TCLB* installation instructions on Bunya

Lisa Taldir

May 2025

*https://github.com/lisataldir/TCLB_arbitrary

Connecting on Bunya[1]

Connection to the login node

```
$ ssh -X username@bunya.rcc.uq.edu.au  # where username is the UQ username
```

→ Users will first be asked for their password (or key) and then be asked for their DUO passcode (6 numbers with no spaces).
→ The `-X` option enables the user to use a graphical interface.
→ Do not start your calculation, compile or environment install on a login node. Make sure you are on a compute node. You can use the command `hostname` to see if you are on a compute node or not. If this shows bunya1, bunya2, or bunya3 you are still on a login node.

Connection to a node

```
$ salloc --nodes=1 --ntasks-per-node=1 --cpus-per-task=1 --mem=5G --job-name=TCLB
--time=01:00:00 --partition=general --qos=debug --account=a_mcf --x11 srun
--export=PATH,TERM,HOME,LANG --pty /bin/bash -l
```

Prerequisites libraries

Modules load

To install GKlib, METIS and ParMETIS, load the following modules.

```
$ module load cmake/3.12.1
$ module load openmpi/5.0.3-gcc-13.3.0
```

Installation of GKlib[2]

```
$ git clone https://github.com/KarypisLab/GKlib.git
$ cd GKlib
$ make config
$ make
$ make install
```

Installation of METIS[3]

```
$ git clone https://github.com/KarypisLab/METIS.git
$ cd METIS
$ make config
$ make install
```

Installation of ParMETIS[4]

```
$ git clone https://github.com/KarypisLab/ParMETIS.git
$ cd ParMETIS
$ make config
$ make install
```

All the created files will be located in the ∼`/local` directory.

[1] https://github.com/UQ-RCC/hpc-docs/blob/main/guides/Bunya-User-Guide.md
[2] https://github.com/KarypisLab/GKlib
[3] https://github.com/KarypisLab/METIS
[4] https://github.com/KarypisLab/ParMETIS

Installation of TCLB

R packages installation

```
$ module load rstudio/2024.04.2-r4.2.1
```

→ *Note.* The two latest versions `rstudio/2024.04.2-r4.4.1` and `rstudio/2024.12.1-r4.4.2` are not "built" on apptainer at this time, that is why `rstudio/2024.04.2-r4.2.1` is the one loaded here.

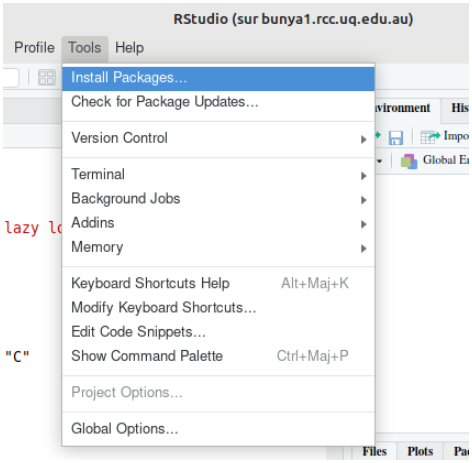
In order to build TCLB, the R packages optparse, rtemplate[5], gvector[6], polyAlgebra[7], and RInside have to be installed. Since rtemplate, gvector, and polyAlgebra are not available on CRAN, we will need to install them via a tar.gz file.

```
$ git clone https://github.com/llaniewski/rtemplate.git
$ tar -cf rtemplate.tar.gz rtemplate
```

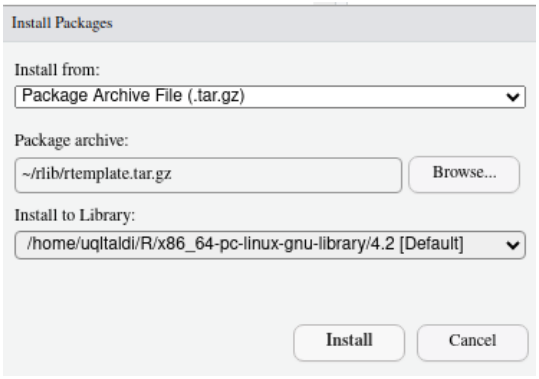
Now, let's launch the graphic interface.

```
$ rstudio  # launch the graphic interface
```

Then, follow these steps.



