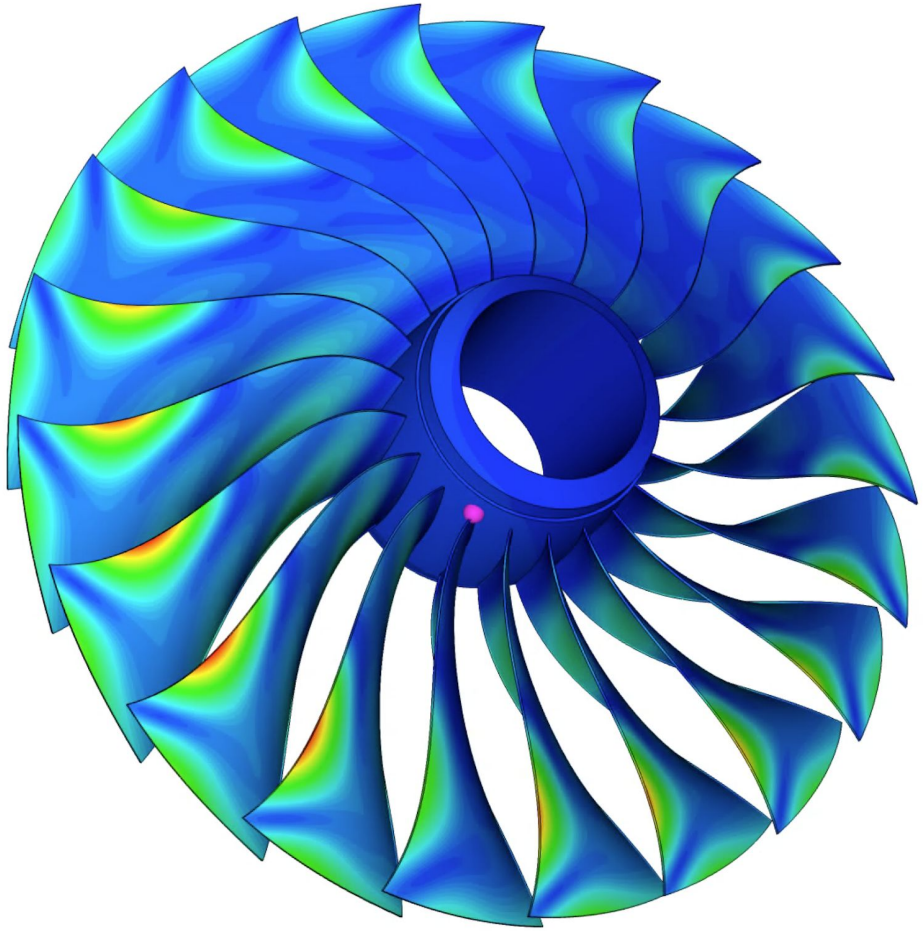
GPU accelerated computing for Finite Element Method



Introduction to GPU

Introduction to GPUs

Objectives

- Ways to utilize GPU

Introduction to GPU

Introduction to GPUs

Several libraries has GPU acceleration

Linear Algebra : NViDIA cuFFT, NViDIA cuBLAS, NViDIA cuSPARSE, MAGMA…

Statistics : NVIDIA Math Lib, ArrayFire…

Features:

- In depth knowledge of GPU programming is not needed
- The libraries follow standard APIs therefore can used in existing code with minor modifications
- High quality and suitable for variety of application

Introduction to GPU

Introduction to GPUs

Compiler Directives

For C, C++, Fortran

Directives could be used similar to OPENMP

For Example: #pragma acc parallel loop …

Features:

- Compiler does parallelism management and data movement
- Different compiler versions give different performance

Introduction to GPU

Introduction to GPUs

Programming Languages

CUDA C, CUDA C++, PyCUDA, LabVIEW…

Features:

- Good control of parallelism and data movement
- Can be used for any type of computation

```
39      // Main function
40      int block_size = 256;
41      int grid_size  = (N+block_size)/block_size;
42      vector_add<<<grid_size,block_size>>>(d_out, d_a, d_b, N);
```

```
10  __global__ void vector_add(float *out, float *a, float *b, int n){
11      int index = blockIdx.x *blockDim.x + threadIdx.x;
12      if (index<n){
13      out[index] = a[index] + b[index];}
14  }
```