



TEXAS TECH UNIVERSITY

Information Technology Division™



Introduction to MATLAB in HPC Environment

Sagnik Singha

*High Performance Computing Center
(on behalf of the HPCC staff)*

Nov 9th, 2022



TEXAS TECH UNIVERSITY

Information Technology Division™

Outline

- MATLAB in the TTU HPCC environment
- Job submission methods
- Batch Submission
- Graphical Client based Submission
- Containerized MATLAB (MathWorks account)



TEXAS TECH UNIVERSITY

Information Technology Division™

Logging on to TTU HPCC

https://www.depts.ttu.edu/hpcc/userguides/general_guides/login_general.php

Connecting to RedRaider – On Campus

ssh <eraider>@login.hpcc.ttu.edu

MATLAB in the TTU HPCC - Environment setup



TEXAS TECH UNIVERSITY

Information Technology Division™

Setup MATLAB environment

```
module load matlab
```

Default version loaded (nocona): matlab/R2021b

check if MATLAB is currently loaded in your environment, please type:

```
module list
```

```
cpu-26-10:/MATLAB_testing/nocona/matlabPar$ ml matlab
cpu-26-10:/MATLAB_testing/nocona/matlabPar$ ml list
```

Currently Loaded Modules:

1) nocona/0.15.4 2) matlab/R2021b

```
cpu-26-10:/MATLAB_testing/nocona/matlabPar$
```

```
quanah:$ ml matlab
quanah:$ ml list
```

Currently Loaded Modules:

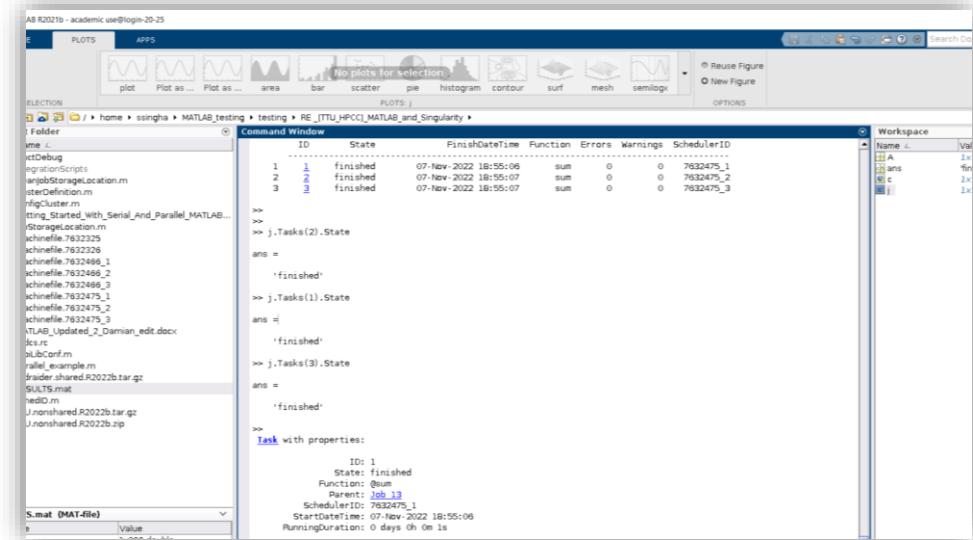
1) matlab/R2020a



TEXAS TECH UNIVERSITY

Information Technology Division™

- Graphical User Interface –
 - ❖ If code is in developmental stages
 - ❖ If direct interaction with output necessary
 - ❖ Worker node
- BATCH submission –
 - ❖ If code is in production stages
 - ❖ If multiple runs needed without direct interaction with output
 - ❖ Login node



```
#SBATCH --job-name=matlab-test
#SBATCH --output=%x.o%j
#SBATCH --error=%x.e%j
#SBATCH --partition nocona
#SBATCH --nodes=1
#SBATCH --ntasks-per-node=1
#SBATCH --time=10:00:00      ##10 hour runtime, you may change or remove this option
#SBATCH --cpus-per-task=1    # 1 core per instance

####SBATCH --mem=512000MB

# Add MATLAB to system path
module load matlab

# Run code
matlab -batch calc_pi_multi_node
~
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Batch submission

Example file's location:

/lustre/work/examples/nocona/matlab

Copy files to own workspace

```
cp -r /lustre/work/examples/nocona/matlab /home/<eraider_ID>/.
```

Example 1:

Matrix multiplication

Files:

A_matrix.dat

B_matrix.dat

atimesb.txt

matlab.sh



TEXAS TECH UNIVERSITY

Information Technology Division™

Batch submission

https://www.depts.ttu.edu/hpcc/userguides/Job_User_Guide.pdf

matlab.sh

```
#!/bin/bash
#SBATCH --job-name=matlab-test
#SBATCH --output=%x.o%j
#SBATCH --error=%x.e%j
#SBATCH --partition nocona
#SBATCH --nodes=1
#SBATCH --ntasks=1
module load matlab
matlab -batch multiplyMatrix
```

matmulfile.m

```
load A.dat;
load B.dat;
atimesb = A*B
save -ascii atimesb.txt atimesb
quit
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Batch submission and Monitoring

Submission

```
sbatch matlab.sh
```

Monitoring

```
squeue -u <username>
```

Status 1:

```
cpu-26-10:/MATLAB_testing/nocona/matlab/Example_1$ sbatch matlab.sh
Submitted batch job 7599523
cpu-26-10:/MATLAB_testing/nocona/matlab/Example_1$ squeue -u ssingha
      JOBID PARTITION      NAME      USER ST      TIME  NODES NODELIST(REASON)
      7599523    nocona matlab-t  ssingha PD      0:00      1 (None)
```

Status 2:

```
cpu-23-38:/MATLAB_testing/nocona/matlab/Example_1$ squeue -u ssingha
      JOBID PARTITION      NAME      USER ST      TIME  NODES NODELIST(REASON)
      7628458    nocona INTERACT  ssingha R     2:13:20      1 cpu-23-38
      7628776    nocona matlab-t  ssingha R      0:11      1 cpu-23-23
```

Status 3:

```
cpu-23-38:/MATLAB_testing/nocona/matlab/Example_1$ squeue -u ssingha
      JOBID PARTITION      NAME      USER ST      TIME  NODES NODELIST(REASON)
      7628458    nocona INTERACT  ssingha R     2:13:22      1 cpu-23-38
cpu-23-38:/MATLAB_testing/nocona/matlab/Example_1$
```



Batch submission - Monitoring

TEXAS TECH UNIVERSITY

Information Technology Division™

Status 2:

