MATLAB Distributed Compute on the Cray CS300

Sean Cleveland Ph.D, Ron Merrill Ph.D,
David Schanzenbach M.S.

Information Technology Services
Cyberinfrastructure
University of Hawai'i

https://www.hawaii.edu/its/ci/
uh-hpc-help@lists.hawaii.edu

March 2, 2016

Overview

1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

Outline

1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

MATLAB on the Cluster

- MATLAB Distributed Compute Server (DCS) R2015b is installed on the Cray CS300
- We currently hold a license for 64 *Workers*.
  - One worker is equal to an instance of MATLAB
- In order to use MATLAB on the Cray CS300, you will need to own the MATLAB product with the Parallel Computing Toolbox
- The workers on the Cray CS300 contain all the MATLAB toolboxes
  - **This does not mean you get free access to all toolboxes**
  - **You will only have access to the MATLAB toolboxes you have purchased**

Pre-requisites to using MATLAB on the Cray CS300

In order to use MATLAB on the cluster you must meet the follow requirements

1. Own a copy of MATLAB, with a license for the Parallel Computing Toolbox
2. Have gone through the On-boarding course for the Cray CS300

Outline

1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

Required Files

In order to submit MATLAB jobs to the Cray CS300, a packet of several scripts, and a configuration file are required:

- Download the zip file containing the MATLAB setup files here: http://go.hawaii.edu/YH
- Download should contain:
  - A directory named slurm_matlab, which contains several .m and .sh files
  - A settings file named uhhpc.settings

Setting up MATLAB

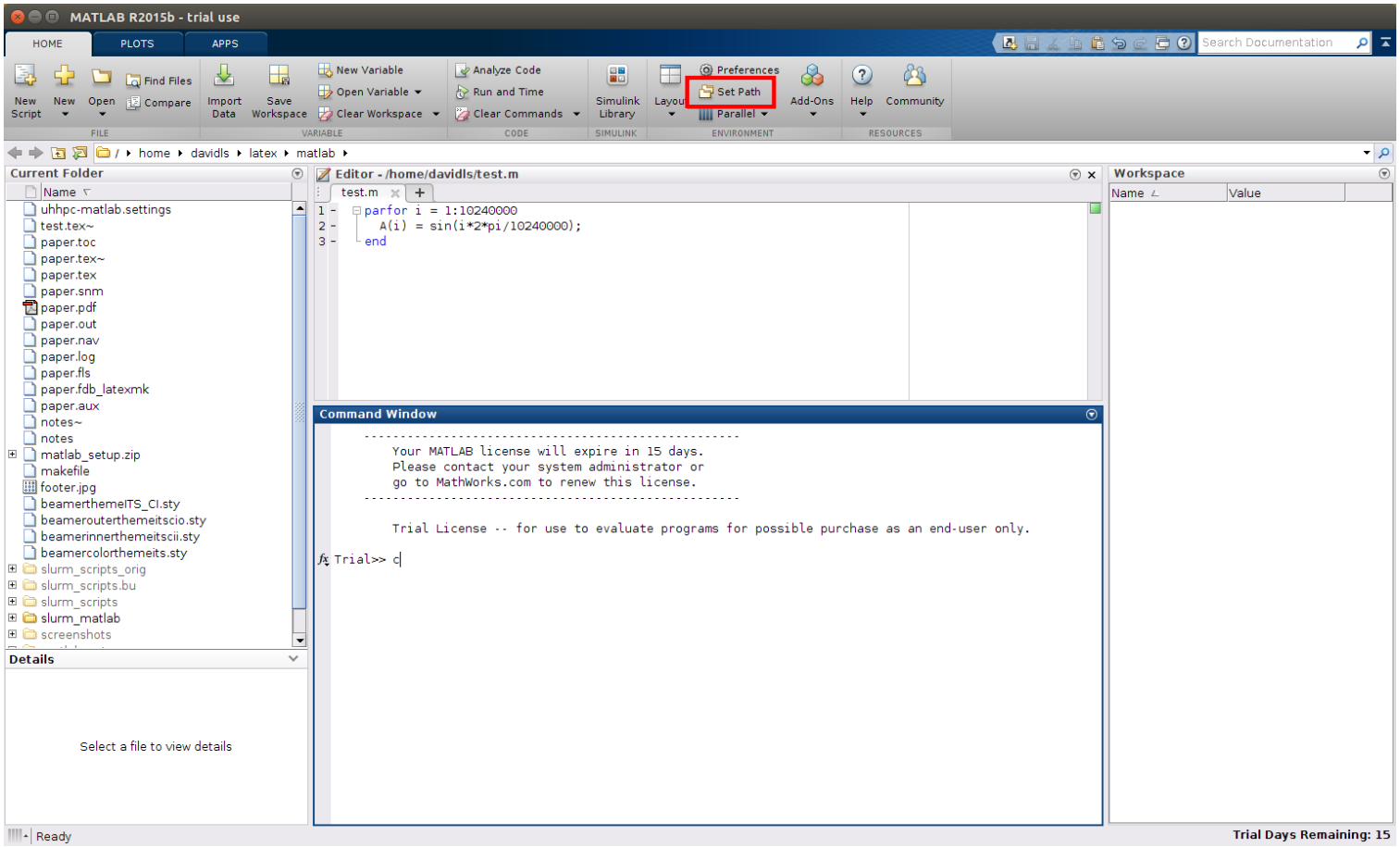
- Unzip setup files
- Move the *slurm_matlab* directory to a location it can reside permanently
  - This directory will be added to your MATLAB path
- Start up MATLAB

The next few Slides will cover what needs to be done in MATLAB to allow communication with the Cray CS300 from MATLAB

