

# Getting started with Kong through MATLAB

# Outline

- Introduction
- Required files
- Basic commands for Kong
- Parallel FOR loops and tutorial
- Summary

# Kong

- Kong is NJIT's high performance computing service
- Kong gives researchers access to parallel computing which can dramatically increase computing power
- To gain access to Kong you must be on the ARCS researcher list. Please ask your adviser to refer you to ARCS for access.
- [ARCS@njit.edu](mailto:ARCS@njit.edu) – Main email for assistance

# Files required for Kong

- Supported for Matlab versions 2013a and 2014a only
- General information can be found on the HPC wiki site:

<https://wiki.hpc.arcs.njit.edu/index.php>

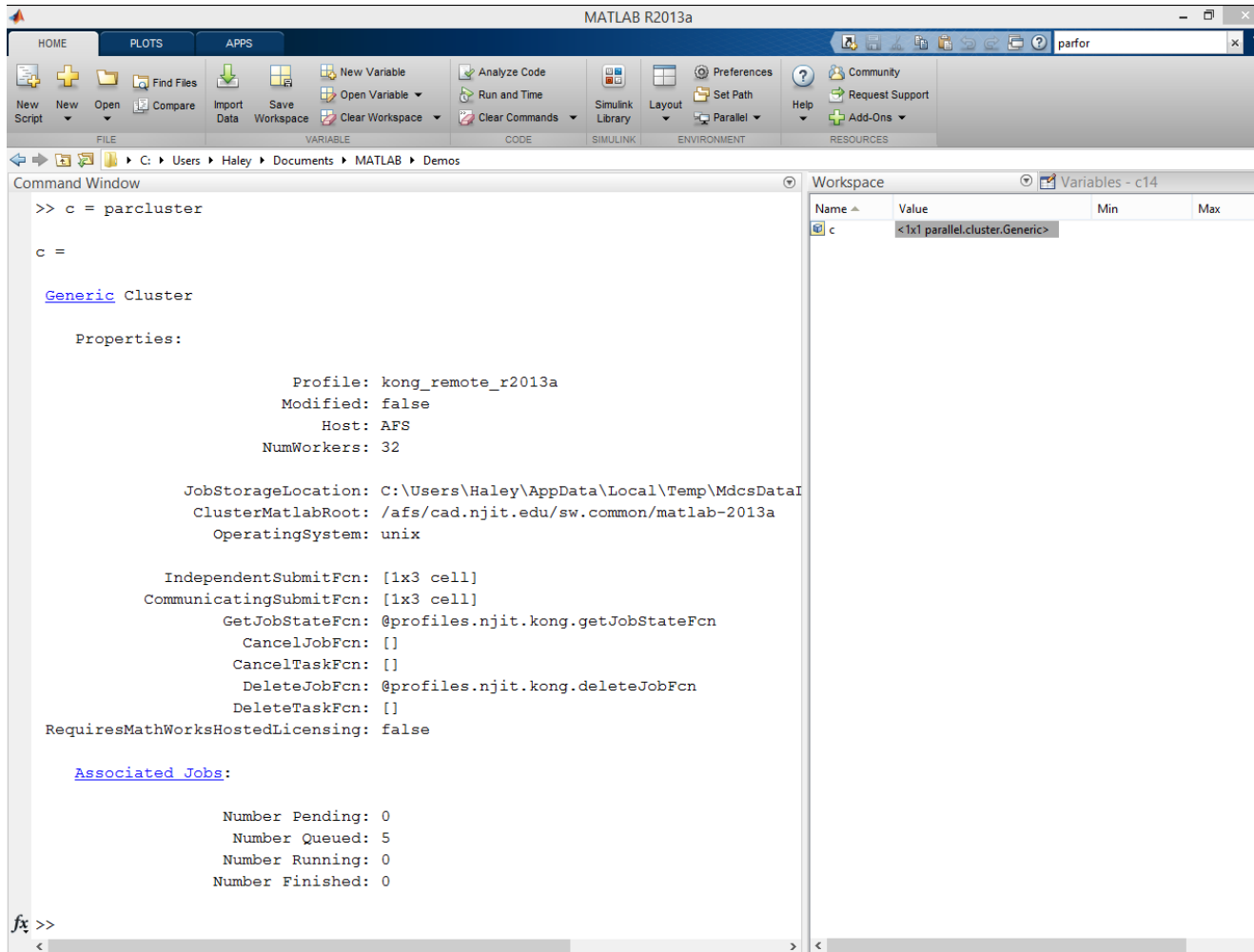
/

[Getting Started with Serial and Parallel MATLAB on Kong and Stheno](#)

