Modelling trees from terrestrial laser scanning data with Matlab and R

Timo P. Pitkänen / Luke
timo.p.pitkanen@luke.fi

Short background for the study

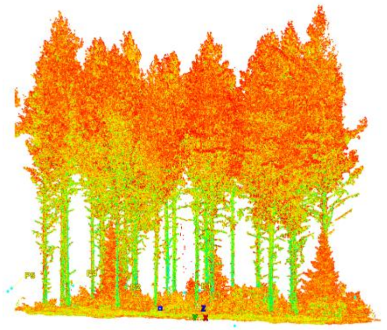
- Forest research is one of the key sectors at LUKE
- Most important product is national forest inventory (NFI)
  - Provides estimates on national and regional forest resources
  - Based on fieldwork, various measured variables
  - Generalized into wall-to-wall map by satellite images → MS-NFI
- Tree measurements are slow and tedious
  - Only few variables can be measured easily from the ground (stump height, DBH, h, species, lowest braches…)
  - For more detailed measurements, tree should be felled first
- Terrestrial laser scanning (TLS) can provide more detailed data
  - Accurate taper curve (diameter/height) measurements
  - Volume (and updated volumetric models) → biomass
  - Branching structure, canopy width, leaves/needles
  - Problem: huge amount of data; answer: CSC

Luke
NATURAL RESOURCES
INSTITUTE FINLAND

Data collection and analysis

Fieldwork at 9.0 m circular plots

- TLS scanning (3-5 stations)
- Field measurements (DBH, h, species)
- Hundreds of plots around Finland



Processing at own workstation (Leica Cyclone)

- Visual data evaluation
- Co-registration of single scans
- Exporting data in binary format

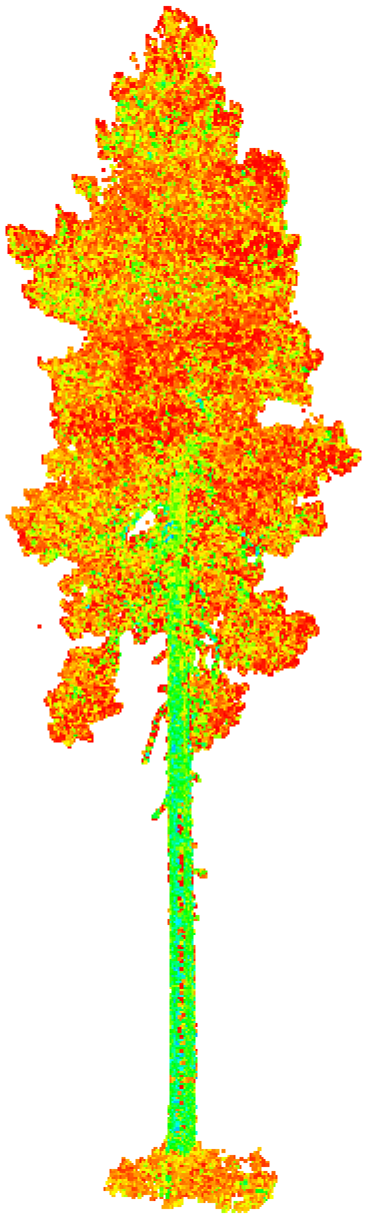
Processing at server (R, Matlab)

- Converting to .txt format, decimating (R)
- Points to cylinder structures (Matlab) → modelled trunks and branches
- Adjusting diameter measurements (R)