Outline

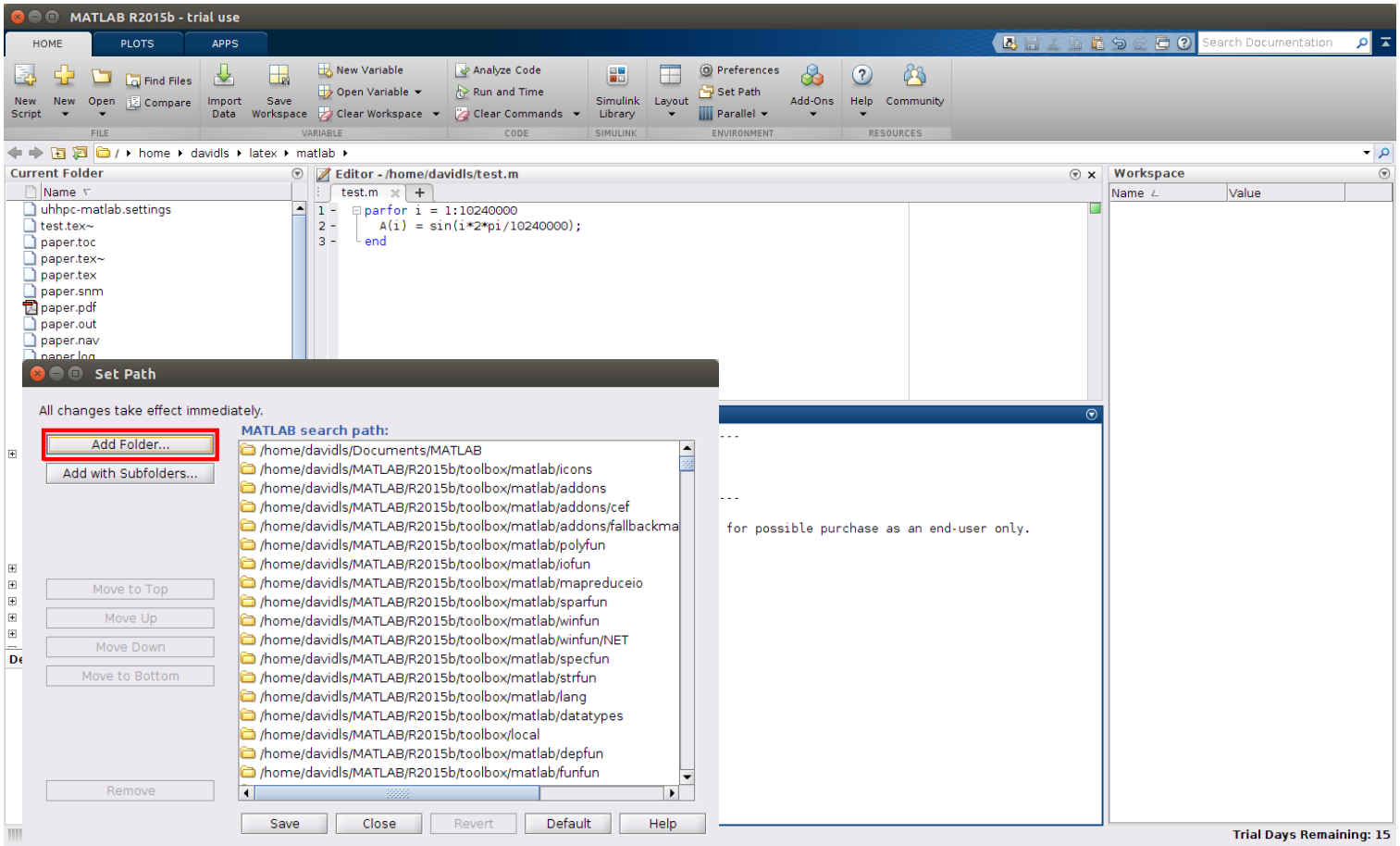
1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

Adding to the MATLAB Path



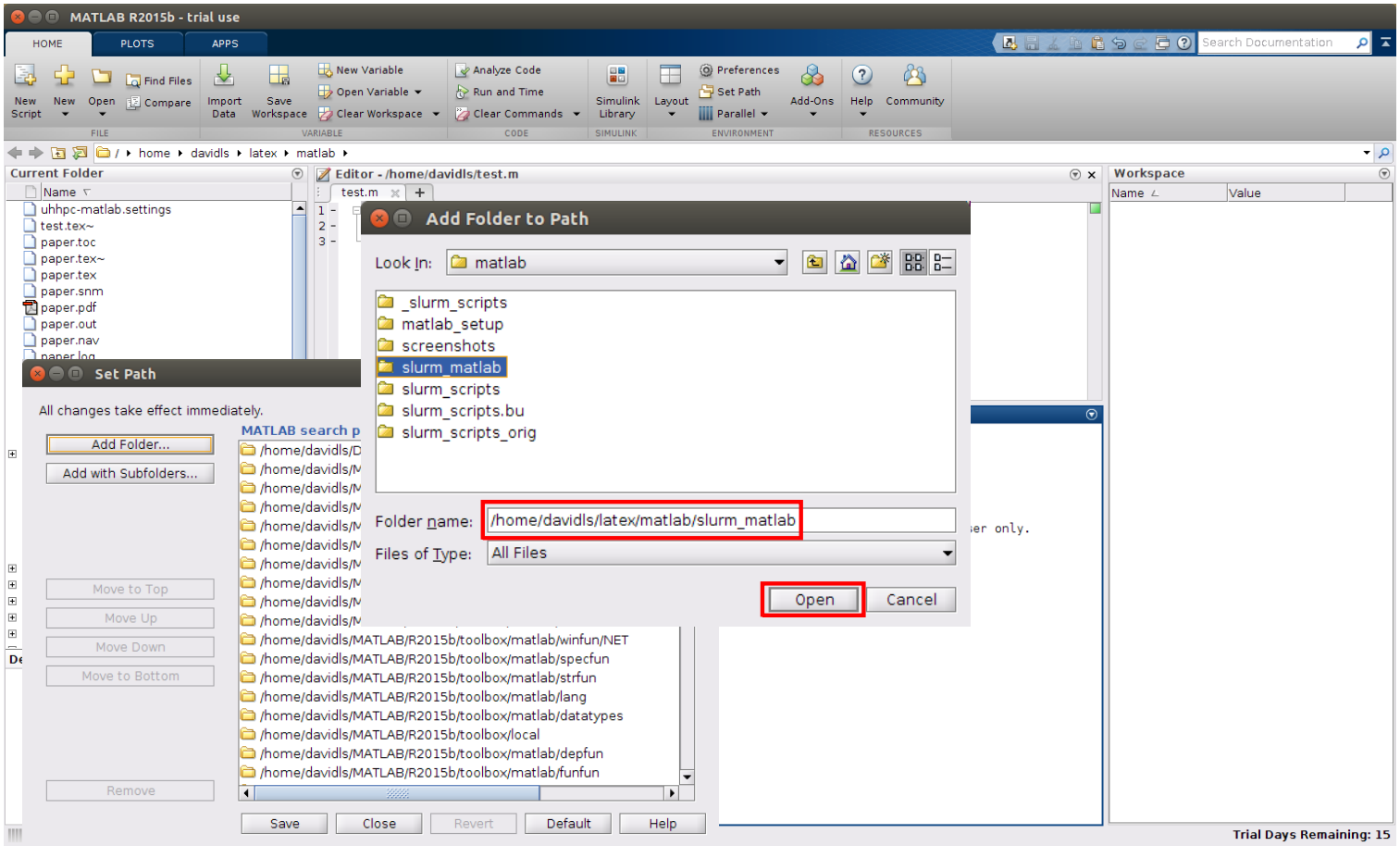
Click the "**Set Path**" button under the "**Home**" tab in the MATLAB UI