Monitoring

```
$ squeue -u <username>
```

cpu-23-38:/MATLAB_testing/nocona/matlab/Example_1\$ squeue -u ssingha							
JOBID	PARTITION	NAME	USER	ST	TIME	NODES	NODELIST(REASON)
7628458	nocona	INTERACT	ssingha	R	2:13:20	1	cpu-23-38
7628776	nocona	matlab-t	ssingha	R	0:11	1	cpu-23-23

cpu-23-1:\$ top grep MATLAB										
5318	ssingha	20	0	21.3g	475432	217068	S	133.3	0.1	0:03.12 MATLAB
4541	ssingha	20	0	21.3g	466260	214360	S	105.6	0.1	0:03.11 MATLAB
4537	ssingha	20	0	21.3g	482448	221040	S	100.0	0.1	0:03.14 MATLAB
4544	ssingha	20	0	21.3g	486596	220236	S	94.4	0.1	0:03.15 MATLAB
4539	ssingha	20	0	21.3g	489100	220840	S	88.9	0.1	0:03.19 MATLAB
5074	ssingha	20	0	21.3g	480076	216208	S	83.3	0.1	0:03.08 MATLAB
4534	ssingha	20	0	21.3g	476296	215080	S	77.8	0.1	0:03.11 MATLAB
4531	ssingha	20	0	21.7g	736408	230976	S	55.6	0.1	0:03.75 MATLAB
4534	ssingha	20	0	22.1g	951032	262028	S	105.0	0.2	0:06.29 MATLAB
5074	ssingha	20	0	22.1g	953908	262056	S	103.6	0.2	0:06.22 MATLAB
4539	ssingha	20	0	22.1g	952092	263620	S	101.0	0.2	0:06.25 MATLAB
4537	ssingha	20	0	22.0g	936720	262352	S	97.7	0.2	0:06.10 MATLAB
4544	ssingha	20	0	22.1g	923068	261824	S	97.7	0.2	0:06.11 MATLAB
4541	ssingha	20	0	22.0g	896352	257788	S	93.4	0.2	0:05.94 MATLAB
4531	ssingha	20	0	22.2g	976064	272068	S	89.1	0.2	0:06.45 MATLAB
5318	ssingha	20	0	22.0g	895616	256076	S	87.8	0.2	0:05.78 MATLAB
3698	ssingha	20	0	23.2g	1.2g	312528	S	0.3	0.2	0:13.07 MATLAB
5074	ssingha	20	0	22.8g	1.1g	290956	S	99.3	0.2	0:09.23 MATLAB
4544	ssingha	20	0	22.5g	1.1g	292092	S	99.0	0.2	0:09.11 MATLAB



TEXAS TECH UNIVERSITY

Information Technology Division™

Batch submission - Monitoring

Status 1:

```
cpu-26-10:/MATLAB_testing/nocona/matlab/Example_1$ sbatch matlab.sh
Submitted batch job 7599523
cpu-26-10:/MATLAB_testing/nocona/matlab/Example_1$ squeue -u ssingha
      JOBID PARTITION      NAME      USER ST      TIME  NODES NODELIST(REASON)
      7599523    nocona matlab-t  ssingha PD      0:00      1 (None)
```

Monitoring

```
squeue -u <username>
```

Status 2:

```
cpu-23-38:/MATLAB_testing/nocona/matlab/Example_1$ squeue -u ssingha
      JOBID PARTITION      NAME      USER ST      TIME  NODES NODELIST(REASON)
      7628458    nocona INTERACT  ssingha R     2:13:20      1 cpu-23-38
      7628776    nocona matlab-t  ssingha R      0:11      1 cpu-23-23
```

Files

- matlab-test.e<JOB_ID>
- matlab-test.o<JOB_ID>
- atimesb.txt

Status 3:

```
cpu-23-38:/MATLAB_testing/nocona/matlab/Example_1$ squeue -u ssingha
      JOBID PARTITION      NAME      USER ST      TIME  NODES NODELIST(REASON)
      7628458    nocona INTERACT  ssingha R     2:13:22      1 cpu-23-38
cpu-23-38:/MATLAB_testing/nocona/matlab/Example_1$
```



Batch submission

TEXAS TECH UNIVERSITY

Information Technology Division™

Example 2:

Matrix multiplication

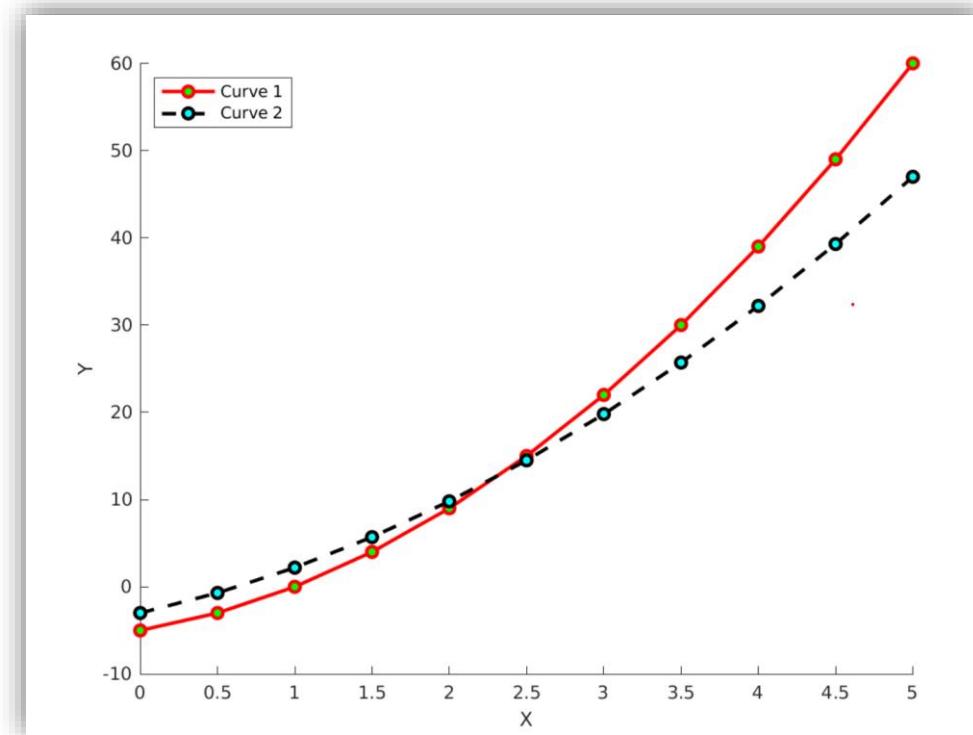
Files:

example_2.m

matlab.sh

Files

- matlab-test.e<JOB_ID>
- matlab-test.o<JOB_ID>
- func_plot.png





Batch submission -

Parallel implementation, Maximum Eigenvalue of a matrix

TEXAS TECH UNIVERSITY

Information Technology Division™

Serial version

Code:

```
%Start timing
tic

%Calculates spectral radius of each matrix and
displays results
n = 10;
A = 500;
a = zeros(n);
for i = 1:n*n
    a(i) = max(abs(eig(rand(A))));  
    ↑
end
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %5.2f
seconds. \n', time);
```

