Euler - for deconvolution and other computations

Date: 27/08/2024

# 0. Before we start:

## 0.1 What is used in this document

* GitHub: Git is a software to share program code easily. Using GitHub, we can share and use the same code without much effort
* Euler: The high-performance computing cluster at the ETH. Allows us to do the decon without needing expensive computers ourselves – but might take a while to get used to
* Terminal: The Euler is a cluster, not a website. It is only accessible via a terminal. This means there is little to no visualization, and instead of clicking on folders and icons you access folders and run programs with written commands in the command-line. The terminal is pre-installed on Linux and Mac, but on Windows devices you need a programme for that. Please see the***get ready section 0.7*** below

## 0.2 Where can I find:

* The decon code: <https://github.com/kleele-lab/iSIM>
* The Euler tutorials: <https://scicomp.ethz.ch/wiki/Tutorials>
* A nice overview of git commands: <https://education.github.com/git-cheat-sheet-education.pdf>
* A cheat sheet of Linux commands: <https://www.reddit.com/r/linux/comments/9rns12/some_linux_commands_cheatsheet/>

## 0.3 The most important linux commands:

Show all files in the directory: *ls*

Show all files in the directory including hidden ones: *ls -a*

Move into a directory: *cd name\_of\_directory*

Move out of a directory: *cd ..*

Create a new folder: *mkdir name\_of\_folder*

Display content of file: *cat name\_of\_file*

Delete file: *rm name\_of\_file*

## 0.5 The most important Euler commands:

Run a programm: *sbatch name\_of\_program*

Check if you have any running jobs: *squeue*

Stop a running job (jobid can be found via squeue): *scancel jobid*

## 0.6 Some convenient Tipps:

If you write something (file-name, command, etc.) and press *tab* the command line will try to autocomplete it (Example: You have only one file named “slurm-9387845.out” - write “slu” and press tab for autocomplete)

The asterisk (\*) is a replacement for all other letters, very useful for removing multiple files with similar names:  
Example: You want to remove all files starting with “slurm” --> command:

## 0.7 Get ready

If you have a **Mac** or **Linux** machine, you simply have to make sure you are either in an ETH Wi-Fi or connected to the VPN and open the terminal.

If you have a windows machine, you need an extra step to download and set up a terminal program. This is due to Windows not providing an SSH client (a secure connection protocol). However, this should be necessary only once.

Download one of the recommended programs: MobaXterm, PuTTY or Cygwin.  
In MobaXterm and PuTTY, you need to login a bit differently (but just as simply). See the visual explanation here: <https://scicomp.ethz.ch/wiki/Accessing_the_clusters#Windows>

# 1. Login to Euler

In case you have a normal terminal, access the cluster with the following command:

ssh [username@euler.ethz.ch](mailto:username@euler.ethz.ch) – ETH Password (same as your e-mail)

In mobaXterm, you gave the host name already:

Name: [username](mailto:username@euler.ethz.ch) Password: ETH Password (same as your e-mail)

**ALWAYS AFTER LOGIN:**

**Set your id to the lab (gives you access to stronger ressources & GPU):**

*newgrp biol-ibc-kleele*

# 2. First time setup

### Download the Decon code

Clone the Github repository:

With this command, we are downloading the decon code from GitHub. (If you want to contribute and change the code, you must establish a ssh connection to your GitHub (there a tutorials on GitHub))

git clone <https://github.com/kleele-lab/iSIM.git>

Now access the new folder called iSIM:

cd iSIM/

**Install additional packages:**

Our decon requires specific programs that are not on the Euler. Flowdec provides the actual Richardson-Lucy deconvolution. The other two are for reading and writing metadata. Before installing, we have to enter the same Python environment we are using in the decon (load modules).

module load stack/2024-06

module load gcc/8.2.0

module load python\_gpu/3.11.2

python -m pip install flowdec

python -m pip install xmltodict

python -m pip install dicttoxml

# 3. Run the decon

Make sure you are in the folder of the decon software. If you just logged in to the Euler, you probably need to enter the iSIM folder via

*ls iSIM*

Our IT connected one of our NAS servers to the Euler. The connected server is kleele\_2. Make sure you have your files there

/nfs/nas22/fs2202/biol\_bc\_kleele\_2

If you check this folder via the ls command like this:

*ls* /nfs/nas22/fs2202/biol\_bc\_kleele\_2

You should see all the folder named after the lab members

## 3.1 Run the deconvolution scripts

### 3.1.1 Run a folder of images:

Using the *sbatch* command, we send the deconvolution program to Euler. Additionally, we give the location of the folder we want to decon.