- Download the following:
- <http://web.njit.edu/downloads/njit.remote.r2013a.zip>
- Follow the short list of instructions under ‘Configuration’ and ‘Credentials’ to set up your MATLAB to send programs to Kong

# Parcluster

- `c=parcluster;` - shows information about cluster



The image shows the MATLAB R2013a interface. The Command Window displays the output of the command `c = parcluster`. The output shows a `Generic Cluster` object with various properties. The Workspace window shows the variable `c` as a `1x1 parallel.cluster.Generic` object.

```
>> c = parcluster

c =

Generic Cluster

Properties:

    Profile: kong_remote_r2013a
    Modified: false
    Host: AFS
    NumWorkers: 32

    JobStorageLocation: C:\Users\Haley\AppData\Local\Temp\MdcsData1
    ClusterMatlabRoot: /afs/cad.njit.edu/sw.common/matlab-2013a
    OperatingSystem: unix

    IndependentSubmitFcn: [1x3 cell]
    CommunicatingSubmitFcn: [1x3 cell]
    GetJobStateFcn: @profiles.njit.kong.getJobStateFcn
    CancelJobFcn: []
    CancelTaskFcn: []
    DeleteJobFcn: @profiles.njit.kong.deleteJobFcn
    DeleteTaskFcn: []
    RequiresMathWorksHostedLicensing: false

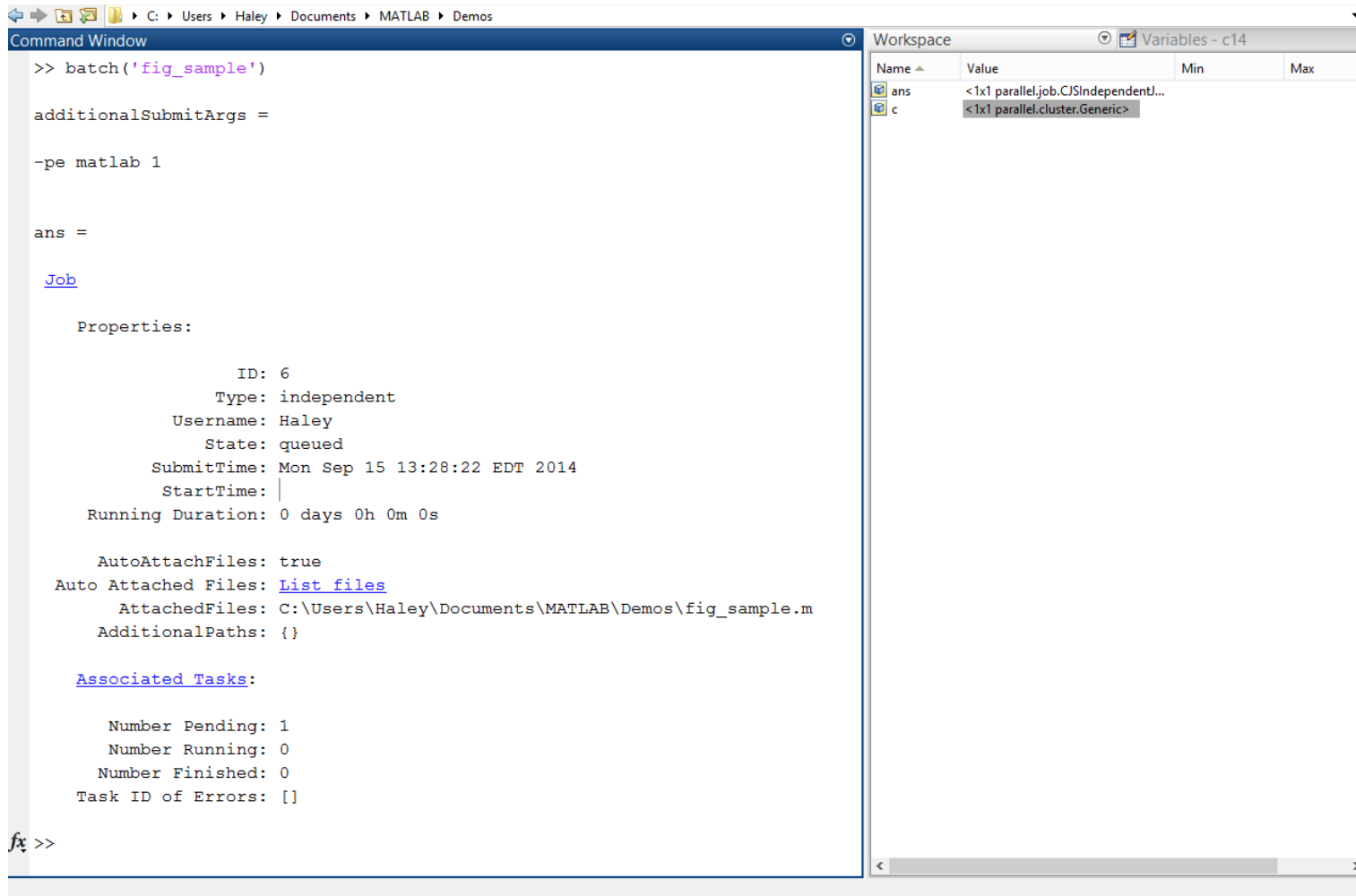
Associated Jobs:

    Number Pending: 0
    Number Queued: 5
    Number Running: 0
    Number Finished: 0
```

