MATLAB EXPO 2016
KOREA
4월 28일 (목)
등록 하기 matlabexpo.co.kr
• Data analytics Workflow
• Map reduce demo refine. (Car register or weather station)
Working with Big Data using MATLAB

성호현 차장
Senior Application Engineer
MathWorks Korea
Challenges of Data

“Any collection of data sets so large and complex that it becomes difficult to process using … traditional data processing applications.”

(Wikipedia)

- Various Data Sources
- Rapid data exploration
- Development of scalable algorithms
- Ease of deployment
Agenda

- How big is big?
- Reading big data
- Processing quite big data
- Processing big data
- Summary
How big is big?
What does “Big Data” even mean?

“Any collection of data sets so large and complex that it becomes difficult to process using … traditional data processing applications.”

(Wikipedia)

“Any collection of data sets so large that it becomes difficult to process using traditional MATLAB functions, which assume all of the data is in memory.”

(MATLAB)
How big is big?

Not a new problem

- In 1085 William 1\textsuperscript{st} commissioned a survey of England
  - ~2 million words and figures collected over two years
  - too big to handle in one piece
  - collected and summarized in several pieces
  - used to generate revenue (tax), but most of the data then sat unused
How big is big?
A new problem

- The Large Hadron Collider was switched back on earlier this year
  - ~600 million collisions per second (only a fraction get recorded)
  - amounts to 30 petabytes per year
  - too big to even store in one place
  - used to explore interesting science, but taking researchers a long time to get through
How big is big?
Sizes of data in this talk

- Most of our data lies somewhere in between
  - a few MB up to a few TB
  - <1GB can typically be handled in memory on one machine (small data)
  - 1-100GB can typically be handled in memory of many machines (quite big data)
  - >100GB typically requires processing in pieces using many machines (big data)
Agenda

- How big is big?
- Reading big data
- Processing quite big data
- Processing big data
- Summary
Reading big data
What tools are there?

- ImageAdapter
- imread
- Import Tool
- readtable
- load
Reading big data

What tools are there?

- ImageAdapter
- imread
- Import Tool
- readable
- load
- SystemObjects (streaming data)
- fread
- database
- textscan
- memmapfile
- xlsread
- matfile API
- SMALL
- BIG
Reading big data
What tools are there?

- ImageAdapter
- fread
- memmapfile
- SystemObjects (streaming data)
- imread
- database
- textscan
- datastore
- xmlhttp
- readtable
- load
- matfile API
Reading big data

Datastore:

- Simple interface for data in multiple files/folders
- Presents data a piece at a time
- Access pieces in serial (desktop) or in parallel (cluster)
- Back-ends for tabular text, images, databases and more
Reading big data

Datastore DEMO

```
>> ds = datastore('mnist_images', ...
    'IncludeSubfolders', true);
>> while hasdata(ds)
    imshow(read(ds)), drawnow
    end
>>
```
Agenda

- How big is big?
- Reading big data
- Processing quite big data
- Processing big data
- Summary
Processing quite big data
When the data fits in cluster memory

- Using distributed arrays
  - Use the memory of multiple machines as though it was your own
  - Client sees a “normal” MATLAB variable
  - Work happens on cluster
Processing quite big data

Distributed array functions

- Many common MATLAB functions supported: (about 250)
- Includes most linear algebra
- Scale up your maths
Processing quite big data
Multiplication of 2 NxN matrices

>> C = A * B

<table>
<thead>
<tr>
<th>N</th>
<th>1 node, 16 workers</th>
<th>2 nodes, 32 workers</th>
<th>4 nodes, 64 workers</th>
</tr>
</thead>
<tbody>
<tr>
<td>8000</td>
<td>19</td>
<td>13</td>
<td>11</td>
</tr>
<tr>
<td>16000</td>
<td>120</td>
<td>75</td>
<td>50</td>
</tr>
<tr>
<td>20000</td>
<td>225</td>
<td>132</td>
<td>86</td>
</tr>
<tr>
<td>25000</td>
<td>-</td>
<td>243</td>
<td>154</td>
</tr>
<tr>
<td>30000</td>
<td>-</td>
<td>406</td>
<td>248</td>
</tr>
<tr>
<td>35000</td>
<td>-</td>
<td>-</td>
<td>376</td>
</tr>
<tr>
<td>45000</td>
<td>-</td>
<td>-</td>
<td>743</td>
</tr>
<tr>
<td>50000</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Processor: Intel Xeon E5-class v2
16 cores, 60 GB RAM per compute node, 10 Gb Ethernet
Processing quite big data
Agenda

- How big is big?
- Reading big data
- Processing quite big data
- Processing big data
- Summary
Processing really big data
When you can never see all the data

- Can never have all the data loaded
- Must process small pieces of data independently
- Extract (“map”) some pertinent information from each independent piece
  - Typically summary statistics, example records, etc.
  - No communication between pieces
- Combine (“reduce”) this information to give a final (small) result
  - Intermediate results from each piece must be communicated
Introduction to Map-Reduce
Introduction to Map-Reduce

Example:
National popularity contest

For each page how many times do “David”, “Nicola” and “Jeremy” get mentioned?

<table>
<thead>
<tr>
<th>Name</th>
<th>Initials</th>
<th>Total mentions</th>
<th>Relative popularity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nicola</td>
<td>Nicola</td>
<td>9%</td>
<td>Nicola</td>
</tr>
<tr>
<td>David</td>
<td>David</td>
<td>53%</td>
<td>David</td>
</tr>
<tr>
<td>Jeremy</td>
<td>Jeremy</td>
<td>38%</td>
<td>Jeremy</td>
</tr>
</tbody>
</table>
Processing medium data

Map-Reduce DEMO
MATLAB with Hadoop

Datastore access data stored in HDFS from MATLAB
MATLAB Distributed Computing Server

with Hadoop

Datastore

HDFS

Node

Data

Map

Reduce

Node

Data

Map

Reduce

Node

Data

Map

Reduce

Hadoop

map.m
reduce.m
Agenda

- How big is big?
- Reading big data
- Processing quite big data
- Processing big data
- Summary
Techniques for Big Data in MATLAB

Load, Analyze, Discard
parfor, datastore,

MapReduce

Distributed Memory
SPMD and distributed arrays

Complexity

Embarrassingly Parallel

out-of-memory
in-memory

Non-Partitionable
Summary

Reading data:
1. When data gets big you need to work on pieces
2. Use datastore to read pieces from a large data-set

Processing data:
1. If it fits in memory, use MATLAB
2. If it fits in cluster memory, use distributed arrays
3. If you need to scale beyond cluster memory, use map-reduce