Adding to the MATLAB Path



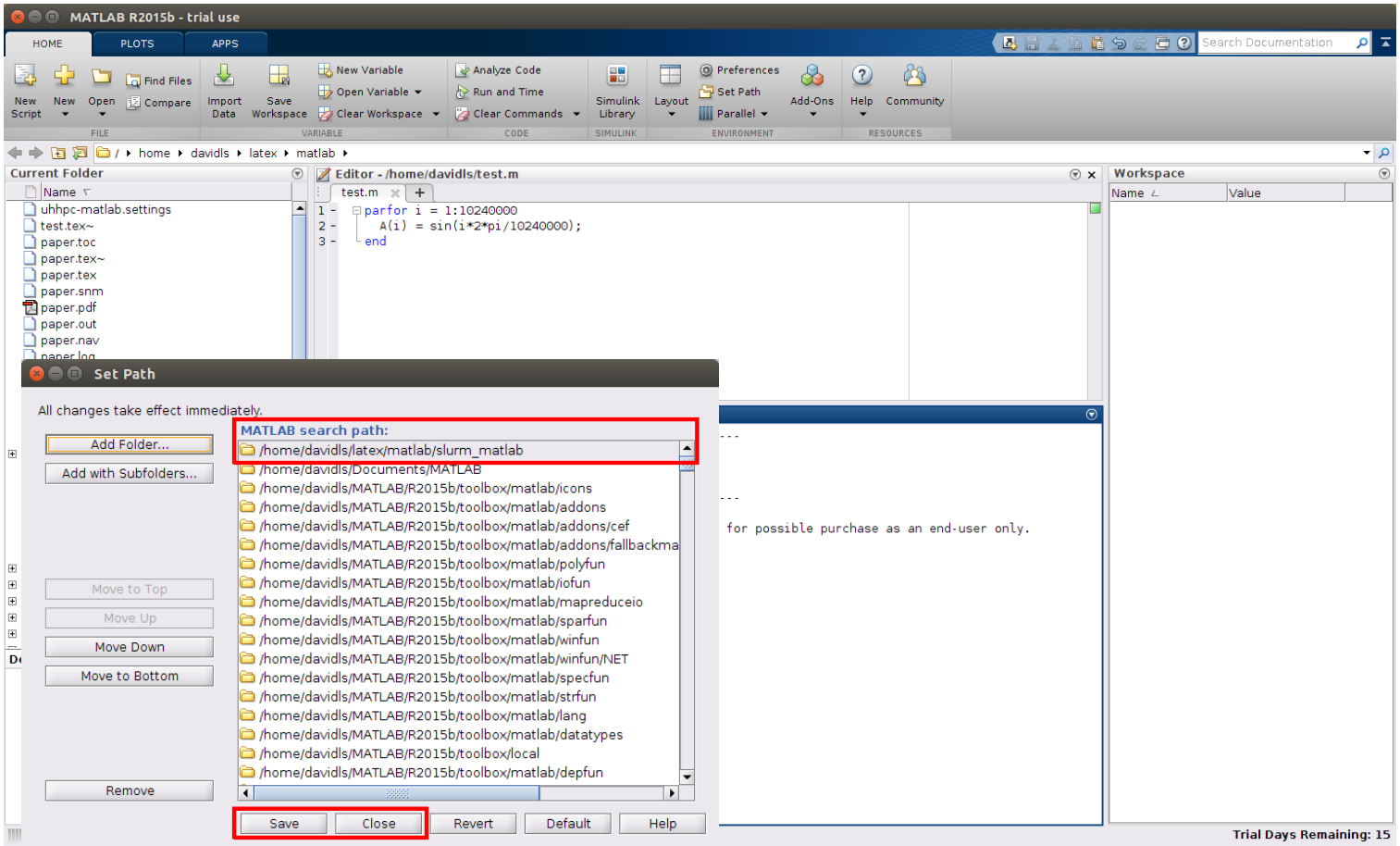
In the Set Path dialogue, click "**Add Folder ...**"

Adding to the MATLAB Path



In the Add Folder to Path dialogue, navigate to the directory the **slurm_matlab** folder resides in and select and **Open** it

Adding to the MATLAB Path

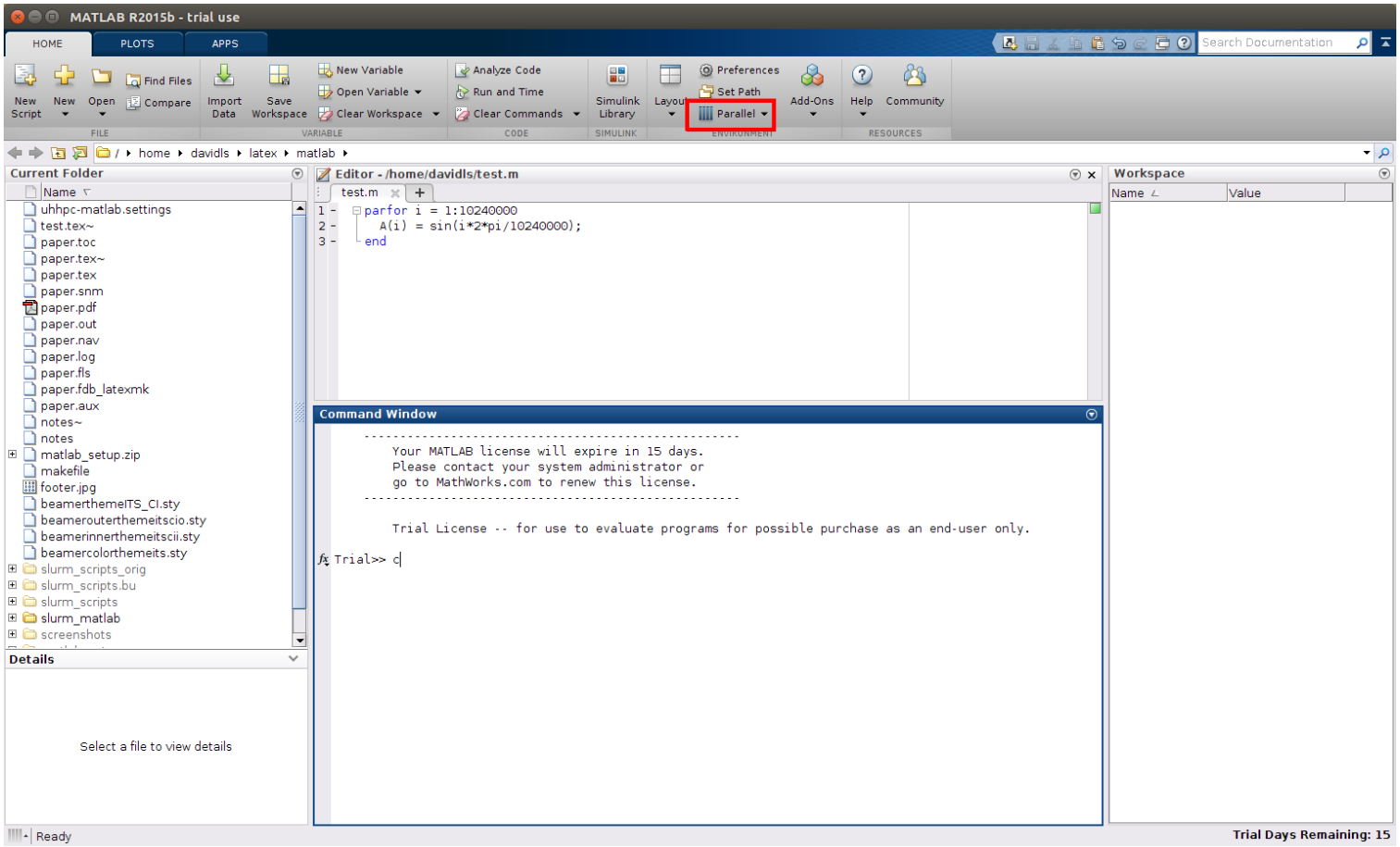


The newly added path should appear in the "**MATLAB search path:**" list. Once it added, go ahead and click "**Save**", then "**Close**"