Results

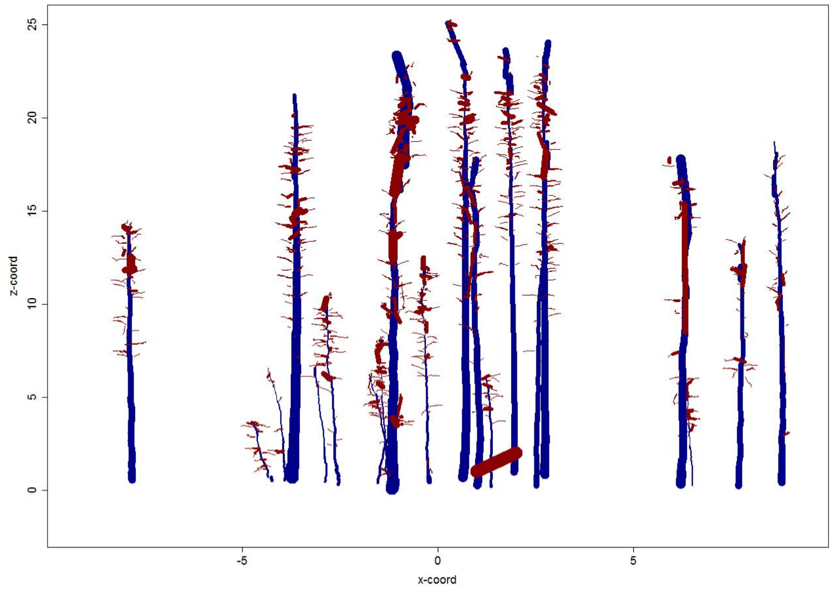
- Completeness and correctness of the extracted trees
- Accuracy of the models (DBH, h; whole stem in some cases)
- Species modelling based on tree structure (to be tested…)

Luke
NATURAL RESOURCES INSTITUTE FINLAND



Matlab modelling of TLS data

- Data can contain ~100M TLS points (x,y,z) per plot
- Step 1: filtering → deleting areas of low point density
- Step 2: extraction of the ground level (DEM)
- Step 3: finding lower parts of the stems
- Step 4: extending cylinder structure to the whole tree
- Step 5: writing the results out



Preconditions and facts to use Matlab at Taito

- You must have a licensed Matlab in your own workstation
- You must have a licensed Parallel Computing Toolbox
- You need to download connecting scripts from CSC
  - Check the correct Matlab version
- Only batch jobs can be submitted
  - Interactive step-by-step processing at Taito is disabled
  - Makes sometimes debugging complicated…
- Batch can called with a function, or a script
  - Input data can be submitted at a call, or can be at Taito
  - Output data can be retrieved to Matlab, or stored at Taito
- While processing, own Matlab / workstation can be off
- Parallel processing (`Parfor` loop) gives power!
  - 500 workers (academic) / 32 (commercial)