*sbatch run\_folder\_decon.sh /nfs/nas22/fs2202/biol\_bc\_kleele\_2/path\_to\_folder*

***CAVEAT: This searches all the folders in the directory “path\_to\_folder” for images with the ending “.ome.tif”***

### 3.1.2 Run a single element:

If you have only a single image you want to deconvolve, you can send it directly to the system with the second script, *run\_image\_decon*.

*sbatch run\_image\_decon.sh /nfs/nas22/fs2202/biol\_bc\_kleele\_2/path\_to\_image/image.ome.tif*

***CAVEAT: Here, you have to give the explicit image ending with “.ome.tif”***

## 3.2 Check the deconvolution results:

When you start the deconvolution, you should see a text reading “Submitted batch job id\_number”, where id\_number is a multiple digit number. Those number are given to every job submitted on the whole Euler in an ascending order.

This output might look confusing at first but it will show the list of images to deconvolve at the beginning and updates on the decon itself. In case something went wrong, you should see some kind of ‘error’ message at the end of the file.

To see the output of the deconvolution, check the ouput file via:

*cat slurm-{id\_number}.out*

Example: cat slurm-618198.out

### 3.2.1 Recommendations for controlling the decon

* Do NOT check the result of your decon by simply opening one of the folders and looking for the deconvolved image. You might miss things that went wrong or, because the process isn’t finished yet, wrongfully think that something went wrong
* From time to time, please delete your old slurm-files. This avoids your storage getting very messy, and you can use the shortcut using ‘tab’ when opening a slurm file without having to remember which one it is.

## 4. Update the git

### 4.1 Check for updates

If you check for updates on the git, you want to see if someone else updated the code which you then will have to download (=pull) to your folder.   
To check if there is anything online that you need, run

*git status*

It should now tell you that there is code, or there is nothing new (“clean”).   
Now, we want to download the changes. Use

*git pull*

To download the new code (pull it from the server). You can also do this if you are not sure if there are changes. You cannot break something (as far as I know and have experienced).

### 4.1.1 Additional commands that might be handy

If, for some reason, you want to download changes from the server to your local folder without overwriting the current folder, one can use

*git fetch*

As discussed in the team meeting presentation, it is possible to create multiple versions of the code (=branches), to work or include different features or make the code ready for different environments. For example, it would be possible to create a version to be run on Euler and one to be run on local PCs with a GPU.  
To see which branches exist, run the following command

*git branch*

To change your branch location to another one named branch\_name, do

*git checkout branch\_name*

You are now on the specified new branch. That means that your code has been changed most likely.

# 5. Miscellanea

### 5.1 GPU issues

If the system fails due to GPU memory issues, this can be increased in the *run\_folder\_decon.sh /* *run\_image\_decon.sh* scripts.  
Change at the top in the line “#SBATCH –gres=gpumem:16G” to the necessary memory amount. Please check here (<https://scicomp.ethz.ch/wiki/Getting_started_with_GPUs>), which memory sizes are available, the system is running on single GPU and has not been tested for multiple ones.

### 5.2 Other run adaptions

If you have other issues because of CPU size, run time, etc., you can also adapt this in the *run\_folder\_decon.sh /* *run\_image\_decon.sh* scripts. It is very similar to the changes for the GPU issues: You have to again adapt the #SBATCH entries at the beginning of the script.  
For an example, see <https://scicomp.ethz.ch/wiki/Using_the_batch_system#Job_scripts>.  
On this site, you can find all the different commands available. Just put them at the top after a *#SBATCH*.