Outline

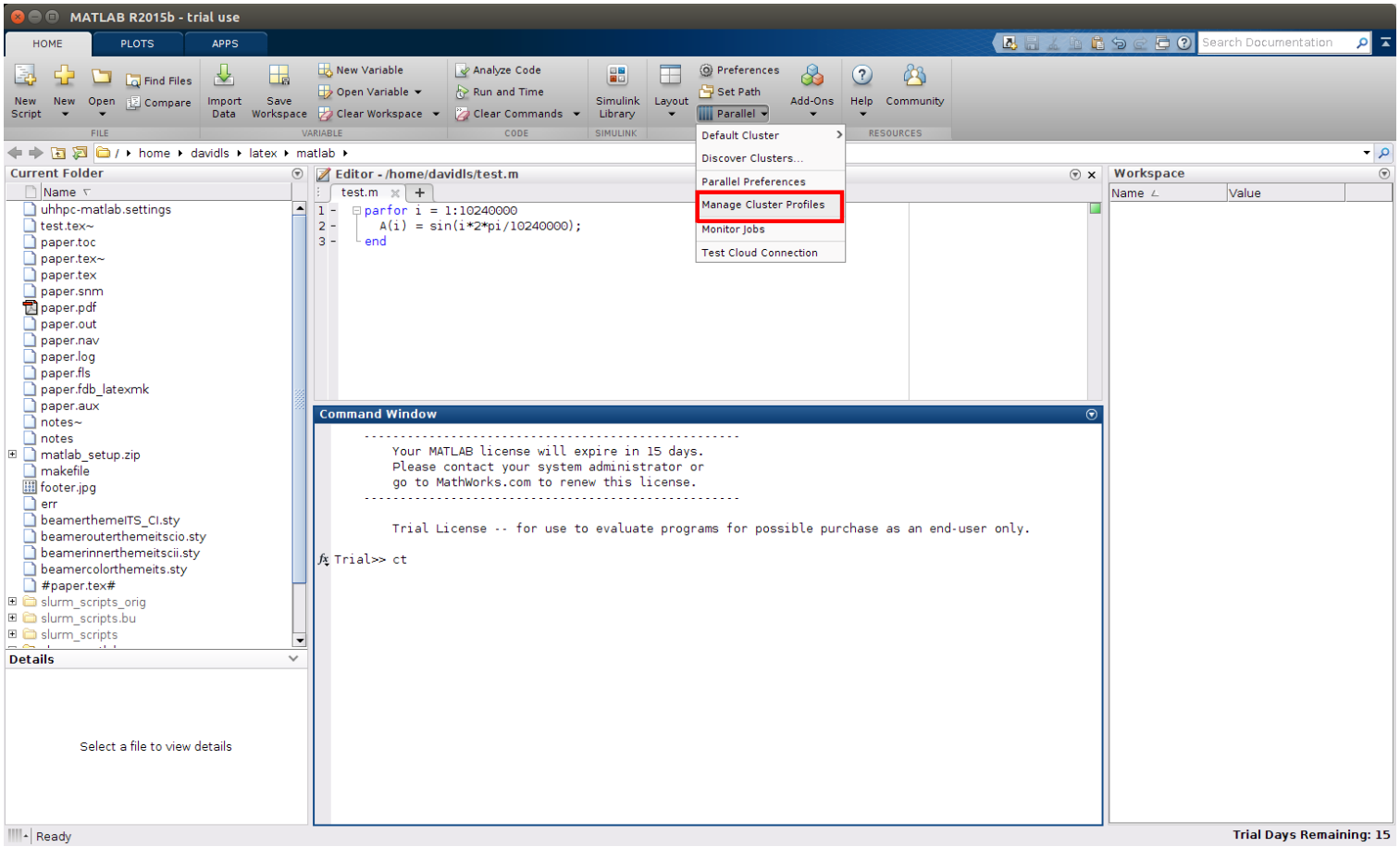
1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

Cluster Profile Setup



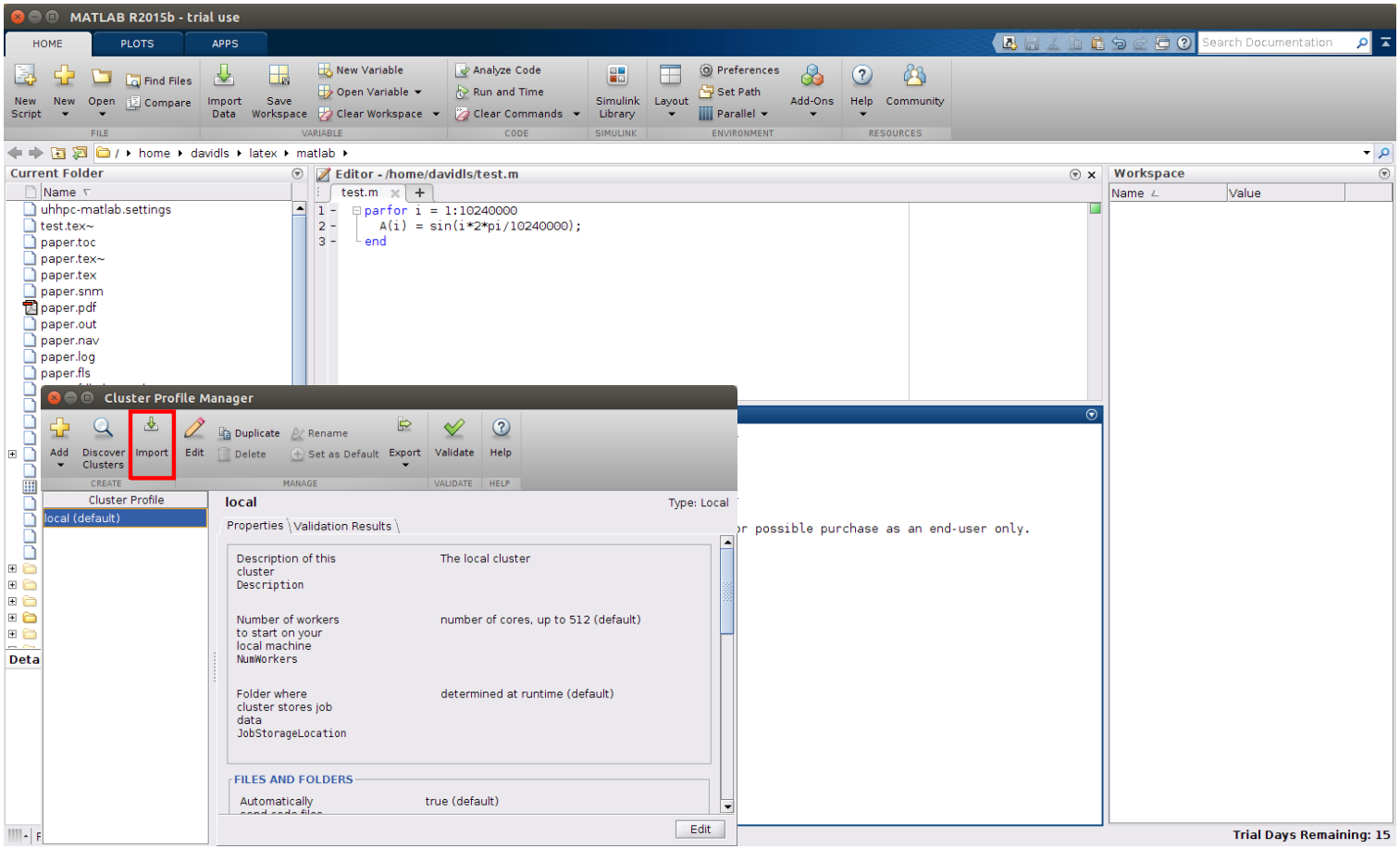
Click the "**Parallel**" button to bring up the Parallel drop down menu

