

Accelerating and Parallelizing MATLAB Code

Training Objectives

This one-day course covers a variety of techniques for making your MATLAB® code run faster. You will identify and remove computational bottle-necks using techniques like preallocation and vectorization. In addition, you will take advantage of multiple cores on your computer by parallelizing for-loops with Parallel Computing Toolbox. If you are working with long-running simulations, you will benefit from the hands-on demonstrations and exercises in the course. Topics include:

- Improving performance within core MATLAB
- Parallelizing computations
- GPU computing

Prerequisites

MATLAB Fundamentals, or equivalent experience using MATLAB

Products

- MATLAB
- Parallel Computing Toolbox
- MATLAB Distributed Computing Server
- MATLAB Coder

Course Outline

Improving Performance (2.5 hrs)

Objective: Analyze code performance and utilize techniques for acceleration within MATLAB.

- Identifying bottle necks
- Preallocating arrays
- Vectorizing operations in various ways
- Rewriting algorithms

Parallelizing Computations (2.0 hrs)

Objective: Parallelize code execution to take advantage of multiple cores.

- Opening additional MATLAB processes
- Running parallel for-loops
- Measuring speedup
- Processing multiple files in parallel

Parallel for-Loops (2.0 hrs)

Objective: Explore parallel for-loops in more detail and apply techniques for converting for-loops to parfor-loops.

- Requirements of parallel for-loops
- Parallelizing for-loops
- Retrieving intermediate results

GPU Computing (1.5 hrs)

Objective: Execute MATLAB code on your computer's graphics card (GPU) as another option for speeding up calculations.

- Overview of GPU architecture and processing
- Applications suitable for GPU processing
- Invoking MATLAB functions on the GPU
- Using pre-existing CUDA code