Uniform distribution
of random numbers

Parallel version

Parallel implementation – Maximum Eigenvalue of a matrix



TEXAS TECH UNIVERSITY

Information Technology Division™

Serial version

Code:

```
%Start timing
tic

%Calculates spectral radius of each matrix and
displays results
n = 10;
A = 500;
a = zeros(n);
for i = 1:n*n
    a(i) = max(abs(eig(rand(A)) ));
end
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %5.2f
seconds. \n', time);
```

P 1

P 2

Parallel version

Code:

```
c = parcluster('local'); % local is a cluster profile name
sz = str2num([getenv('SLURM_CPUS_PER_TASK'));
if isempty(sz), sz = maxNumCompThreads; end
if isempty(gcp('nocreate')), c.parpool(sz); end

%Start timing
tic

%Calculates spectral radius of each matrix and displays
results
n = 10;
A = 500;
a = zeros(n);
parfor i = 1:n*n
    a(i) = max(abs(eig(rand(A)) ));
end
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %5.2f seconds.
\n', time);
```

Parallel implementation – Components



TEXAS TECH UNIVERSITY

Information Technology Division™

P1-

- Creates a cluster object representing the cluster identified by the cluster profile name *local*

P1

Code:

```
c = parcluster('local'); % local is a cluster profile name
sz = str2num([getenv('SLURM_CPUS_PER_TASK'));
if isempty(sz), sz = maxNumCompThreads; end
if isempty(gcp('nocreate')), c.parpool(sz); end

%Start timing
tic

%Calculates spectral radius of each matrix and displays
results
n = 10;
A = 500;
a = zeros(n);
parfor i = 1:n*n
    a(i) = max(abs(eig(rand(A)))); % calculate spectral radius
end
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %5.2f seconds.\n', time);
```

Parallel version

Parallel implementation – Components



TEXAS TECH UNIVERSITY

Information Technology Division™

P1-

- Picks up *--cpus-per-task* from batch submit script
- Assigns to the variable *sz*
- String to number check to ensure correct format

P1

Parallel version

Code:

```
c = parcluster('local'); % local is a cluster profile name
sz = str2num([getenv('SLURM_CPUS_PER_TASK'));
if isempty(sz), sz = maxNumCompThreads; end
if isempty(gcp('nocreate')), c.parpool(sz); end

%Start timing
tic

%Calculates spectral radius of each matrix and displays
results
n = 10;
A = 500;
a = zeros(n);
parfor i = 1:n*n
    a(i) = max(abs(eig(rand(A)))); % calculate spectral radius
end
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %5.2f seconds.\n', time);
```

Parallel implementation – Components



TEXAS TECH UNIVERSITY

Information Technology Division™

P1-

- If *--cpus-per-task* not defined, maximum number of cores are allotted

P1

Parallel version

Code:

```
c = parcluster('local'); % local is a cluster profile name
sz = str2num([getenv('SLURM_CPUS_PER_TASK'));
if isempty(sz), sz = maxNumCompThreads; end
if isempty(gcp('nocreate')), c.parpool(sz); end

%Start timing
tic

%Calculates spectral radius of each matrix and displays
results
n = 10;
A = 500;
a = zeros(n);
parfor i = 1:n*n
    a(i) = max(abs(eig(rand(A)) ));
end
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %5.2f seconds.\n', time);
```

Parallel implementation – Components



TEXAS TECH UNIVERSITY

Information Technology Division™

P1-

- If a parallel pool is not created, creates one and takes the size from *--cpus-per-task*

P1

Parallel version

Code:

```
c = parcluster('local'); % local is a cluster profile name
sz = str2num([getenv('SLURM_CPUS_PER_TASK'));
if isempty(sz), sz = maxNumCompThreads; end
if isempty(gcp('nocreate')), c.parpool(sz); end

%Start timing
tic

%Calculates spectral radius of each matrix and displays
results
n = 10;
A = 500;
a = zeros(n);
parfor i = 1:n*n
    a(i) = max(abs(eig(rand(A)))); % Compute spectral radius
end
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %5.2f seconds.\n', time);
```

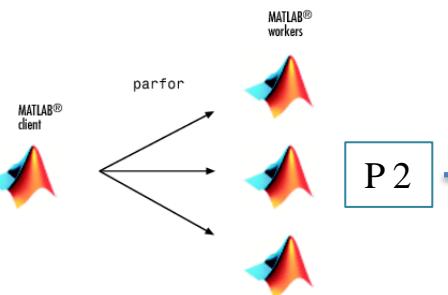
Parallel implementation – Components



TEXAS TECH UNIVERSITY

Information Technology Division™

P2- *parfor* splits the execution of
for-loop iterations over the workers
in a parallel pool



Parallel version

Code:

```
c = parcluster('local'); % local is a cluster profile name
% Query for available cores (assume either Slurm or PBS)
sz = str2num([getenv('SLURM_CPUS_PER_TASK'));
if isempty(sz), sz = maxNumCompThreads; end
if isempty(gcp('nocreate')), c.parpool(sz); end

%Start timing
tic

%Calculates spectral radius of each matrix and displays
results
n = 10;
A = 500;
a = zeros(n);
parfor i = 1:n*n
    a(i) = max(abs(eig(rand(A)) ));
end
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %.5f seconds.\n', time);
```

Parallel implementation- Benefits



TEXAS TECH UNIVERSITY

Information Technology Division™

1st Run

Serial version	Parallel version
1 core	10 cores
1304.93 seconds(~21.75 mins)	133.62 seconds (~2.3 mins)

Next Run

Serial version	Parallel version
1 core	10 cores
10.31 seconds	1.89 seconds



Parallel implementation- Multi core parallel implementation

TEXAS TECH UNIVERSITY

Information Technology Division™

Nocona – 128 cores/node

<https://www.depts.ttu.edu/hpcc/operations/equipment.php>

Partition:	Nocona	Quanah / XLQuanah	Matador	Toreador	Ivy
Type	CPU	CPU	GPU	GPU	Auxiliary CPU*
Total Nodes	240	467 / 16	20	11	50 / 2
Theoretical Max	983 TFLOPS /19 TFLOPS	565 TFLOPS /19 TFLOPS	280 TFLOPS	287 TFLOPS	40 TFLOPS
Benchmarked	804 TFLOPS / (N/A)	485 TFLOPS / (N/A)	226 TFLOPS		N/A
OS	CentOS 8.1	CentOS 7.4 /CentOS 8.1	CentOS 8.1	CentOS 8.1	Rocky Linux 8.5 /CentOS 8.1
Manufacturer	Dell	Dell	Dell	Dell	Dell
Node Model	PowerEdge C6525	PowerEdge C6320	PowerEdge R740	Poweredge R7525	PowerEdge C6220 II
Cooling	Liquid Cooled	Air Cooled	Air Cooled	Air Cooled	Air Cooled
Processor Make and Model	AMD EPYC™ 7702	Intel Xeon E5-2695 v4	Intel Xeon Gold 6242	AMD EPYC™ 7262	Intel Xeon E5-2670v2
Family	Rome	Broadwell	Cascade Lake	Rome	Ivy Bridge
Cores/Processor	64	18	20	8	10
Cores/Node	128	36	40 cpu + 1280 tensor + 10,240 CUDA	16 cpu + 1,296 tensor + 20,736 CUDA	20
Total Cores In Partition	30,720 / 576	16,812 / 576	800 cpu + 25,600 tensor + 204,000 CUDA	528 cpu + 14,256 tensor+ 200,000 CUDA	1,000 / 40