(a) Go to install packages



(b) Change Repository (CRAN) to Package Archive File and browse to the location of the tar.gz file

Figure 1: Installation example of rtemplate.

For RInside and optparse, just search for them in the CRAN packages.

[5] https://github.com/llaniewski/rtemplate
[6] https://github.com/llaniewski/gvector
[7] https://github.com/llaniewski/polyAlgebra

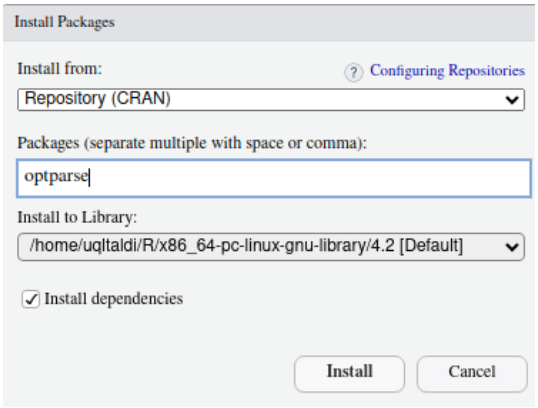


Figure 2: Installation of optparse

CUDA activation

```
$ exit
$ salloc --nodes=1 --ntasks-per-node=1 --cpus-per-task=1 --mem=5G --job-name=TCLB
--time=01:00:00 --partition=gpu_cuda --qos=gpu --gres=gpu:nvidia_a100_80gb_pcie_1g.10gb:1
--account=a_mcf --x11 srun --export=PATH,TERM,HOME,LANG --pty /bin/bash -l
$ module load cuda/11.7.0   # need to be on a gpu node to load it
$ module load rstudio/2024.04.2-r4.2.1 # need to re-load rstudio since we are on a new node
```

TCLB

In the same directory as METIS and ParMETIS do:

```
$ git clone https://github.com/lisataldir/TCLB_arbitrary
$ cd TCLB_arbitrary
```

And then:

```
$ make configure
$ ./configure --enable-cpp17 --with-metis=/home/[USERNAME]/local
```

Utilisation of TCLB

In this section, I will explain how to use the average and the sample handlers on the d3q27_cumulant_AVG model. *Note.* Job should be run from /scratch. It is not advisable to use /home as the space to run (submit) jobs.

Compilation

To compile the d3q27_cumulant_AVG model, just do:

```
$ make d3q27_cumulant_AVG
```

Example

Now, let's use this model on a simple example, to assert that the setup went well.

→ With a Cartesian lattice:

```
$ CLB/d3q27_cumulant_AVG/main example/flow/3d/cummulant_test.xml
```

→ With an arbitrary lattice:

```
$ CLB/d3q27_cumulant_AVG/main example/flow/3d/cummulant_test.xml . @toArb = true
$ CLB/d3q27_cumulant_AVG/main cummulant_test_ARB_P00.xml
```

Getting to know the handlers

In the xml file, make the changes shown below.

Average handler

The <Average/> handler is used to reset the average value. Use it as follows:

```
<Solve Iterations="1" />
<Average />
<Solve Iterations="1" />
```

Sample handler

The <Sample/> handler allows you to get the value of one or several quantities for a set of nodes in a csv file. Use it as follows (put it right after calling the <Model/> handler) :

```
<Sample Iterations="1" what="P, averageP">
  <Point dx="1m" dy="2m" dz="1m" />
  <Point dx="0m" dy="0m" dz="1m"/>
</Sample>
```

Annex: File transfer

From the cluster

```
$ scp username@bunya.rcc.uq.edu.au:/scratch/user/username/test.dat .
```

will copy the file test.dat to the user's current directory.

To the cluster

```
$ scp test.dat username@bunya.rcc.uq.edu.au:/scratch/user/username/
```

will copy the file test.dat to the user's scratch directory.