Parallel Computing & Big Data with MATLAB

Lyamine Hedjazi  Application Engineer
Sophie Louage  Account manager
Parallel Computing can help you to...

- **Reduce computation times**
  - Accelerate your simulations
  - By taking advantage of unused CPU cores and/or GPUs

- **Run more simulations** within the same timeframe
  - Test more values for parameter calibration and design optimization
  - 2x more runs in Monte Carlo simulations ⇒ 41% increase in accuracy

- **Run larger simulations** within the same timeframe
  - Use finer meshes for Finite Element Analysis
  - Not enough memory on a single computer?
    ⇒ Distribute the mesh on a cluster!
How much faster can my computations run?

- For an application that is well-suited for parallel computing...
  - Monte Carlo simulations
  - Parameter calibration / Design optimization
  - Linear algebra on very large matrices
  - Processing of large images and/or large collection of images

- ... the speed-up mainly depends on the **number of available CPU cores**

<table>
<thead>
<tr>
<th>Laptop</th>
<th>Workstation</th>
<th>HPC cluster (multiple computers)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.5x to 3x</td>
<td>1.5x to 20x</td>
<td>10x to 1000x or even more!</td>
</tr>
</tbody>
</table>
MathWorks Solutions for Parallel Computing

Parallel Computing Toolbox

- MATLAB Toolbox
- Allows to write parallel applications by giving access to parallel programming constructs
- Take advantage of local CPU cores and GPUs

MATLAB Distributed Computing Server

- Standalone product
- Run computations on remote resources
- With MDCS, a single application can be executed on multiple computers

Laptop
1.5x to 3x

Workstation
1.5x to 20x

HPC cluster (multiple computers)
10x to 1000x or even more!
Scale Computer Power
Parallel Computing tools with MATLAB

Parallel Computing Toolbox
MATLAB Distributed Computing Server (MDCS)

Cloud
and
Scaling-up locally

Use the full processing power of desktops (CPUs, GPUs)

- Leverage GPU accelerator
  
  ```
  d=gpuDevice()
  ```

- Leverage Multicore CPU
  
  ```
  p=parpool('local');
  ```
Programming Parallel Applications
Leverage GPU Accelerators

- Built-in GPU support
  - Neural Network Toolbox
  - Many signal & image processing features

- GPU-accelerated functions
  - Transfer data to the GPU and run computations on the GPU
  - Overloaded functions, no need to rewrite your code
  - Most linear algebra functions, FFT, statistics...

- Import CUDA code into MATLAB
  - If you have written C/C++/FORTRAN code that uses the CUDA library, you can import it in MATLAB using MEX files

- Important remarks
  - Only NVIDIA GPUs are supported
  - Make sure to use GPUs with full double precision support (NVIDIA Tesla)
Leveraging your parallel computing resources

*Multicore CPUs and systems*

- **Built-in Parallel Computing support**
  - Many toolboxes can benefit from parallel speed-up
    - Optimization & Global Optimization Toolboxes
    - Statistics and Machine Learning Toolbox
    - Neural Network Toolbox
    - Image Processing Toolbox & Computer Vision System Toolbox
    - ...
  - Check the documentation for a `UseParallel` property
  - Or refer to [www.mathworks.com/builtin-parallel-support](http://www.mathworks.com/builtin-parallel-support)

- **`parfor` (parallel-for loops)**
  - Speed up for loops with `independent` iterations
  - Monte Carlo simulations, parameter calibration
Parallel-for Loops

- Parallel-for loops used to **speed-up for loops** with **independent** iterations
  - Parameter sweeps
  - Monte Carlo simulations

- **Very easy to set up**: just replace `for` with `parfor`

- Ideal use cases for parallel computing
  - Low overhead, no communication between tasks
  - Excellent scalability and speed-ups
Leveraging your parallel computing resources

*Multicore CPUs and systems*

- **Built-in Parallel Computing support**
  - Many toolboxes can benefit from parallel speed-up
    - Optimization & Global Optimization Toolboxes
    - Statistics and Machine Learning Toolbox
    - Neural Network Toolbox
    - Image Processing Toolbox & Computer Vision System Toolbox
    - ...
  - Check the documentation for a `UseParallel` property
  - Or refer to [www.mathworks.com/builtin-parallel-support](http://www.mathworks.com/builtin-parallel-support)