| Name | Value                          | Min | Max |
|------|--------------------------------|-----|-----|
| c    | <1x1 parallel.cluster.Generic> |     |     |

# Send a program to Kong

- `batch('filename')` – Sends a file to Kong to be ran



The image shows a MATLAB Command Window and Workspace. The Command Window displays the execution of the `batch('fig_sample')` command, which has returned a `Job` object with various properties. The Workspace shows the variables `ans` and `c`.

```
>> batch('fig_sample')

additionalSubmitArgs =

-pe matlab 1

ans =

Job

Properties:

    ID: 6
    Type: independent
    Username: Haley
    State: queued
    SubmitTime: Mon Sep 15 13:28:22 EDT 2014
    StartTime: |
    Running Duration: 0 days 0h 0m 0s

    AutoAttachFiles: true
    Auto Attached Files: List files
    AttachedFiles: C:\Users\Haley\Documents\MATLAB\Demos\fig_sample.m
    AdditionalPaths: {}

Associated Tasks:

    Number Pending: 1
    Number Running: 0
    Number Finished: 0
    Task ID of Errors: []

fx >>
```

| Name | Value                                 | Min | Max |
|------|---------------------------------------|-----|-----|
| ans  | < 1x1 parallel.job.CJSIndependentT... |     |     |
| c    | < 1x1 parallel.cluster.Generic>       |     |     |

# View Jobs

- `c.Jobs()` – Shows list of jobs sent, allows one to select job

```
C:\Users\Haley\Documents\MATLAB\Demos
Command Window
>> c.Jobs

ans =

5x1 Job array:

      ID      Type      State      FinishTime      Username      Tasks
-----
1      1    independent    queued
2      2    independent    queued
3      3         pool        queued
4      4         pool        queued
5      5    independent    queued

fx >>
```

# View Jobs cont'

The image shows a MATLAB interface with two main windows: Command Window and Workspace.