Luke
NATURAL RESOURCES
INSTITUTE FINLAND

Submitting a batch job to Taito

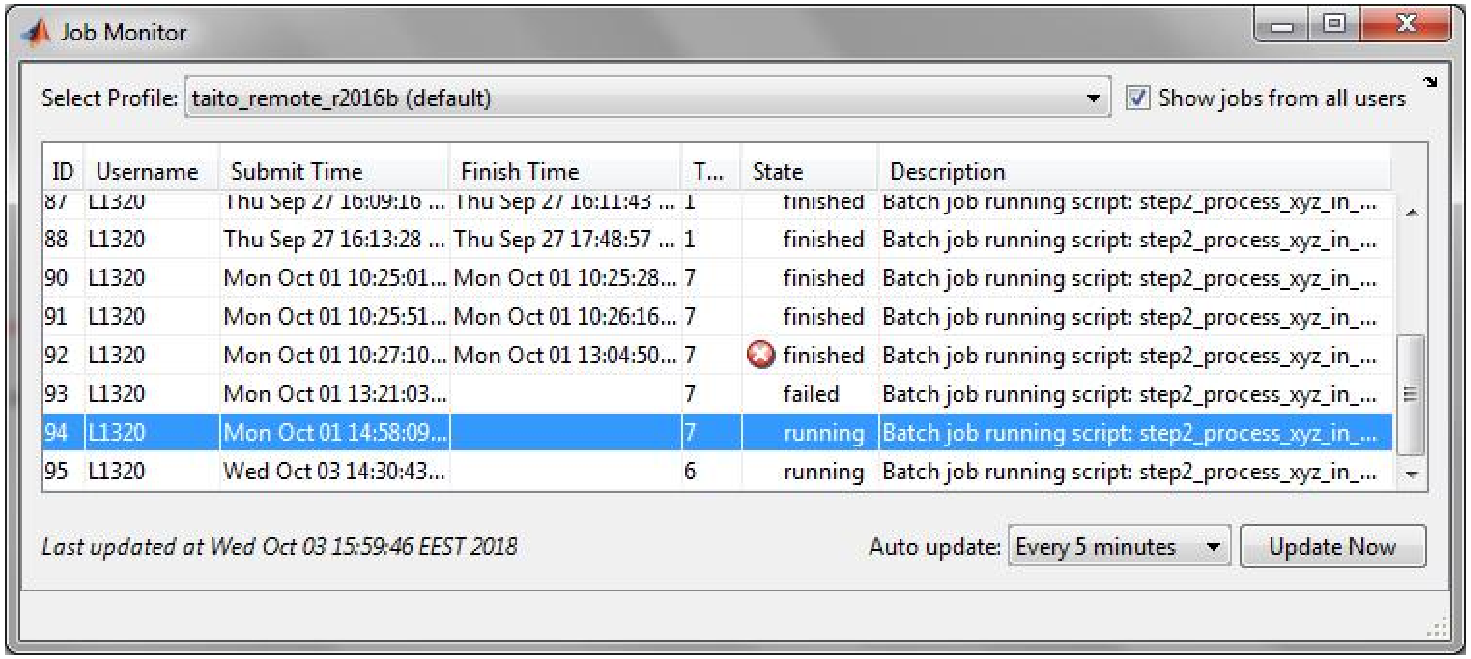
- Parallel cluster (Taito) is called from own Matlab
  - Cluster is configured on the first time of use
  - Following times just calling the configured cluster
- Definitions for e.g. running time and mem usage are needed
  - In later versions (after 2016b) stored in preferences
- `batch` command defines the job to be submitted

```
c = parcluster;
ClusterInfo.setWallTime('24:00:0')
ClusterInfo.setMemUsage('8g')
ClusterInfo.setQueueName('parallel')
j = c.batch(’theScriptName','CurrentFolder',
   '/wrk/tpitkane/theScriptFolder/','CaptureDiary',true,
   'pool',5);
```

Luke
NATURAL RESOURCES
INSTITUTE FINLAND

Monitoring the submitted job (1)

- From own Matlab: *Job Monitor*
  - Shows if the job is running, finished or failed
  - In case of failure, error messages can be retrieved
  - Can be used to cancel the job (but may still run in Taito!)



Select Profile: taito_remote_r2016b (default)