TEXAS TECH UNIVERSITY

Information Technology Division™

MATLAB Graphical Interface

Interactive session

```
$ interactive -p nocona -c 8
```

```
$ interactive -h
```

Number of cores

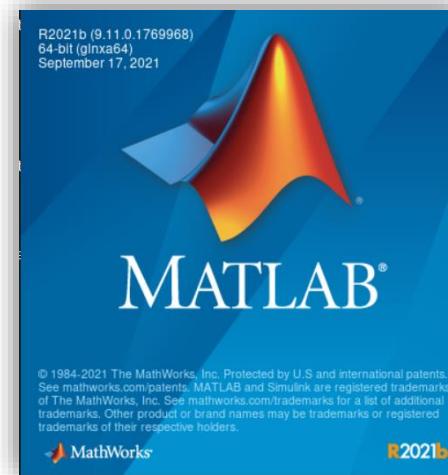
```
cpu-24-33:/MATLAB_testing/nocona/matlabPar$ ml matlab
cpu-24-33:/MATLAB_testing/nocona/matlabPar$ ml list
```

Currently Loaded Modules:

```
1) nocona/0.15.4 2) matlab/R2021b
cpu-24-33:/MATLAB_testing/nocona/matlabPar$
```

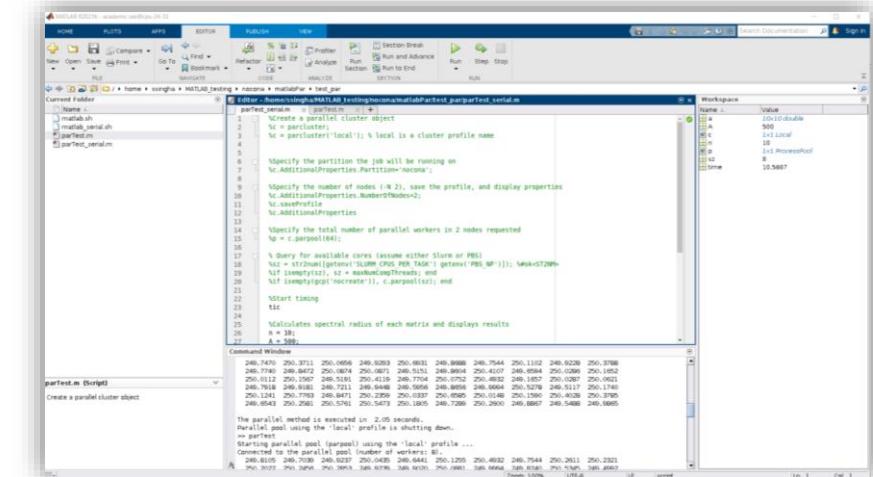
Setting up environment

```
$module load matlab
$ml list
```



Launching Client

```
$matlab
```



Parallel implementation – Maximum Eigenvalue of a matrix



TEXAS TECH UNIVERSITY

Information Technology Division™

Link: /matlab/matlabPar/parTest_interactive

Serial version

parTest_serial.m

Parallel version

parTest.m

Parallel implementation – Maximum Eigenvalue of a matrix



TEXAS TECH UNIVERSITY

Information Technology Division™

Serial version

Code:

```
%Start timing
tic

%Calculates spectral radius of each matrix and
displays results
n = 10;
A = 500;
a = zeros(n);
for i = 1:n*n
    a(i) = max(abs(eig(rand(A))));
```

end

```
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %5.2f
seconds. \n', time);
```

Parallel version

Code:

```
c = parcluster('local'); % local is a cluster profile name

p = c.parpool(10);

%Start timing
tic

%Calculates spectral radius of each matrix and displays
results
n = 10;
A = 500;
a = zeros(n);
parfor i = 1:n*n
    a(i) = max(abs(eig(rand(A))));
```

end

```
disp(a)

%End timing
time = toc;

%Display the time of computation
fprintf('The parallel method is executed in %5.2f seconds.
\n', time);
```

Parallel implementation – Maximum Eigenvalue of a matrix



TEXAS TECH UNIVERSITY

Information Technology Division™

Object representing cluster identified by the cluster profile name *local*

Starts a parallel pool of 10 workers using the cluster object *c*

- Splits the execution of for-loop iterations over the workers in a parallel pool
- Say $n^2=10$, 10 workers available, MATLAB assigns it to each core

Parallel version

Code:

```
c = parcluster('local'); % local is a cluster profile name  
  
p = c.parpool(10);  
  
%Start timing  
tic  
  
%Calculates spectral radius of each matrix and displays  
results  
n = 10;  
A = 500;  
a = zeros(n);  
parfor i = 1:n*n  
    a(i) = max(abs(eig(rand(A))));  
end  
disp(a)  
  
%End timing  
time = toc;  
  
%Display the time of computation  
fprintf('The parallel method is executed in %.5.2f seconds.\n', time);
```

MATLAB Graphical Interface – Parallel implementation via parpool



TEXAS TECH UNIVERSITY

Information Technology Division™

Client outputs

```
>> parTest_serial  
...  
The parallel method is executed in 12.94 seconds.
```

```
>> parTest  
Starting parallel pool (parpool) using the 'local' profile ...  
Connected to the parallel pool (number of workers: 1).  
...  
The parallel method is executed in 10.69 seconds.  
Parallel pool using the 'local' profile is shutting down.
```

```
>>... Connected to the parallel pool (number of workers: 2).  
The parallel method is executed in 5.70 seconds ...
```