**Command Window:**

```
>> c2 = c.Jobs (2)

c2 =

Job

Properties:

    ID: 2
    Type: independent
    Username: Haley
    State: queued
    SubmitTime: Mon Sep 15 11:37:43 EDT 2014
    StartTime:
    Running Duration: 0 days 0h 0m 0s

    AutoAttachFiles: true
Auto Attached Files: List files
    AttachedFiles: C:\Users\Haley\Documents\MATLAB\GMTI\ROC_CS.m
    AdditionalPaths: {}

Associated Tasks:

    Number Pending: 1
    Number Running: 0
    Number Finished: 0
    Task ID of Errors: []

fx >>
```

**Workspace:**

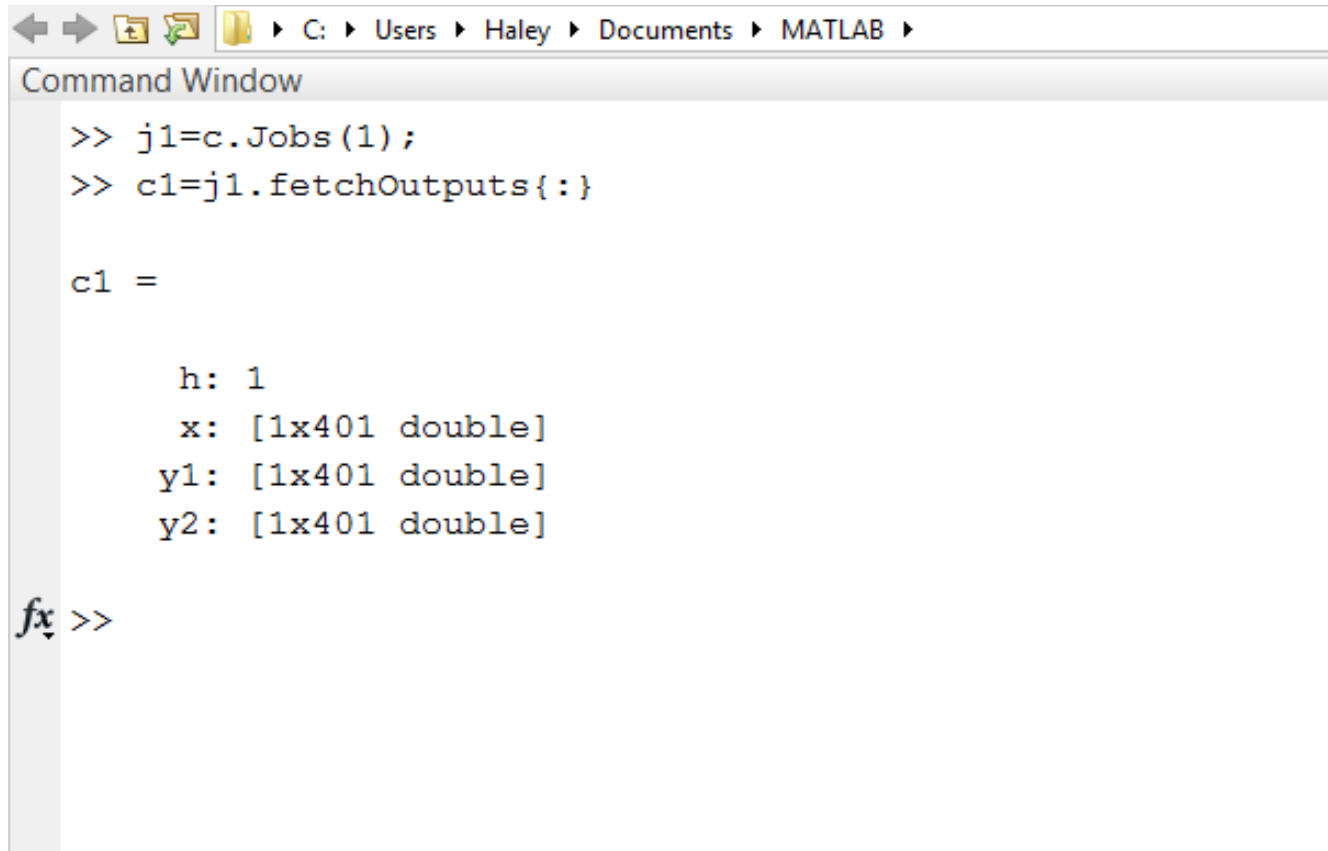
| Name | Value                                 | Min | Max |
|------|---------------------------------------|-----|-----|
| ans  | <1x1 parallel.job.CJSIndependentJ...> |     |     |
| c    | <1x1 parallel.cluster.Generic>        |     |     |
| c2   | <1x1 parallel.job.CJSIndependentJ...> |     |     |

Click and drag to move Command Window...



