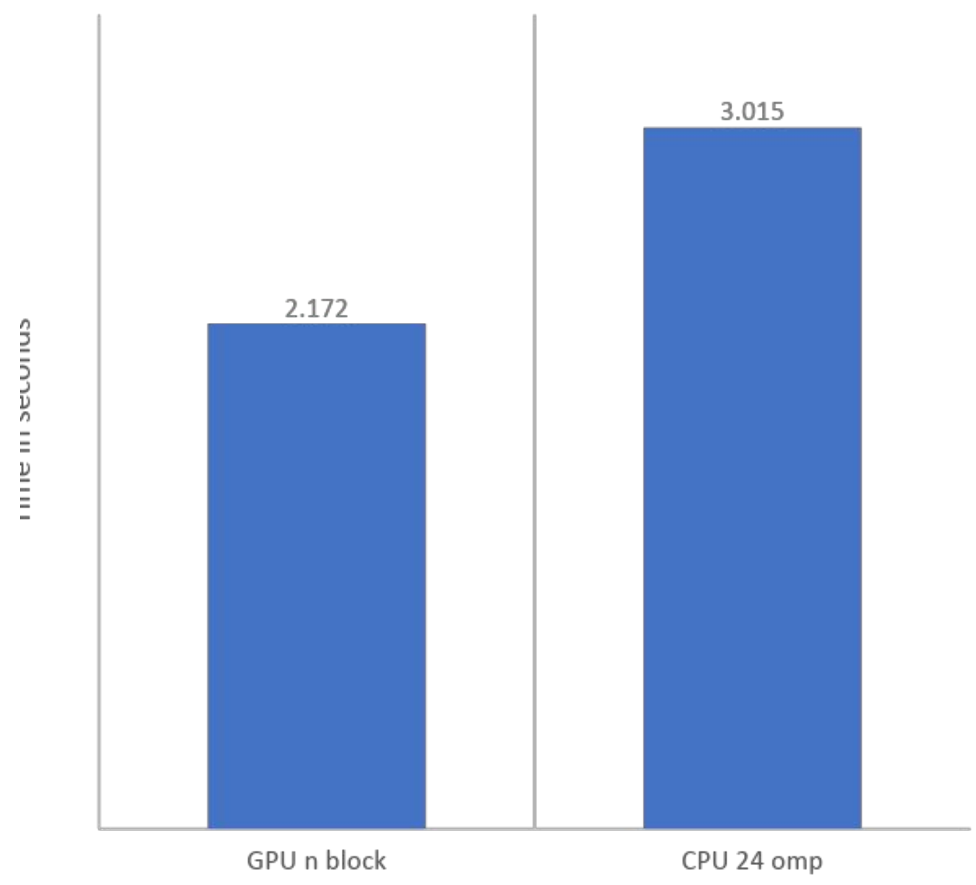
Source code:
https://github.com/sivasanarul/FEMwithGPU/tree/master/EX3_repmatrixvectmul

