**Guide on deploying the LUTO model**

**on a High-performance computing (HPC) node**

Why deploy LUTO on an HPC?

Running LUTO (Land-Use Trade-Offs) requires 50G to 200G, depending on the spatial gratuity, of memory, and 10 to 30 CPU cores to parallelize the parameter searching and output writing. To answer the many “what-if” questions, we often want to test LUTO with different scenario settings such as climate change, demand trajectory, and cost and revenue. Intensive computation and multiple scenarios exploiting demand for an efficient way to run LUTO and deploy it on HPC is a proper choice.

Knowledge requirement.

Although we have tried our best to wrap the deploying process into simple scripts, it is still suggested that users have a basic level of understanding of Python and Linux commands.

* For the Python part, users are assumed to know what packages are, understand how to create a Python environment with [Conda](https://conda.io/projects/conda/en/latest/user-guide/install/index.html), and be able to type commands using IPython.
* For the Linux part, users are assumed to understand the concept of relative and absolute paths, know how to change directories, and use bash to execute a script.
* For the code management part, users are assumed to have the knowledge of cloning (i.e., downloading) the LUTO model and updating the code from [GitHub](https://github.com/land-use-trade-offs/luto-2.0).

The minimal deploying case.

|  |  |  |
| --- | --- | --- |
| No. | Operation | One-time operation? |
| 1 | Download the Gurobi TM optimizer | Yes |
| 2 | Login to the HPC and install Gurobi | No/Yes |
| 3 | Clone LUTO from GitHub | Yes |
| 4 | Download and convert input data | Yes |
| 5 | Install the conda management app | Yes |
| 6 | Create the Python environment | Yes |
| 7 | Generate setting file | No |
| 8 | Submit tasks to HPC | No |

1. Download the Gurobi TM optimizer.

The HPC is mostly a Linux platform so we chose Gurobi’s Linux version for demonstration. Gurobi is a commercial software but offers a free trial for academic users. Below are the steps to obtain an academic trial account, make sure you have an education email before applying.

* [Register for a free Gurobi account](https://portal.gurobi.com/iam/register/) as an academic.
* [Login](https://portal.gurobi.com/iam/login/) an click the A blue background with white text and a file

  Description automatically generated button, then A blue circle with a white cross in it

  Description automatically generated for WLS Academic license.
* Go back to A blue background with white text and a file

  Description automatically generated and observe the  row, click the right-most A black arrow in a square

  Description automatically generated
* In the newly opened page, find the license block and hit
* Hit the , fill in the required columns, and you will see a line of code.
* Hit the , you will see a file being downloaded.
* Go the [the Gurobi download](https://www.gurobi.com/downloads/gurobi-software/) page and download its Linux version.

1. Login to the HPC and install Gurobi.

Now you can use Gurobi optimizer, but the software is not yet installed. Unlike simple double-click to install apps in a Windows system, installing apps in a Linux platform can be challenging because it requires extra steps to config the paths. Below is a walk-through to install Gurobi on a remote Linux server.

* Install an SSH tool. Here use [MobaXterm](https://mobaxterm.mobatek.net/download.html) as an example.
* After installing the MobaXterm, open it and hit the  to create a new connection.
* Click the A yellow key in a square with black text

  Description automatically generated to specify the connection type as SSH.
* Fill in the ， , and . These are the settings you should obtain from the HPC manager.
* Hit OK, and now you are connected to the HPC.
* By default, you should see a white panel on the left and a black screen on the right.
* The left white panel is your “home” directory, just think of it as a regular file folder is fine.
* The right black screen is the “terminal”, which is the place we give instructions to HPC.
* Drag the downloaded Gurobi file to the left white panel.
* After a while, you should see the file appear in the left white panel.
* Now type in `ls` in the right black screen and hit enter. You should see a line showing the file like .
* Now type in `tar -xzf gurobi11.0.1\_linux64.tar.gz` in the terminal. After a few seconds, there should be nothing happened in the terminal.
* Now go to the left pannel, hit the “refresh folder” button , you will see a new folder appears .

Now Gurobi is technically functional, but the problem is how to use it. Linux needs us to tell it the existence of Gurobi, which, in formal terms, is to configure the system path to Gurobi. Below are the steps to do that.

* Let's try to enter a Gurobi command `gurobi.sh` in the terminal, and you will get an error message as follows . This is because Linux does not know the location to execute this command.
* Hit the ‘Show hidden files’ button 
* Now we can see many transparent icons and the  is the file that Linux hears from us to answer “which app has been installed on where”.
* Double click the , a text editor will pop out and possibly show many alien-like text.
* Scroll down to the last line and we need to type in something to inform Linux where Gurobi has been installed.
* Go back to the file explorer panel (left white panel), double click the , and make sure you can see some directories like , , and .
* If there is only a directory, then you need to double-click it to make sure you can see many directories like in the previous step.
* Now we can see the many directories, we are confident that all Gurobi files are stored here, and we need to send this message to Linux.
* Copy the path in the navigation column , then go to the opened  text editor, and type in the following information. Your path should be different from the example, make sure you copied the path from the navigation line.
  + export GUROBI\_HOME=/home/jinzhu/gurobi1101/linux64/
  + PATH=$PATH:/home/jinzhu/gurobi1101/linux64/bin
  + LD\_LIBRARY\_PATH=/home/jinzhu/gurobi1101/linux64/lib
* Now let’s run the command `source .bashrc`, which can be conceptually understood as “refresh” in a Windows system.
* If we run the `grbgetkey` command in the terminal, we can observe some output. We can confirm that Linux now understands the Gurobi command.
* At last, let’s move the to our home directory. To get the path of “home directory”, we can first enter `cd` in the terminal and then enter ‘pwd’. The returning path is the “home directory”.
* Change the file explorer panel to the “home directory”, and then drag the to it.
* Now if we run `gurobi.sh`, we will get some information saying Gurobi is setting some arguments and so on. Enter `exit()` to exit from this secession. Congratulations, you are now able to use Grurobi optimizer, and we probably will not do anything with Grurobi in the future.