Cluster Profile Setup



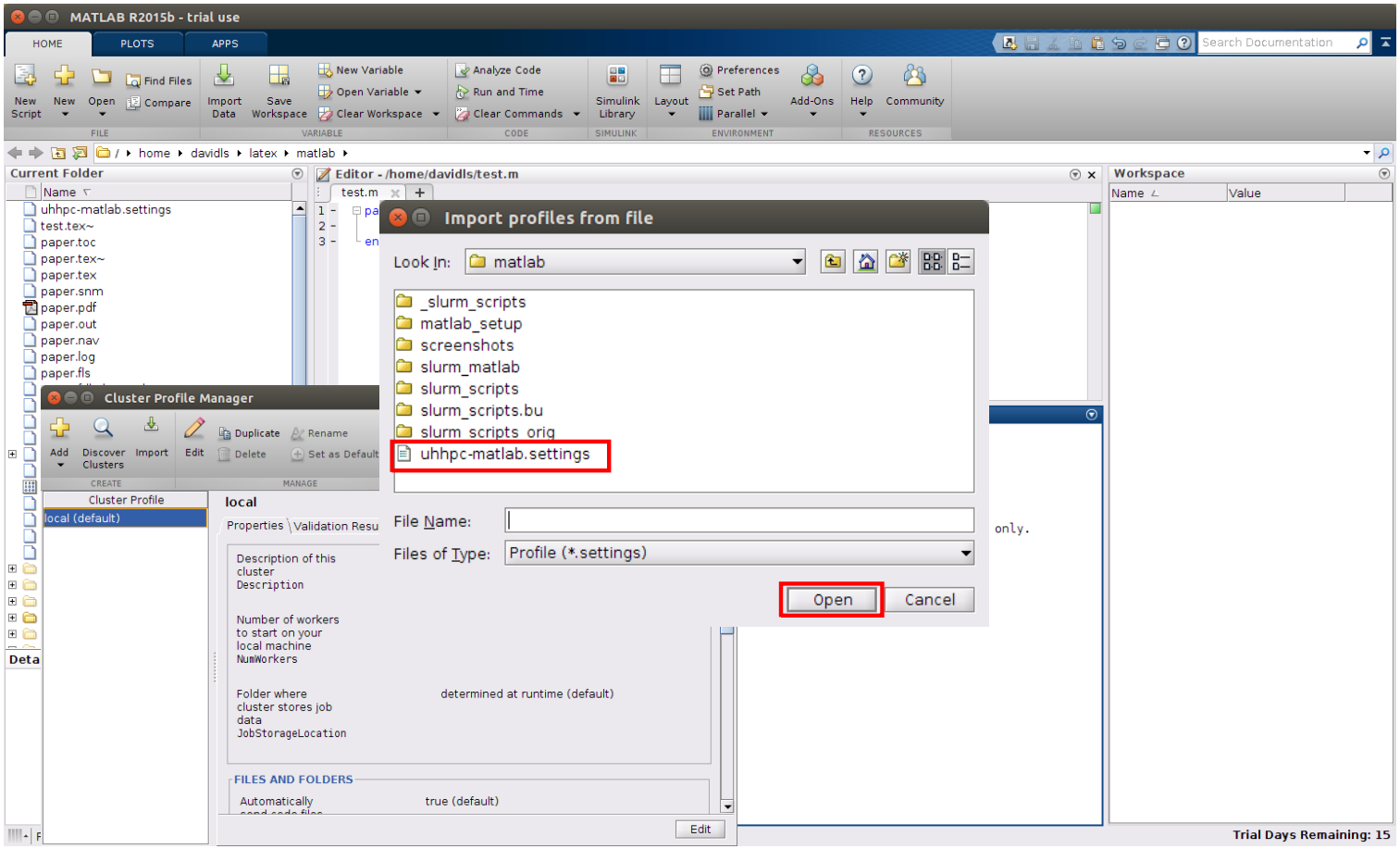
Select **Manage Cluster Profiles** in the Parallel drop down menu