- **parfor (parallel-for loops)**
  - Speed up for loops with `independent` iterations
  - Monte Carlo simulations, parameter calibration

- **MPI-like programming constructs**
  - `spmd` blocks
  - `labSend`, `labReceive`, `gop`
**spmd blocks**

```
spmd
  % single program across workers
end
```

- Mix parallel and serial code in the same function
- Run on a pool of MATLAB resources
- **Single** Program runs simultaneously across workers
- **Multiple** Data spread across multiple workers
Independent tasks- Monte Carlo Simulations in Parallel

N-body problem

- **Formal problem description**
  - 100 balls are placed randomly in a 2-D domain
  - Initial velocity is set randomly (speed and direction)
  - Balls will bounce on walls and when they encounter other balls
  - The simulation is run until a given final time

- **Question**
  - What is the average ball distribution at the end of the simulation?
  - In other words, if we take a small portion of the whole domain, how many balls is it likely to contain?
Running jobs interactively on the cluster

cluster = parcluster('MyCluster');
parpool(cluster);
Submitting MATLAB Jobs to an HPC Cluster

- Create and configure a **Cluster Profile** in MATLAB
  - Local profile (default) allows access to your computer’s hardware resources
  - Additional profiles can be configured to access remote hardware resources, typically shared workstations or HPC clusters

- Using remote hardware resources allows to...
  - Run your application faster
  - Run calculations in **batch** mode: once the job is submitted, MATLAB can be closed
  - **Improved license usage**
What about scalability to an increasing volume of data (Big Data)?
Big Data with MATLAB

- **Access**
- **Explore**
- **Prototype**
- **Scale**
- **Share/Deploy**

**Load, Analyze, Discard**

**MapReduce**

**Tall Arrays**

**Distributed Memory**

**SPMD**

**Complexity**

- **Embarrassingly Parallel**
- **Non-Partitionable**

- **out-of-memory**
- **in-memory**

- **datasetstore**
Big Data with MATLAB

Load, Analyze, Discard

**Datastore**

MapReduce

Distributed Memory

SPMD

Tall Arrays

Prototype

for k=1:max
  x = fft(data);
  y = 20*log1;

Access

Explore

Scale

Share/Deploy

Complexity

Embarrassingly Parallel

Non-Partitionable

out-of-memory

in-memory
Access Big Data
datastore

- Easily specify data set
  - Single text file (or collection of text files)
  - Hadoop HDFS
  - Images
  - Database (SQL, No SQL, )

- Preview data structure and format

- Select data to import using column names

- Incrementally read subsets of the data

```matlab
airdata = datastore('*\.csv');
airdata.SelectedVariables = {'Distance', 'ArrDelay'};
data = read(airdata);
```
Example: Access Big Data with MATLAB

- **Objective:** Access data and calculate the average trip duration of New York taxi.

- **Description:**
  - Monthly taxi ride log files
  - The local data set is **small** (~20 MB)
  - The full data set is **big** (~25 GB)

- **Approach:**
  - Preprocess data
  - Calculate average trip duration
Big Data with MATLAB

Load, Analyze, Discard

datastore

MapReduce

Tall Arrays

Distributed Memory

Embarrassingly Parallel

Non-Partitionable

Complexity

out-of-memory

in-memory

SPMD
Analyze Data using MapReduce

- Use the powerful MapReduce programming technique to analyze big data
  - **mapreduce** uses a **datastore** to process data in small chunks that individually fit into memory
  - Useful for processing multiple keys, or when intermediate results do not fit in memory

- **mapreduce** on the desktop
  - Analyze big databases
  - Increase compute capacity (Parallel Computing Toolbox)
  - Access data on HDFS to develop algorithms for use on Hadoop

- Prototype your **mapreduce** algorithms locally to easily scale to Hadoop cluster
Example: MapReduce with MATLAB

- **Objective:** Find the maximal fare amount for each payment type

- **Description:**
  - Monthly taxi ride log files
  - The local data set is small (~20 MB)
  - The full data set is big (~25 GB)