Introduction to GPU

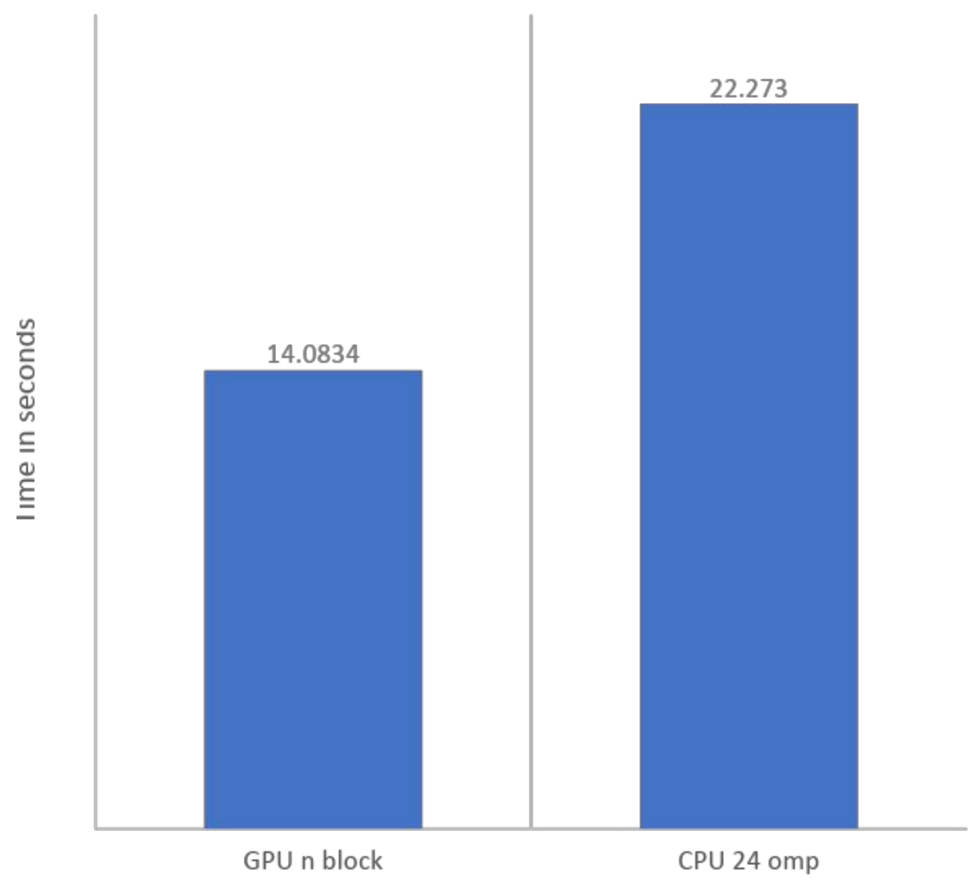
Introduction of GPU programming

Vector Addition Example : matrix_vector_product()

100 iterations



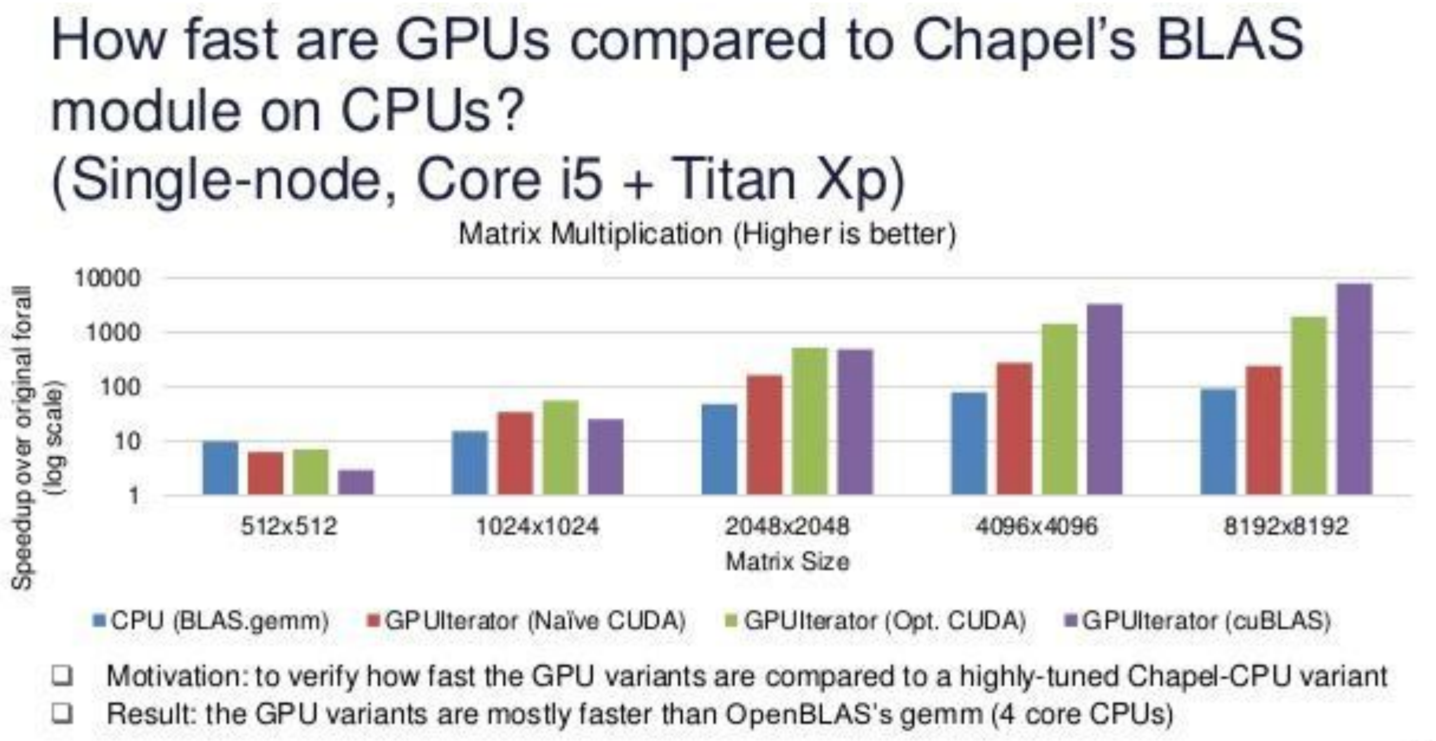
1000 iterations



Introduction to GPU

Introduction of GPU

Use library and compiler directives instead of programming.



The ACM SIGPLAN 6th Annual Chapel Implementers and Users Workshop (CHIUW2019) co-located with PLDI 2019 / ACM FCRC 2019.
https://www.slideshare.net/ahayashi10/gpuiterator-bridging-the-gap-between-chapel-and-gpu-platforms

Introduction to GPU

GPU programming syntax

EXERCISE 4 : Matrix Vector product using GPU CUBLAS program

Source code:
https://github.com/sivasanarul/FEMwithGPU/tree/master/EX4_matrixvectmul_blas

https://developer.nvidia.com/sites/default/files/akamai/cuda/files/Misc/mygpu.pdf