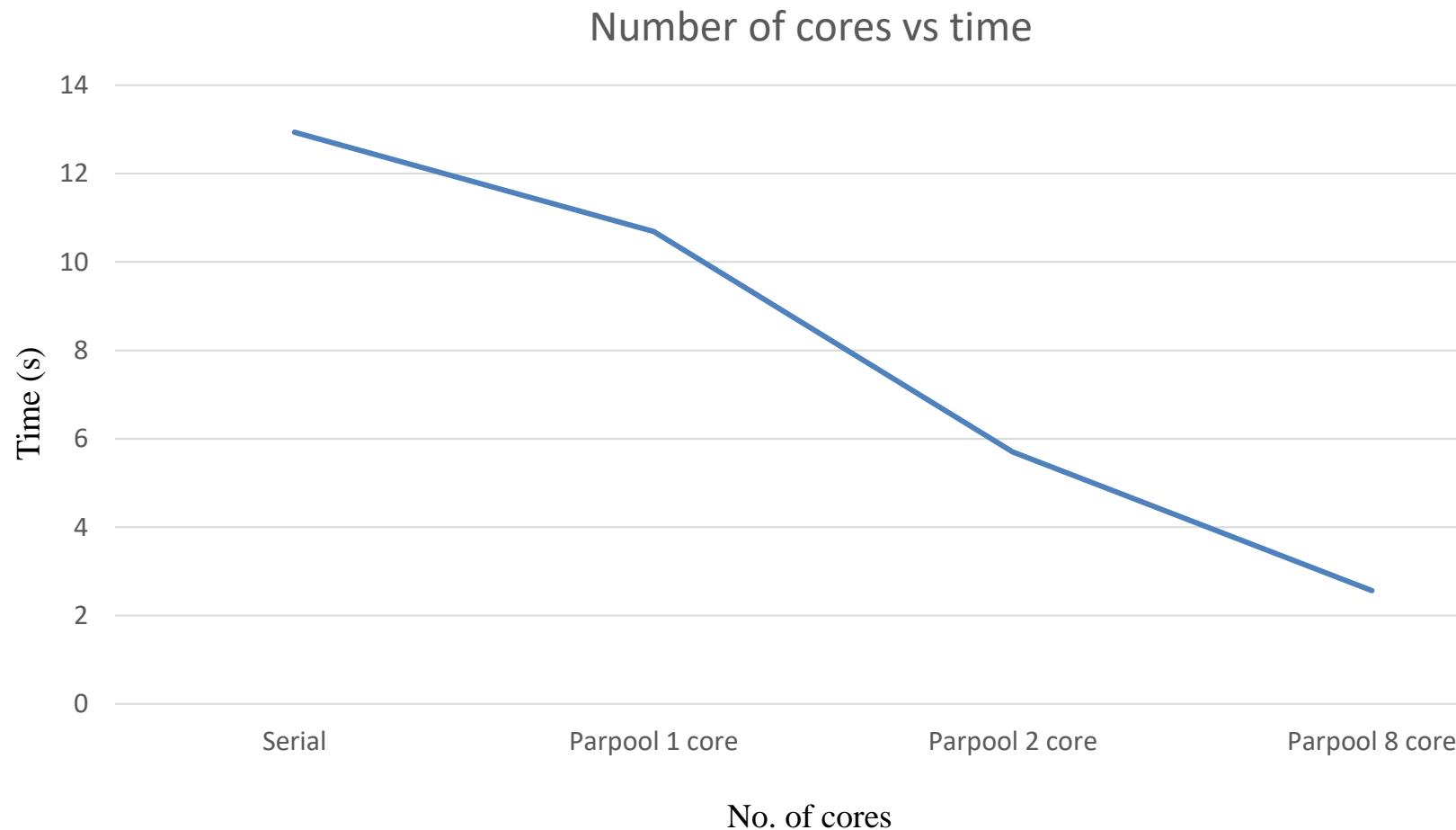
```
>>... Connected to the parallel pool (number of workers: 8).  
The parallel method is executed in 2.56 seconds ...
```

MATLAB Graphical Interface – Parallel implementation via parpool



TEXAS TECH UNIVERSITY

Information Technology Division™





Parallel implementation- Multi core parallel implementation

TEXAS TECH UNIVERSITY

Information Technology Division™

Nocona – 128 cores/node

<https://www.depts.ttu.edu/hpcc/operations/equipment.php>

Partition:	Nocona	Quanah / XLQuanah	Matador	Toreador	Ivy
Type	CPU	CPU	GPU	GPU	Auxiliary CPU*
Total Nodes	240	467 / 16	20	11	50 / 2
Theoretical Max	983 TFLOPS /19 TFLOPS	565 TFLOPS /19 TFLOPS	280 TFLOPS	287 TFLOPS	40 TFLOPS
Benchmarked	804 TFLOPS / (N/A)	485 TFLOPS / (N/A)	226 TFLOPS		N/A
OS	CentOS 8.1	CentOS 7.4 /CentOS 8.1	CentOS 8.1	CentOS 8.1	Rocky Linux 8.5 /CentOS 8.1
Manufacturer	Dell	Dell	Dell	Dell	Dell
Node Model	PowerEdge C6525	PowerEdge C6320	PowerEdge R740	Poweredge R7525	PowerEdge C6220 II
Cooling	Liquid Cooled	Air Cooled	Air Cooled	Air Cooled	Air Cooled
Processor Make and Model	AMD EPYC™ 7702	Intel Xeon E5-2695 v4	Intel Xeon Gold 6242	AMD EPYC™ 7262	Intel Xeon E5-2670v2
Family	Rome	Broadwell	Cascade Lake	Rome	Ivy Bridge
Cores/Processor	64	18	20	8	10
Cores/Node	128	36	40 cpu + 1280 tensor + 10,240 CUDA	16 cpu + 1,296 tensor + 20,736 CUDA	20
Total Cores In Partition	30,720 / 576	16,812 / 576	800 cpu + 25,600 tensor + 204,000 CUDA	528 cpu + 14,256 tensor+ 200,000 CUDA	1,000 / 40



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Asynchronous Batch GUI

- Run once
- Re-runs deletes older profile, creates a new profile
- All saved data deleted

- `>>c=parcluster;`
- Default cluster profile is set as **redraider**
- Always makes a new secondary job submission to Slurm

- `>>c=parcluster('local');`
- Opens cluster profile to utilize present allocation
- Uses just resources on present node, no submission to scheduler

```
Command Window
>> configCluster
    Must set Partition and NumberOfNodes before submitting jobs to REDRAIDER. E.g.

    >> c = parcluster;
    >> c.AdditionalProperties.Partition = 'nocona';
    >> c.AdditionalProperties.NumberOfNodes = 1;
    >> c.saveProfile

Complete. Default cluster profile set to "redraider R2021b".
>> c=parcluster

c =
    Generic Cluster

    Properties:

        Profile: redraider R2021b
        Modified: false
        Host: login-20-25.localdomain
        NumWorkers: 100000
        NumThreads: 1

        JobStorageLocation: /home/ssingha/.matlab/Bp_cluster_jobs/redraider/R2021b/shared
        ClusterMatlabRoot: /opt/apps/snfs/RedRaider/matlab/R2021b
        OperatingSystem: unix

        RequiresOnlineLicensing: false
        PluginScriptsLocation: /home/ssingha/MATLAB_testing/testing/PE_[TTU_HPCC]_MATLAB_and_Singularity/IntegrationScr...
        AdditionalProperties: List properties

    Associated Jobs:

        Number Pending: 0
        Number Queued: 0
        Number Running: 2
        Number Finished: 4

>> |
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Asynchronous Batch GUI

Profile Properties

- Slurm properties that can be modified
- *AdditionalSubmitArgs* – Manually allows user to pass arguments, flags say constrains of the cluster
- `c.AdditionalProperties.NumberofNodes = 1;`
- `c.AdditionalProperties.Partitions='nocona';`
- `c.saveProfile`
- Creates a submission script and performs the submission for the user

```
Command Window
AdditionalSubmitArgs: ''
  Constraint: ''
  EmailAddress: ''
  EnableDebug: 0
  GpusPerNode: 0
  MemUsage: '4gb'
  NumberOfNodes: 1
  Partition: ''
  ProcsPerNode: 0
  RequireExclusiveNode: 0
  Reservation: ''
  UseSmpd: 0
  WallTime: ''

>> t
Unrecognized function or variable 't'.