# Retrieve Results

- `j = (job).fetchOutputs{:}` – Retrieves variables from (job)



```
C:\Users\Haley\Documents\MATLAB
Command Window
>> j1=c.Jobs(1);
>> c1=j1.fetchOutputs{:}

c1 =

    h: 1
    x: [1x401 double]
   y1: [1x401 double]
   y2: [1x401 double]

fx >>
```

# Other Commands

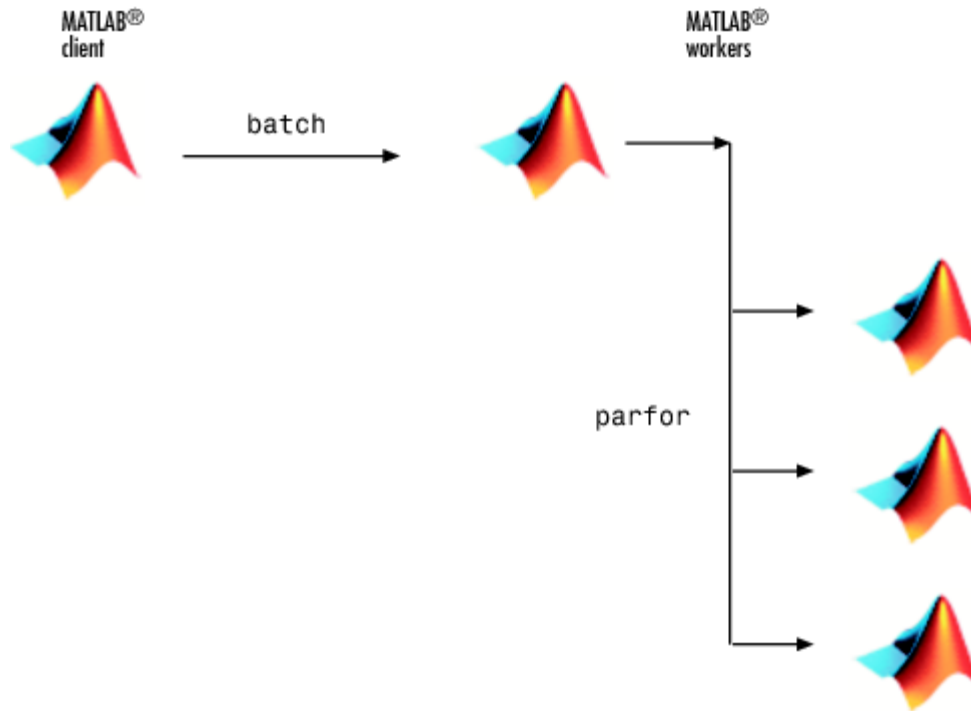
- delete(job) – Deletes the job data
- Delete is needed to prevent your previous data from cluttering. If you do not remove your data using delete Kong may be unable to store your data.
- Additional miscellaneous configurations can be set, and can be found at

<https://wiki.hpc.arcs.njit.edu/index.php>

[/](#)  
[Getting Started with Serial and Parallel MATLAB on](#)  
[Kong and Stheno](#)