- **Approach:**
  - Preprocess data
  - Calculate the maximal fare amount for each payment type
    - Work with this subset of data for prototyping
Running jobs interactively on the Hadoop cluster

```matlab
cluster = parallel.cluster.Hadoop;
mr = mapreduce(cluster);

ds = datastore('hdfs://host/airlinedata.csv');
meanDelay = mapreduce(ds,@Mapper,@Reducer,mr);
```
Big Data with MATLAB

- Access
- Explore
- Prototype
- Scale
- Share/Deploy

Load, Analyze, Discard

Load, Analyze, Discard
datastore

MapReduce

Distributed Memory
SPMD

Tall Arrays

Complexity

Embarrassingly Parallel

Non-Partitionable

out-of-memory
in-memory
Analyze Big Data using Tall Arrays

- New data type in MATLAB

- Applicable when:
  - Data is **columnar** – with many rows
  - Overall data size is **too big to fit into memory**
  - Operations are mathematical/statistical in nature

- Statistical and machine learning applications
  - Hundreds of functions supported in MATLAB and Statistics and Machine Learning Toolbox

- Prototype your algorithms on desktop to easily scale your big data applications to compute clusters and clusters with Hadoop/Spark
Datastore and tall arrays for Big Data

- **datastore**
  - Data container for voluminous text data

- **Tall array**
  - Pointer to the data; not actually available in memory

- **Functions**
  - Operations are queued up; execution is deferred

- **Gather step**
  - Function that forces execution and gathers results back to the workspace *(result must fit in memory)*
  - Happens with `gather` function, visualizations, and fitting/modeling functions

```matlab
>> fileLoc = 'alltaxiData/*.csv';
>> ds = datastore(fileLoc);
>> tt = tall(ds);
>> tt.mins = minutes(tt.dropoff-tt.pickup);
>> mn = gather(mnTrip)
```
Example: Working with Big Data in MATLAB

- **Objective:** Create a model to predict the cost of a taxi ride in New York City

- **Inputs:**
  - Monthly taxi ride log files
  - The local data set is *small* (~20 MB)
  - The full data set is *big* (~25 GB)

- **Approach:**
  - Preprocess and explore data
  - Develop and validate predictive model (linear fit)
    - Work with subset of data for prototyping
    - Scale to full data set on a cluster
Analyze Big Data on Spark enabled Hadoop cluster

Program using “tall arrays” Program using MATLAB API for Spark

MATLAB Distributed Computing Server (MDCS)
Example: Running on Spark + Hadoop

```matlab
% Hadoop/Spark Cluster
numWorkers = 16;

setenv('HADOOP_HOME', '/dev_env/cluster/hadoop');
setenv('SPARK_HOME', '/dev_env/cluster/spark');

cluster = parallel.cluster.Hadoop;
cluster.SparkProperties('spark.executor.instances') = num2str(numWorkers);
mr = mapreduce(cluster);

% Access the data
ds = datastore('hdfs://hadoop01:54310/datasets/taxiData/*.csv');
tt = tall(ds);
```
Example: Running on Spark + Hadoop

tall Arrays for Big Data in MATLAB

Predict Cost of Taxi Ride in New York City

Analyze data from .csv files containing taxi trip information, separated by month. The data set is available from the City of New York.

Set up execution environment

```matlab
numWorkers = 16;
setenv('HADOOP_HOME', '/mathworks/test/hadoop');
setenv('SPARK_HOME', '/mathworks/test/spark');
cluster = parallel.cluster.Hadoop;
cluster.SparkProperties('spark.executor.instances') = num2str(numWorkers);
```
Summary

- Easily develop parallel MATLAB applications without being a parallel programming expert

- Speed up the execution of your MATLAB applications using additional hardware

- Develop parallel applications on your desktop and easily scale to a cluster when needed

- Scale-up your MATLAB applications developed locally to Big Data systems
MATLAB Central

- Community for MATLAB and Simulink users

- File Exchange
  - Access free files including functions, apps, examples, and models

- MATLAB Answers
  - Ask programming questions or search community-answered questions

- Newsgroup
  - Participate in technical discussions

- Blogs
  - Read commentary from engineers who design, build, and support MathWorks products
Merci

Questions?