

# Undocumented Matlab

unbelievable features; unbelievable quality; unbelievable cost effectiveness; unbelievable service

## MATLAB PERFORMANCE TUNING – PART 1

### WEBINAR OUTLINE

#### 1. Profiling Matlab performance

- a. When to profile and when not to bother
- b. When should we stop optimizing the code?
- c. Profiling techniques
- d. Real-time profiling limitations
- e. Using the profiler vs. tic/toc
- f. Matlab's JIT and its effect on profiling
- g. Trade-offs: performance vs. maintainability, robustness, development time, repeatability
- h. Vertical vs. horizontal scalability



#### 2. Standard programming techniques

- a. Loop optimizations
- b. Caching data
- c. Smart checks bypass
- d. Exception handling and performance
- e. Sizing data sets
- f. Code inlining
- g. Externally-connected systems
- h. Perceived vs. actual performance
- i. Data analysis techniques
- j. Using mathematical identities

#### 3. Matlab-specific techniques

- a. Using different storage types
- b. Object-orient Matlab and performance
- c. Using internal helper functions
- d. Strings and dates/times
- e. Matlab's Startup "Accelerator"

#### 4. Using binary code

- a. Mex
- b. Matlab Compiler vs. Coder
- c. 3<sup>rd</sup>-party libraries

#### 5. I/O speedup techniques

- a. XLS/CSV read/write
- b. Binary vs. text format
- c. Reducing disk access
- d. Buffered, consolidated and chunked I/O

### Summary

You will learn:

- when to profile, when not to bother
- how to determine the most effective optimization route
- trade-offs of program performance
- how to profile Matlab programs to determine where the hotspots are
- how to avoid and solve potential hotspots in your program's execution
- how to use a variety of techniques to maximize the performance of your application at reasonable cost
- skills that will enable you to discover additional techniques for improving program performance by yourself

### Target audience

Matlab users with some experience using Matlab, who wish to improve the quality and effectiveness of their programs;  
Matlab users who wish to distribute professional Matlab-based software where timely run-time performance is important.

Basic familiarity with Matlab environment, data types and coding is assumed.