Cluster Profile Setup



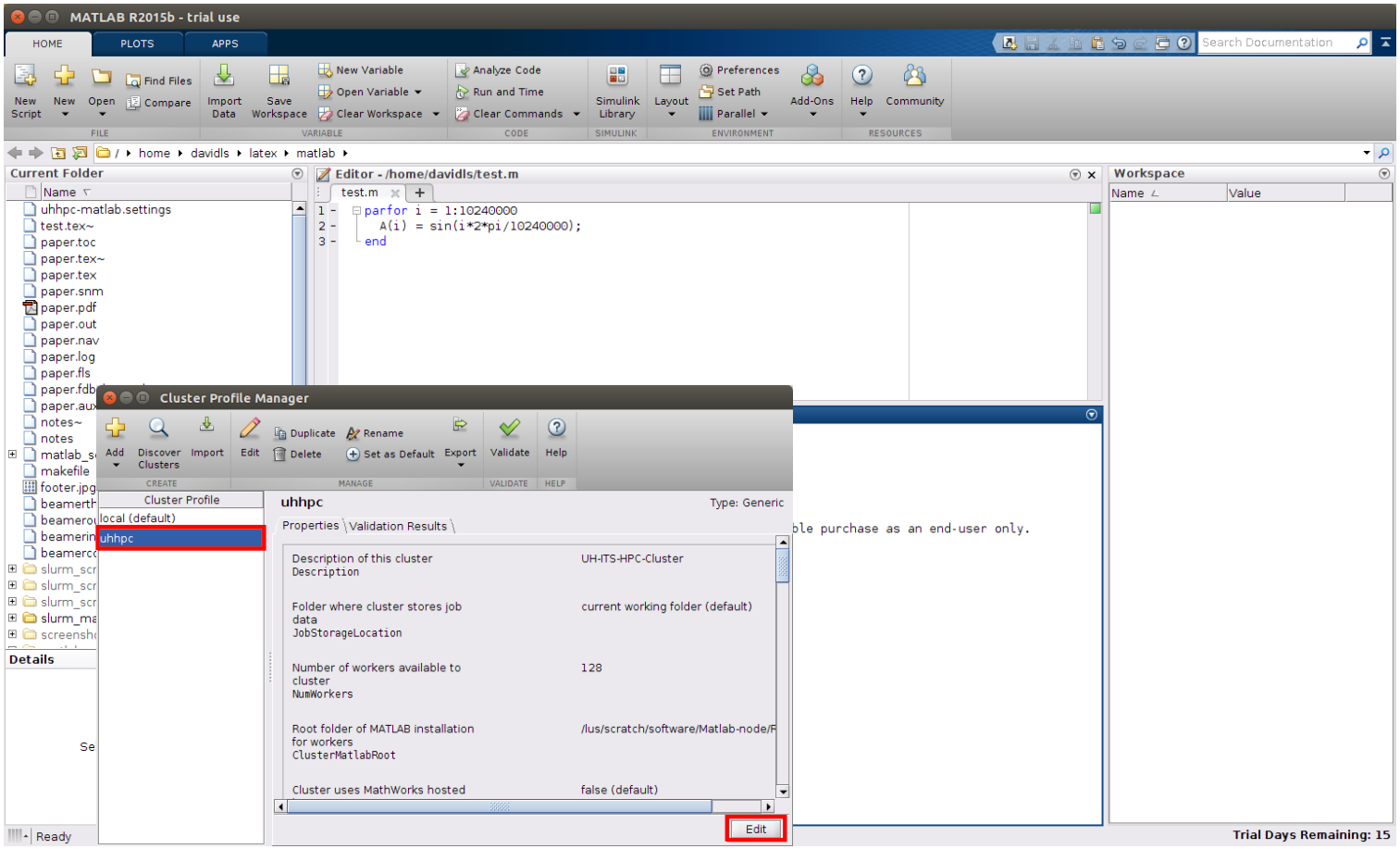
In the **Cluster Profile Manager** dialogue box, click on the "**Import**" icon

Cluster Profile Setup



Navigate to and **Open** the uhhpc-matlab.settings file that was downloaded

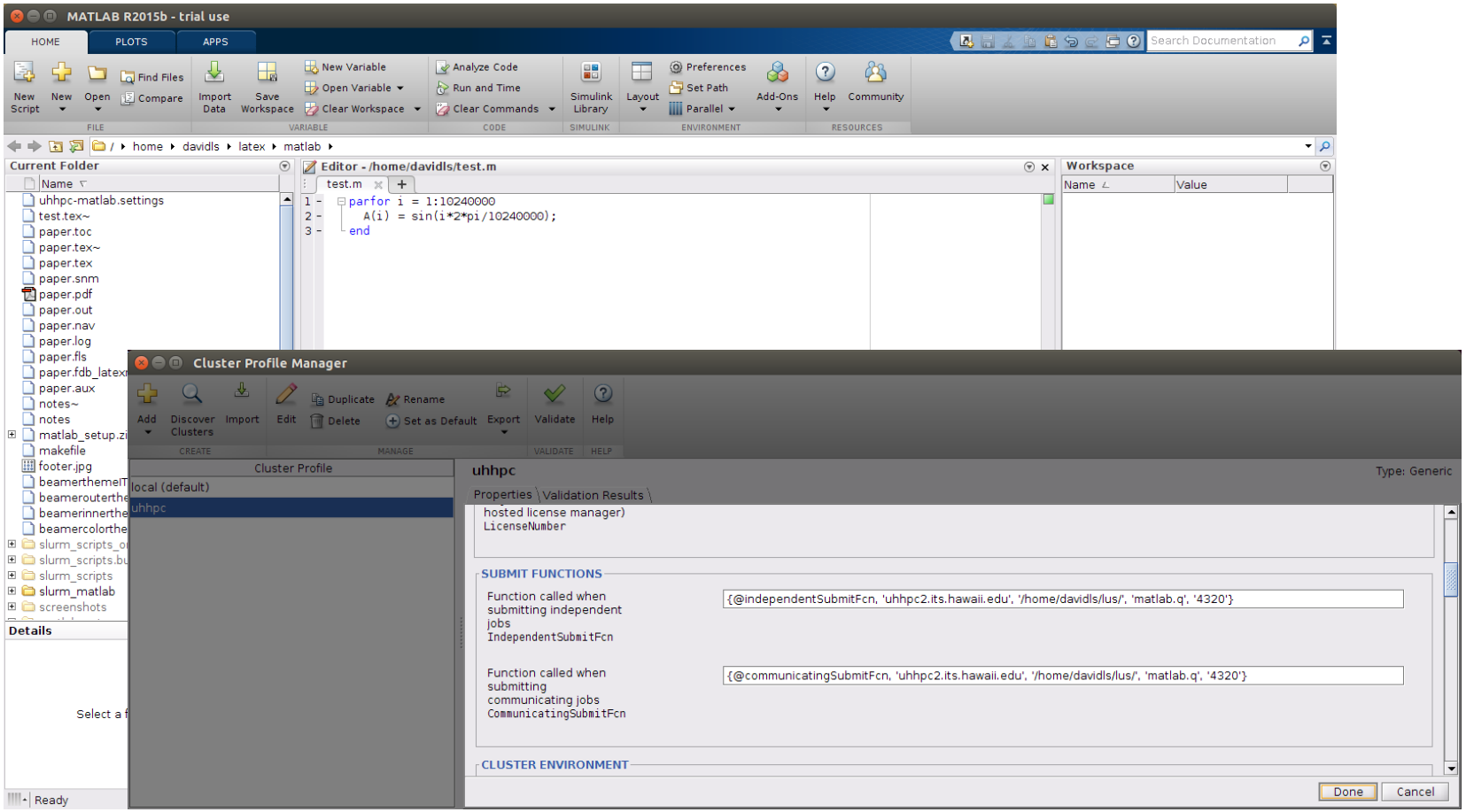
Cluster Profile Setup



Select the **uhhpc** cluster profile, and press the "**Edit**" button.