>>
>> c.AdditionalProperties.NumberofNodes=1;
>> c.AdditionalProperties.Partition='nocona';
>> c.saveProfile
>>
>> c.AdditionalProperties

ans =
AdditionalProperties with properties:
  AccountName: ''
  AdditionalSubmitArgs: ''
  Constraint: ''
  EmailAddress: ''
  EnableDebug: 0
  GpusPerNode: 0
  MemUsage: '4gb'
  NumberOfNodes: 1
  Partition: 'nocona'
  ProcsPerNode: 0
  RequireExclusiveNode: 0
  Reservation: ''
  UseSmpd: 0
  WallTime: ''
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Asynchronous Batch GUI

Job object name Function name Number of outputs

Inputs to function

```
>>j=batch(c,@pwd,1,{});
```

>>j.State – returns status of job

>>j.fetchOutputs – returns output of job

>>j.diary – returns command line outputs

Command Window

```
>> j=batch(c,@pwd,1,{});  
  
additionalSubmitArgs =  
  
    '--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1 -p nocona --mem-per-cpu=4gb --nodes=1'  
  
>> j.State  
  
ans =  
  
    'finished'  
  
>> j.fetchOutputs  
|  
ans =  
  
    1x1 cell array  
  
    {'/home/ssingha/MATLAB_testing/testing/RE_[TTU_HPCC]_MATLAB_and_Singularity'}
```

```
>> j=batch(c,@parallel_example,1,{300},{'pool'},10);  
  
additionalSubmitArgs =  
  
    '--ntasks=11 --cpus-per-task=1 --ntasks-per-core=1 -p nocona --mem-per-cpu=4gb --nodes=1'  
  
>> j.State  
  
ans =  
  
    'running'  
  
>> j.State  
  
ans =  
  
    'finished'  
  
>> j.fetchOutputs  
  
ans =  
  
    1x1 cell array  
fx
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Asynchronous Batch GUI

Serial Example

```
function [t, A] = parallel_example(iter)
if nargin==0
    iter = 8;
end
disp('Start sim')
t0 = tic;
for idx = 1:iter
    A(idx) = idx;
    pause(2)
    idx
end
t = toc(t0);
disp('Sim completed')
save RESULTS A
end
```

```
>> j=batch(c,@serial_example,1,{8});

>> j.State

>> j.fetchOutputs

ans =
1×1 cell array
 {[16.0098]}
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Asynchronous Batch GUI

Parallel Example

```
function [t, A] = parallel_example(iter)
if nargin==0
    iter = 8;
end
disp('Start sim')
t0 = tic;
parfor idx = 1:iter
    A(idx) = idx;
    pause(2)
    idx
end
t = toc(t0);
disp('Sim completed')
save RESULTS A
end
```

```
>> j=batch(c,@serial_example,1,{8});
{[16.0098]}

>>j=batch(c,@parallel_example,1,{8},'pool',2);
{[8.3045]}

>> =batch(c,@parallel_example,1,{8},'pool',4);
{[4.3124]}
```



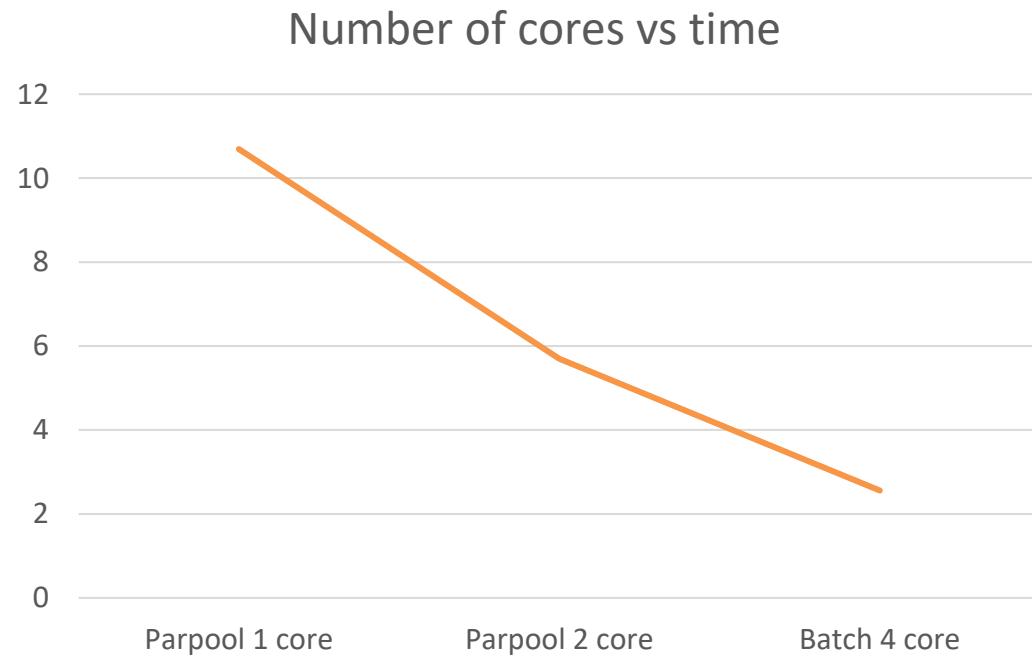
Job submission – Asynchronous Batch GUI

TEXAS TECH UNIVERSITY

Information Technology Division™

Parallel Example

```
function [t, A] = parallel_example(iter)
if nargin==0
    iter = 8;
end
disp('Start sim')
t0 = tic;
parfor idx = 1:iter
    A(idx) = idx;
    pause(2)
    idx
end
t = toc(t0);
disp('Sim completed')
save RESULTS A
end
```





TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Asynchronous Batch GUI

```
>> j=batch(c,@sum,1,{{1 1}});  
  
>> j=createJob(c);  
  
createTask(j,@sum,1,{{1 1}});  
  
createTask(j,@sum,1,{{1 1}});  
  
createTask(j,@sum,1,{{1 1}});  
  
submit(j);
```