☑ Show jobs from all users

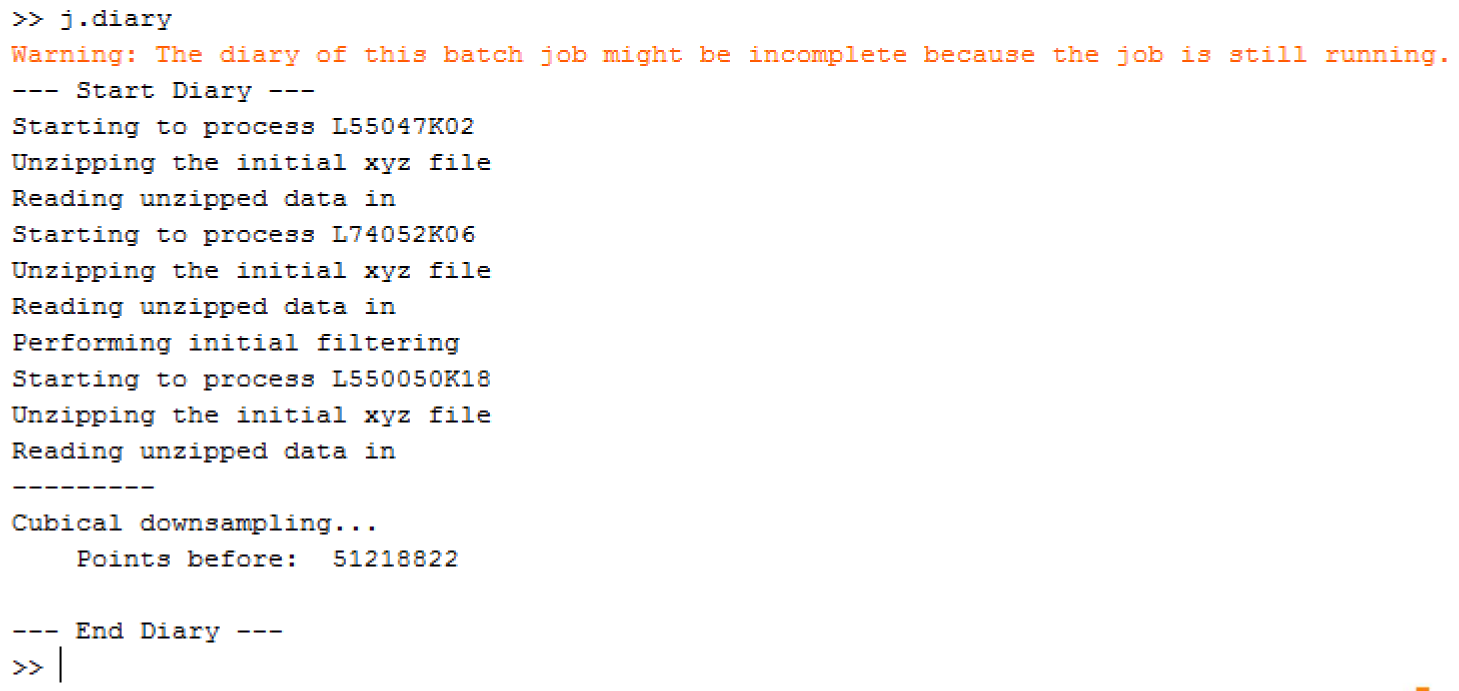
ID	Username	Submit Time	Finish Time	T...	State	Description
87	L1320	Thu Sep 27 16:09:16 ...	Thu Sep 27 16:11:43 ...	1	finished	Batch job running script: step2_process_xyz_in_...
88	L1320	Thu Sep 27 16:13:28 ...	Thu Sep 27 17:48:57 ...	1	finished	Batch job running script: step2_process_xyz_in_...
90	L1320	Mon Oct 01 10:25:01...	Mon Oct 01 10:25:28...	7	finished	Batch job running script: step2_process_xyz_in_...
91	L1320	Mon Oct 01 10:25:51...	Mon Oct 01 10:26:16...	7	finished	Batch job running script: step2_process_xyz_in_...
92	L1320	Mon Oct 01 10:27:10...	Mon Oct 01 13:04:50...	7	✖ finished	Batch job running script: step2_process_xyz_in_...
93	L1320	Mon Oct 01 13:21:03...		7	failed	Batch job running script: step2_process_xyz_in_...
94	L1320	Mon Oct 01 14:58:09...		7	running	Batch job running script: step2_process_xyz_in_...
95	L1320	Wed Oct 03 14:30:43...		6	running	Batch job running script: step2_process_xyz_in_...