For ease of editing, you may want to increase the window size

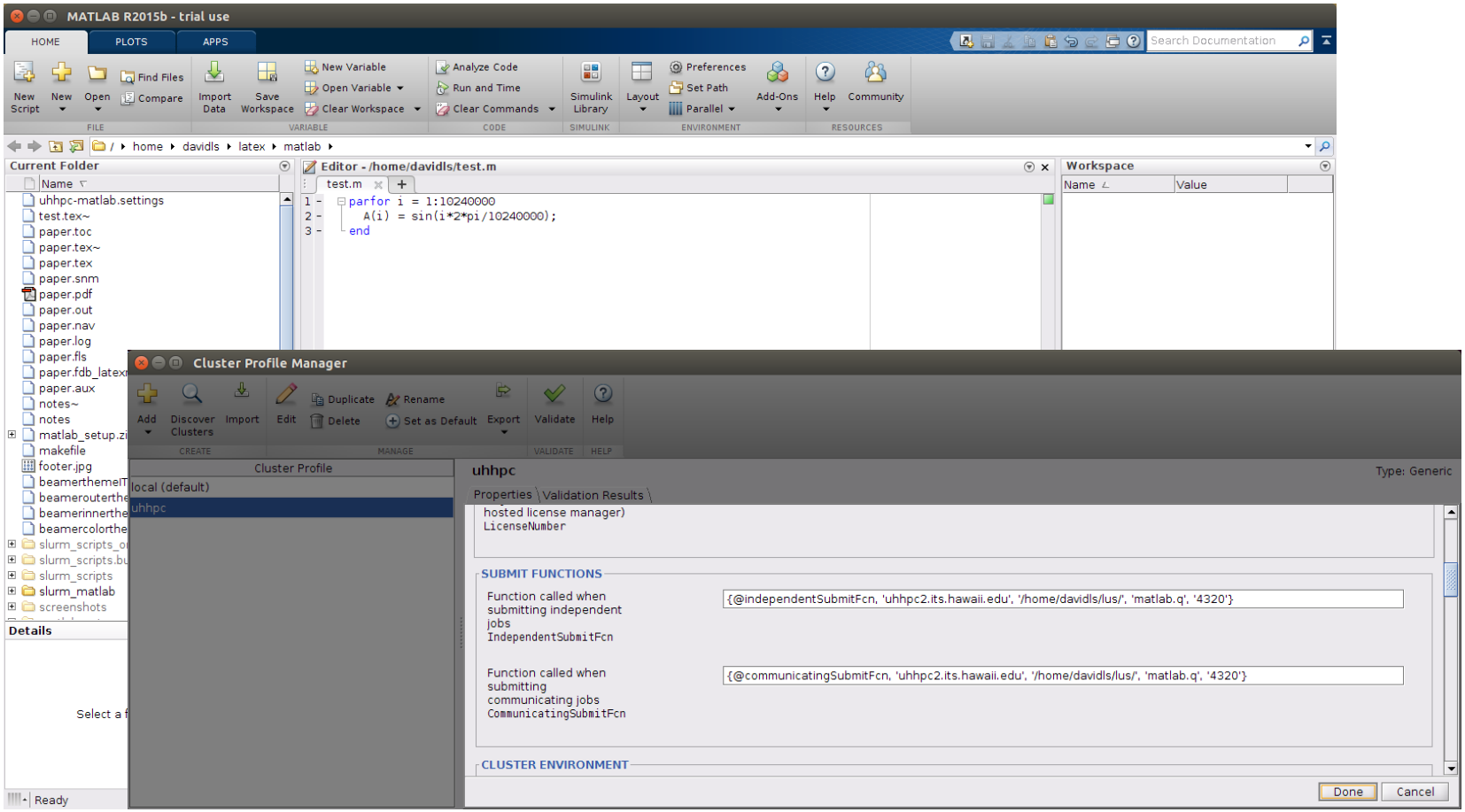
Cluster Profile Setup



Each of the functions in the **SUBMIT FUNCTIONS** section, have the follow format:

{<**func**>,'<**server addr**>', '<**remote storage dir**>', '<**partition**>', '<**timelimit**>', [<**mem usage in MB**>] }

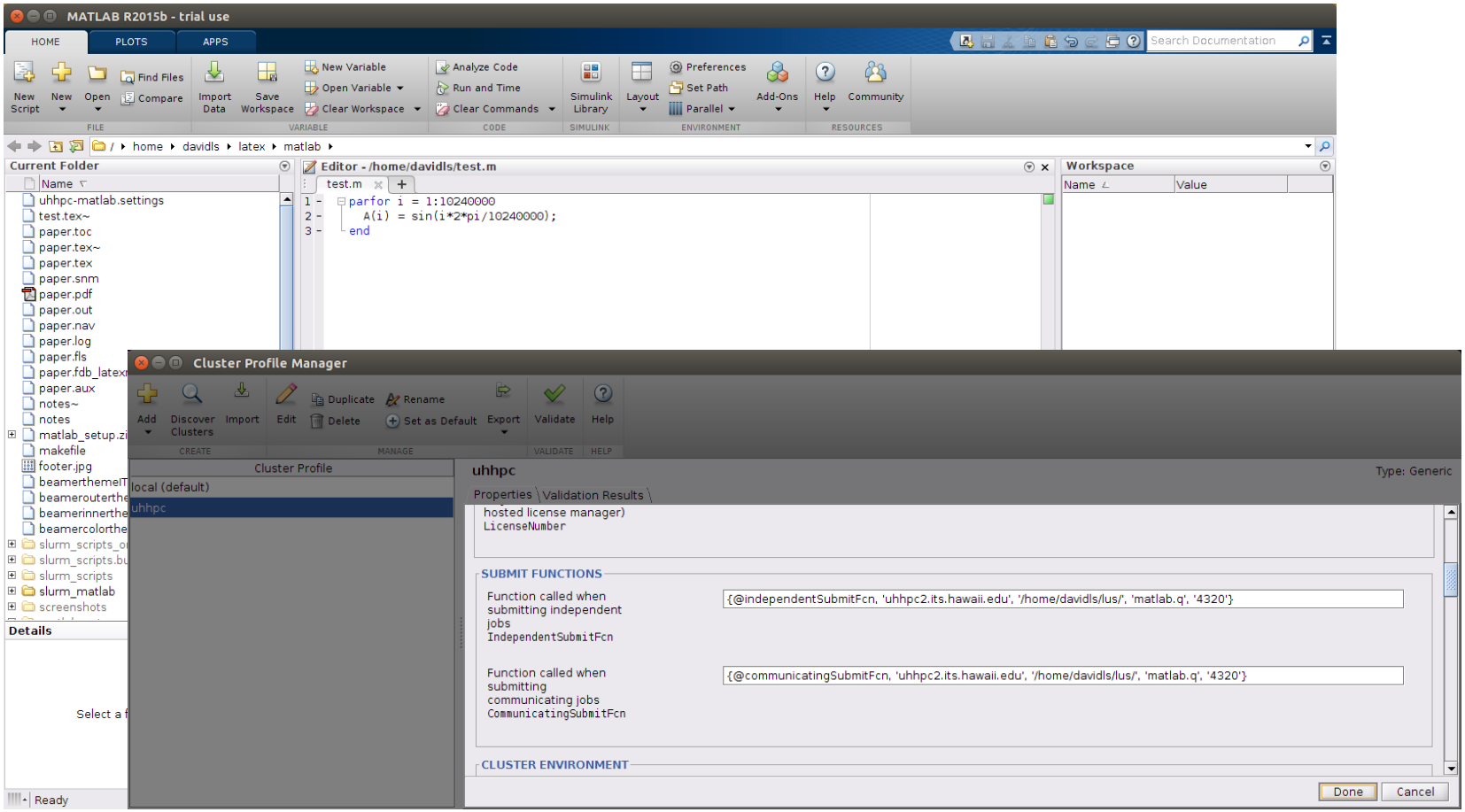
Cluster Profile Setup



{'<**func**>','<**server addr**>', '<**remote storage dir**>', '<**partition**>', '<**timelimit**>', [<**mem usage in MB**>] }