Job Arrays

```
>>  
>> j=createJob(c);  
createTask(j,@sum,1,{{1 1}});  
createTask(j,@sum,1,{{1 1}});  
createTask(j,@sum,1,{{1 1}});  
submit(j);  
  
additionalSubmitArgs =  
  
'--ntasks=1 --cpus-per-task=1 --ntasks-per-core=1 -p nocona --mem-per-cpu=4gb --nodes=1'  
  
>> j.Tasks  
  
ans =  
  
3x1 Task array:  
  


|   | ID                | State   | FinishDateTime | Function | Errors | Warnings | SchedulerID |
|---|-------------------|---------|----------------|----------|--------|----------|-------------|
| 1 | <a href="#">1</a> | running |                | sum      | 0      | 0        | 7632475_1   |
| 2 | <a href="#">2</a> | pending |                | sum      | 0      | 0        | 7632475_2   |
| 3 | <a href="#">3</a> | pending |                | sum      | 0      | 0        | 7632475_3   |

  
>>  
>>  
>>
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Asynchronous Batch GUI

Job Arrays

```
>> j.Tasks
```

```
>> j.Tasks
```

	ID	State	FinishDateTime	Function	Errors	Warnings
1	<u>1</u>	pending		sum	0	0
2	<u>2</u>	pending		sum	0	0
3	<u>3</u>	pending		sum	0	0

```
>> j.Tasks(2).cancel  
>> j.Tasks
```

```
ans =
```

	ID	State	FinishDateTime	Function	Errors	Warnings
1	<u>1</u>	pending		sum	0	0
2	<u>2</u>	finished	09-Nov-2022 00:11:14	sum	1	0
3	<u>3</u>	pending		sum	0	0

Monitoring job parameters

```
ans =
```

ID	InputArguments	listAutoAttachedFiles	loadobj	Name	NumOutputArguments	OutputArguments	Parent
1	<u>ce</u>						

```
>> j.Tasks(1).
```

```
ans =
```

1
<u>>> j.Tasks(1).</u>



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Interactive pool submission

Opening an interactive parpool (local profile)

```
>> c=parcluster('local');
>> p=c.parpool(n)
```

n is the number of cores across the cluster requested

Command Window

```
>>
>> p=c.parpool(10)
Starting parallel pool (parpool) using the 'redraider R2021b' profile ...

additionalSubmitArgs =
    '--ntasks=10 --cpus-per-task=1 --ntasks-per-core=1 -p nocona --mem-per-cpu=4gb --nodes=1'

Connected to the parallel pool (number of workers: 10).

p =
ClusterPool with properties:

    Connected: true
    NumWorkers: 10
    Cluster: redraider R2021b
    AttachedFiles: {}
    AutoAddClientPath: true
    IdleTimeout: 30 minutes (30 minutes remaining)
    SpmdEnabled: true
    EnvironmentVariables: {}
```

Running a parpool job (local profile)

```
>> parallel_example
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Interactive pool submission

Another parallel
construct like
parfor

```
>> spmd,getCurrentWorker,end
Worker 1:
ans =
Worker
    Host: cpu-26-56.localdomain
    ComputerType: GLNXA64
    ProcessId: 95462

Worker 2:
ans =
Worker
    Host: cpu-26-56.localdomain
    ComputerType: GLNXA64
    ProcessId: 95463
...
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Interactive pool submission

Shutting down Parallel pool

redraider.shared.R2022b.tar.gz
RESULTS.mat
schedID.m
TTU.nonshared.R2022b.tar.gz
TTU.nonshared.R2022b.zip

parallel_example.m (Function)

parallel_example(iter)

Shut Down Parallel Pool

Parallel Preferences

Shut Down Parallel Pool

```
296
ans =
296
ans =
300
Sim completed
ans =
60.3814
f1 >>
```

redraider.shared.R2022b.tar.gz
RESULTS.mat
schedID.m
TTU.nonshared.R2022b.tar.gz
TTU.nonshared.R2022b.zip

parallel_example.m (Function)

parallel_example(iter)

```
ans =
300
Sim completed
ans =
60.3814
Parallel pool using the 'redraider R2021b' profile is shutting down
>> clear all
>> maxNumCompThreads

ans =
2
f1 >>
```

>>clear all



TEXAS TECH UNIVERSITY

Information Technology Division™

Job submission – Interactive pool submission

Monitoring

The screenshot shows the MATLAB interface with the Parallel tab selected in the top menu bar. The main window displays a table of jobs under the 'Monitor jobs' section. The table includes columns for ID, Username, Submit Time, Finish Time, Status, and Description. The status column indicates the current state of each job, such as 'finished', 'pending', or 'queued'. The description column provides a brief summary of the job type.

ID	Username	Submit Time	Finish Time	Status	Description	
1	ssingha	Fri Oct 28 14:26:40 CDT 2022	Fri Oct 28 14:26:56 CDT 2022	1	finished	Batch job running function
2	ssingha	Fri Oct 28 14:36:56 CDT 2022	Fri Oct 28 14:38:25 CDT 2022	11	finished	Batch job running function
3	ssingha	Fri Oct 28 14:46:09 CDT 2022	Fri Oct 28 14:46:24 CDT 2022	3	finished	Independent job
5	ssingha	Fri Oct 28 15:09:42 CDT 2022	Fri Oct 28 15:14:19 CDT 2022	201	finished	Batch job running function
6	ssingha	Fri Oct 28 15:14:55 CDT 2022	Fri Oct 28 15:16:21 CDT 2022	11	finished	Batch job running function
8	ssingha	Mon Nov 07 10:31:45 CST 2022	Mon Nov 07 10:32:01 CST 2022	1	finished	Batch job running function
9	ssingha	Mon Nov 07 10:33:56 CST 2022	Mon Nov 07 10:35:26 CST 2022	11	finished	Batch job running function
10	ssingha			0	pending	Independent job
11	ssingha	Mon Nov 07 10:53:59 CST 2022	Mon Nov 07 10:54:11 CST 2022	3	finished	Independent job
12	ssingha			3	pending	Independent job
13	ssingha	Mon Nov 07 10:54:55 CST 2022	Mon Nov 07 10:55:07 CST 2022	3	finished	Independent job
14	ssingha	Mon Nov 07 20:18:34 CST 2022		21	queued	Batch job running function

Below the monitor window, the MATLAB interface shows the Current Folder browser, Command Window, and Workspace browser. The Command Window displays MATLAB code and its execution results, indicating a job is currently queued.



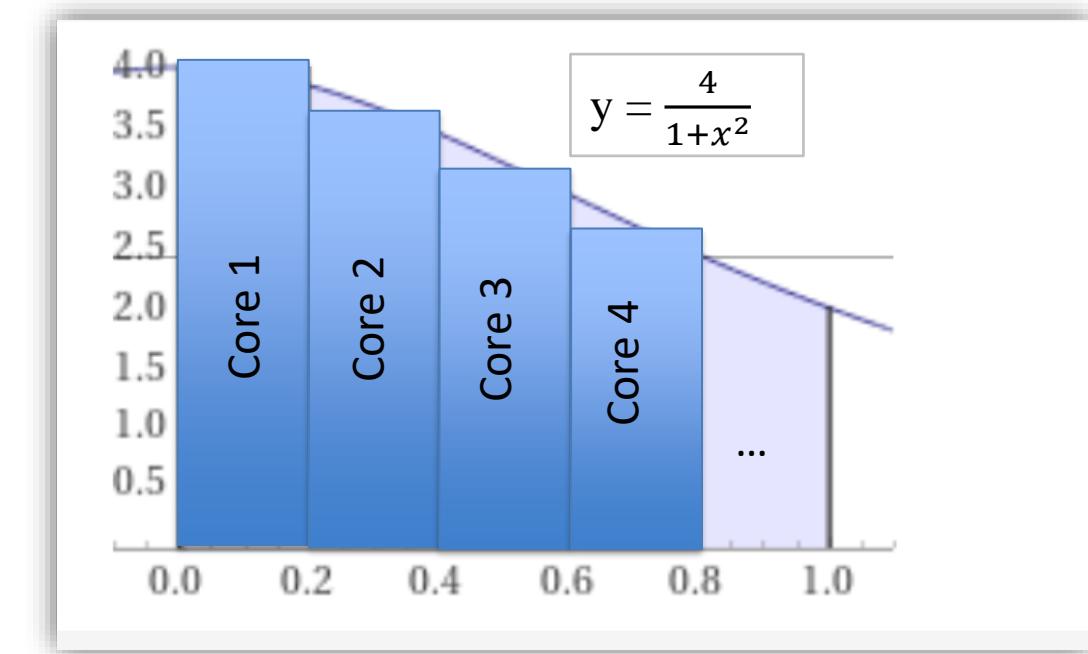
TEXAS TECH UNIVERSITY

Information Technology Division™

Parallel job implementation example

Calculation of π (based on Leibniz formula)

$$\int_0^1 \frac{4}{1+x^2} dx = 4 \arctan = \pi$$



Plotted in Wolfram Alpha



TEXAS TECH UNIVERSITY

Information Technology Division™

Single node parallel