# Parallel FOR Loops (PARFOR)

- Allows one to utilize multiple workers

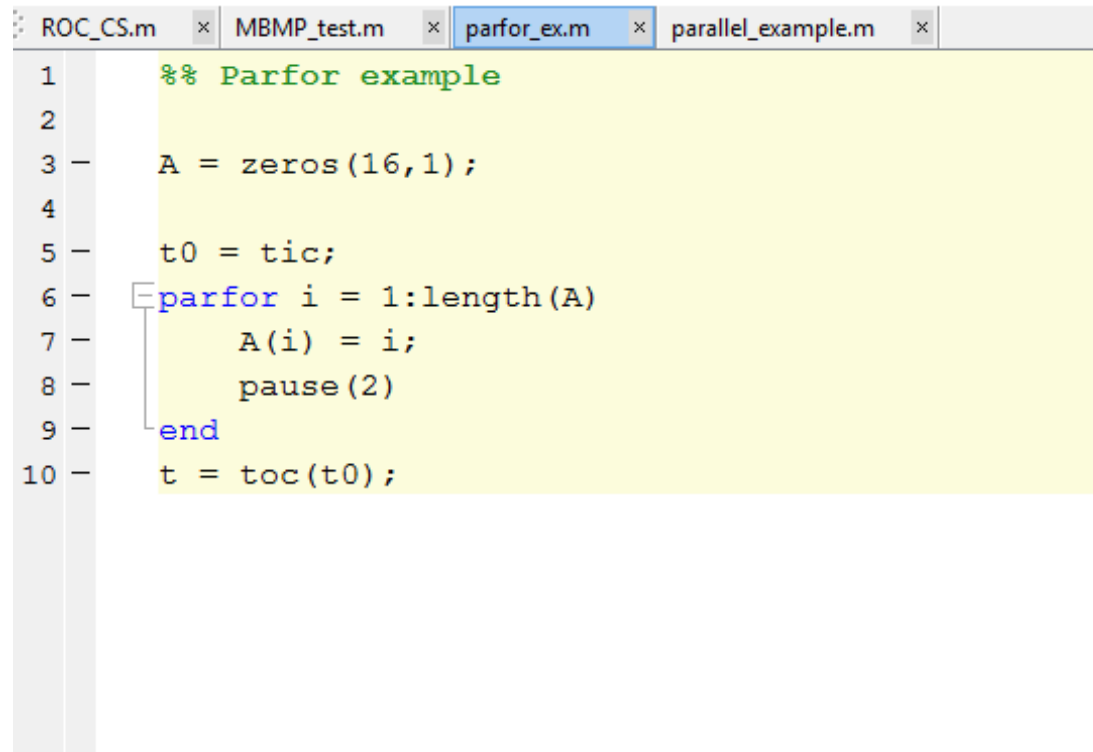


- Gives user ability to save considerable time by computing For loops in parallel

# Batch for Parallel Computing

- To send a job using multiple workers use:
- `batch('filename', 'matlabpool', numworkers)`
- Keep in mind that the command will require `numworkers+1` workers be available.

• Example:



```
ROC_CS.m x MBMP_test.m x parfor_ex.m x parallel_example.m x
1 %% Parfor example
2
3 A = zeros(16,1);
4
5 t0 = tic;
6 parfor i = 1:length(A)
7     A(i) = i;
8     pause(2)
9 end
10 t = toc(t0);
```

# Parfor example

```
Command Window
>> batch('parfor_ex', 'matlabpool', 8);

additionalSubmitArgs =

-pe matlab 9

fx >>
```

# Parfor output

Through PARFOR we get the following

```
Command Window
>> c.Jobs(4).fetchOutputs{:}

ans =

    A: [16x1 double]
    t: 4.6475
    t0: 1410970426915701

fx >> |
```

# Parallel FOR Loops (PARFOR)

Using FOR instead of PARFOR we get

Command Window

```
>> cd Demos  
>> parallel_example
```

```
ans =
```

```
    31.9931
```

```
fx >> |
```

# Summary

- Kong gives researchers access to large computing power
- Parfor allows one to use multiple workers to evaluate a loop in parallel. This is particularly useful for Monte Carlo simulations.
- HPC wiki site:

<https://wiki.hpc.arcs.njit.edu/index.php>

/

[Getting Started with Serial and Parallel MATLAB on Kong and Stheno](#)

- Email [arcs@njit.edu](mailto:arcs@njit.edu) for assistance