Replace the remote storage dir with the path that matches your users lus folder location i.e., /home/<username>/lus/

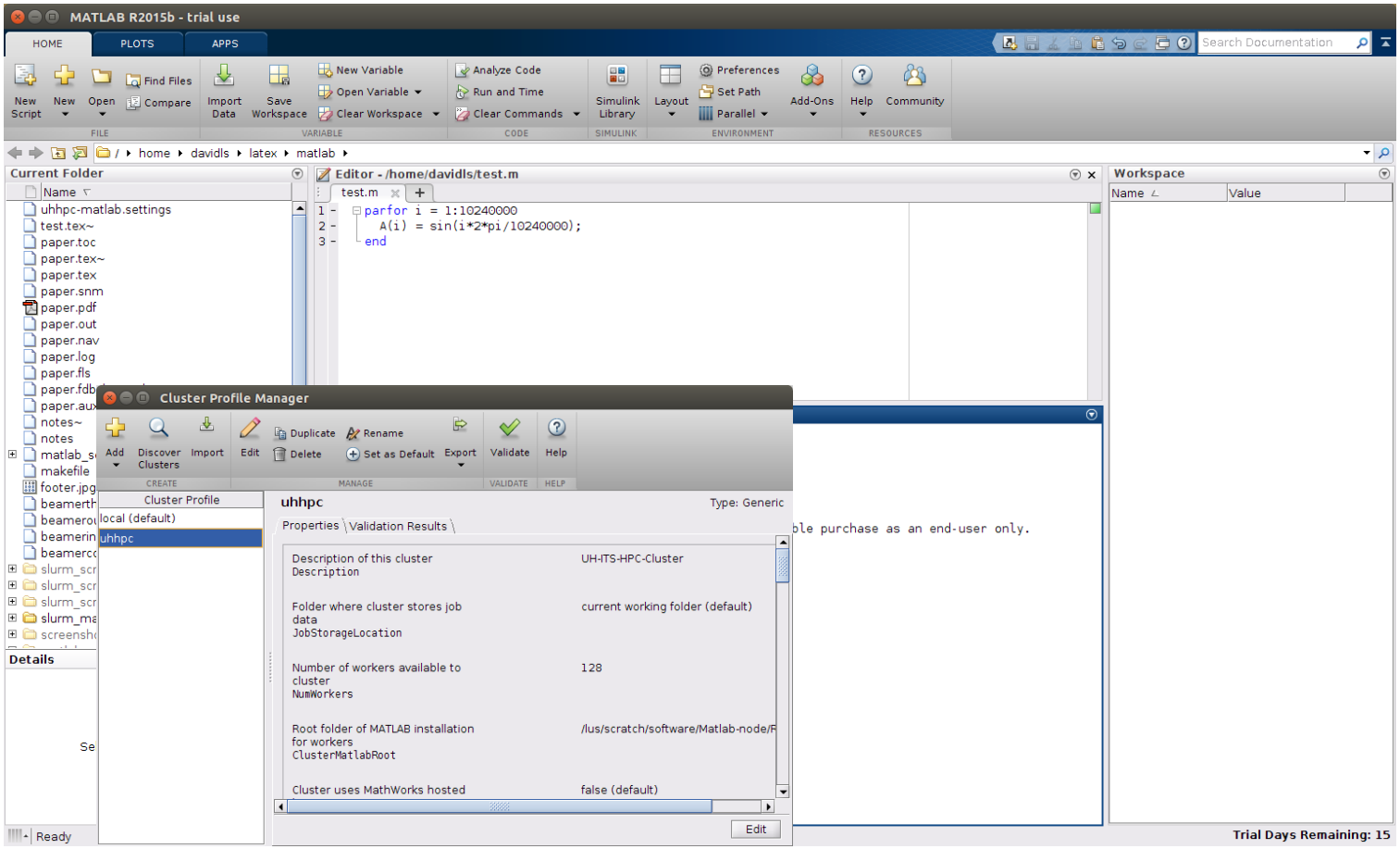
Cluster Profile Setup



{'<**func**>','<**server addr**>', '<**remote storage dir**>', '<**partition**>', '<**timelimit**>', [<**mem usage in MB**>] }

Adjust the slurm partition and slurm timelimit to fit the partition you wish to run on, and finally click **done**

Cluster Profile Setup



The profile should now be configured for your user on the Cray CS300

Cluster Profile

- To simplify the use of MATLAB on the cluster, we suggest setting the UHHPC cluster profile as the default profile
  - Many of the parallel toolbox commands will use the default cluster when an alternate cluster object is not provided
- If the UHHPC cluster profile is not set as the default, the cluster profile name, **'uhhpc'**, must be provided when creating the cluster object
- If you plan to change partitions or run in multiple partitions on the Cray CS300, you will need to modify the existing profile, or create duplicate profiles with the required changes to the remote storage location, partition and timelimit

Outline

1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

Outline

1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

Commands

- pctconfig – Configure settings for Parallel Computing Toolbox client session
  - Is not persistent
  - Must be set before any interaction with the parallel toolbox occurs
  - Used to set the correct 'hostname' in MATLAB
    e.g., > pctconfig('hostname', '192.168.0.1')
    - A valid 'hostname' is required for validation to pass as well as for many of the parallel functions to work
- parcluster – Creates a cluster object which can be passed to functions used to create parallel tasks
- gcp – Get current parallel pool
- addAttachedFiles – Attach files or folders to parallel pool

Commands

- batch – Run MATLAB script or function on worker
  - Non-interactive. MATLAB can be closed
- parpool – Create parallel pool on cluster
- spmd – Execute code in parallel on workers of parallel pool

For more information and other commands, visit the **Distributed toolbox help manual** on the MathWorks website

Outline

1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

Simple Tests

- Simplest test is to use "*validate*" in the **Cluster Profile Manager**
- Before testhing, make sure to reduce the value for "Number of workers available to cluster" (NumWorkers) value to something lower than 64
  - Typically we test with 2 – 10 workers
- NumWorkers must be lowered since validate uses this as the number of workers to request
- You also must make sure the hostname in MATLAB is set correctly (using pctconf), or the parpool test may fail
- Remember to reset the NumWorkers value back to 64 or you may get errors saying not enough workers when requesting more than the current NumWorkers value

Outline

1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

Demo

Live Demo . . .

Outline

1. MATLAB Distributed Compute
2. Configuration
   - Setting up the MATLAB path
   - Importing the Cray CS300 MATLAB profile
3. Basic Commands & Testing
   - Commands
   - Validation
4. Demo
5. Resources

Resources

- Newest revision of these slides – `http://go.hawaii.edu/H5`
- Parallel MATLAB (PCT and DCS) @ Harvard
  - Not all sections pertain to our cluster, but some sections provide some information on running jobs with the parallel toolbox
  - The "Submitting DCS jobs from within MATLAB" section covers using the batch command to submit parallel jobs to a cluster

Questions?