```
function calc_pi

c = parcluster('local');

%sz = str2num([getenv('SLURM_CPUS_PER_TASK')]); %#ok<ST2NM>
sz = str2num([getenv('SLURM_CPUS_PER_TASK')]); %#ok<ST2NM>
if isempty(sz), sz = maxNumCompThreads; end

if isempty(gcp('nocreate')), c.parpool(sz); end

spmd
    a = (labindex - 1)/numlabs;
    b = labindex/numlabs;
    fprintf('Subinterval: [%-4g, %-4g]\n', a, b)

    myIntegral = integral(@quadpi, a, b);
    fprintf('Subinterval: [%-4g, %-4g] Integral: %4g\n', a, b, myIntegral)

    piApprox = gplus(myIntegral);
end

approx1 = piApprox{1}; % 1st element holds value on worker 1
fprintf('pi : %.18f\n', pi)
fprintf('Approximation: %.18f\n', approx1)
fprintf('Error : %g\n', abs(pi - approx1))

function y = quadpi(x)
%QUADPI Return data to approximate pi.

% Derivative of 4*atan(x)
y = 4./(1 + x.^2);
```

Multi node parallel

```
function calc_pi_multi_node

c = parcluster;

% Required fields
c.AdditionalProperties.WallTime = '00:20:00';
c.AdditionalProperties.Partition = 'nocona';
c.AdditionalProperties.NumberOfNodes = 1;

if isempty(gcp('nocreate')), c.parpool(20); end

spmd
    a = (labindex - 1)/numlabs;
    b = labindex/numlabs;
    fprintf('Subinterval: [%-4g, %-4g]\n', a, b)

    myIntegral = integral(@quadpi, a, b);
    fprintf('Subinterval: [%-4g, %-4g] Integral: %4g\n', a, b, myIntegral)

    piApprox = gplus(myIntegral);
end

approx1 = piApprox{1}; % 1st element holds value on worker 1
fprintf('pi : %.18f\n', pi)
fprintf('Approximation: %.18f\n', approx1)
fprintf('Error : %g\n', abs(pi - approx1))

function y = quadpi(x)
%QUADPI Return data to approximate pi.

% Derivative of 4*atan(x)
y = 4./(1 + x.^2);
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Parallel job implementation example

Accuracy of numerical calculation

```
>> calc_pi
Starting parallel pool (parpool) using the 'local' profile ...
Connected to the parallel pool (number of workers: 4).
Worker 1:
Subinterval: [0    , 0.25]
Subinterval: [0    , 0.25]  Integral: 0.979915
Worker 2:
Subinterval: [0.25, 0.5 ]
Subinterval: [0.25, 0.5 ]  Integral: 0.874676
Worker 3:
Subinterval: [0.5  , 0.75]
Subinterval: [0.5  , 0.75]  Integral: 0.719414
Worker 4:
Subinterval: [0.75, 1    ]
Subinterval: [0.75, 1    ]  Integral: 0.567588
pi      : 3.141592653589793116
Approximation: 3.141592653589793560
Error    : 4.44089e-16
```

Lower

```
>> calc_pi_multi_node
Starting parallel pool (parpool) using the 'teton R2022a' profile ...
additionalSubmitArgs =
    '--ntasks=20 --cpus-per-task=1 --ntasks-per-core=1
Connected to the parallel pool (number of workers: 20).
Worker 1:
Subinterval: [0    , 0.025]
Worker 2:
Subinterval: [0.025, 0.05]
Worker 3:
Subinterval: [0.05, 0.075]
.
.
.
Worker 18:
Subinterval: [0.425, 0.45]  Integral: 0.0839331
Worker 20:
Subinterval: [0.525, 0.55]  Integral: 0.0775848
pi      : 3.141592653589793116
Approximation: 3.141592653589793116
Error    : 0
```

Lower



TEXAS TECH UNIVERSITY

Information Technology Division™

Containerized MATLAB

1. Login into login.hpcc.ttu.edu using your eRaider account and password.
2. Get an interactive session on a partition (say nocona)
\$ interactive -p nocona
3. Check singularity installation and version
\$ singularity --version
singularity version 3.7.3-1.el8
4. Pull a sample singularity container to get started
\$ singularity pull shub://singularityhub/hello-world

```
cpu-25-27:/singularity_testing/nocona$ singularity pull shub://singularityhub/hello-world
INFO:  Downloading shub image
59.8MiB / 59.8MiB [=====] 100 % 44.5 MiB/s 0s
cpu-25-27:/singularity_testing/nocona$ ls
hello-world_latest.sif
```



TEXAS TECH UNIVERSITY

Information Technology Division™

Containerized MATLAB

1. In the event of errors, re-runs execute
\$ singularity cache list
to list container files, present in the account memory
\$ singularity cache clean
to remove such files before beginning a fresh pull
2. Execute the singularity image file (*.sif)
\$./hello-world_latest.sif
Tacotacotaco
3. Alternate ways to explore yourself:
Directly run a singularity container
\$ singularity run hello-world_latest.sif
4. Pull latest MATLAB docker container:
\$ singularity pull docker://mathworks/matlab:r2022b



TEXAS TECH UNIVERSITY

Information Technology Division™

Thoughts ?

Feedback