Last updated at Wed Oct 03 15:59:46 EEST 2018

Auto update: Every 5 minutes

Monitoring the submitted job (2)

- From own Matlab: *diary* (if defined when calling the job)
  - Shows the “screen input” from the batch run
  - Can look a bit confusing in parallel processing

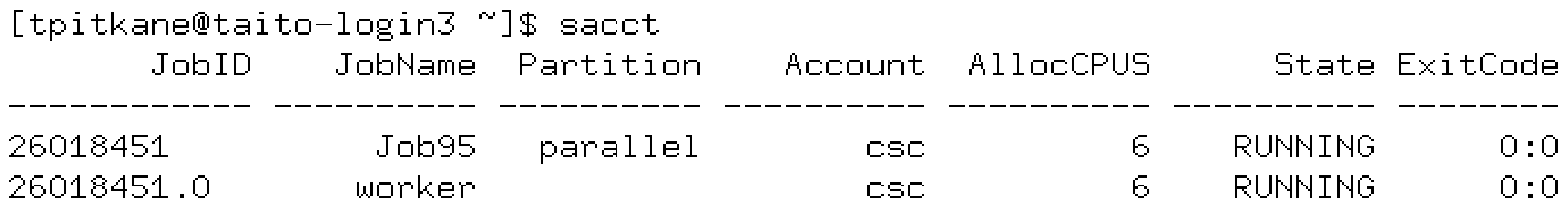


```
>> j.diary
Warning: The diary of this batch job might be incomplete because the job is still running.
--- Start Diary ---
Starting to process L55047K02
Unzipping the initial xyz file
Reading unzipped data in
Starting to process L74052K06
Unzipping the initial xyz file
Reading unzipped data in
Performing initial filtering
Starting to process L550050K18
Unzipping the initial xyz file
Reading unzipped data in
---------
Cubical downsampling...
    Points before:   51218822

--- End Diary ---
>> |
```

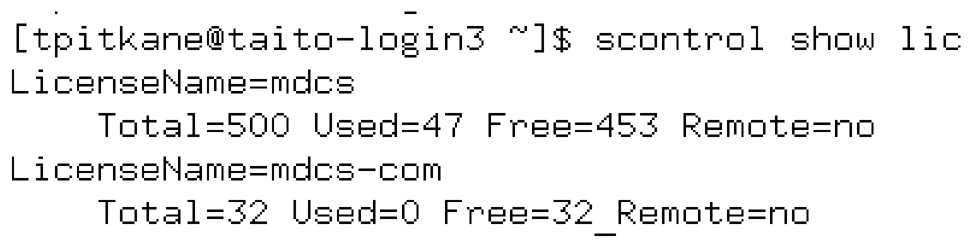
Monitoring the submitted job (3)

- From Taito (ssh): checking status using `sacct` command



```
[tpitkane@taito-login3 ~]$ sacct
       JobID    JobName  Partition    Account  AllocCPUS      State ExitCode
------------ ---------- ---------- ---------- ---------- ---------- --------
26018451          Job95   parallel        csc          6    RUNNING      0:0
26018451.0       worker                   csc          6    RUNNING      0:0
```

- License use can be checked by `scontrol show lic`



```
[tpitkane@taito-login3 ~]$ scontrol show lic
LicenseName=mdcs
    Total=500 Used=47 Free=453 Remote=no
LicenseName=mdcs-com
    Total=32 Used=0 Free=32 Remote=no
```

- Other important commands:
  - Checking the time / queue status: `squeue -u <your_username>`
  - Canceling the job (for good): `scancel <JobID>`

Some final notes

- Matlab `batch` processing enables quite complicated procedures
  - But keeping it simple makes things often working better!
- My choice: all the scripts and input / output data are in Taito
  - Main script, located in Taito, is called from the submitted job
  - Additional scripts (in Taito) are given as path definitions
  - Input data must have been transferred before to Taito, and output data must be fetched from Taito afterwards
  - Submitting the batch job is extremely fast
- Using Matlab instead of R?
  - Matlab is faster, particularly in modelling / optimization problems
  - Matlab has extensive set of tools, made with professional quality
  - R, however, may have more implementations for new / rare cases
  - Programming language is different (mostly syntax, not that much logic)
  - Matlab is not free (especially for non-academic users)

Thank you!

Luke
NATURAL RESOURCES
INSTITUTE FINLAND