Table of contents

Contents

1 Introduction

STEPS

The Short Term Ensemble Prediction System (STEPS) has several modules to

- disaggregate rainfall [2]
- simulate time-series of mean area rainfall [4]
- make design-storm simulations [3]
- and to produce seamless forecast when cascaded with NWP models [1]

STEPS is mainly used for nowcasting and multiple simulations of the past extreme events.

HiDRUS

- HiDRUS[1] is an implementation of the STEPS library to downscale GCM/RCM rainfall to a very high space-time resolution (1 km, 6 minutes).
- HiDRUS-1 is a fully stochastic downscaling scheme which uses a broken-line model to generate the mean area rainfall time-series (MARTS). Linear regression models are use to generate cascade parameters for each time-step.
- HiDRUS-2 uses sampling of the past MARTS to generate future rainfall projections.

This document describes the steps in running HiDRUS-2 model to downscale rainfall.

[1] Pronounced "haɪdrəs"

2 Running HiDRUS-2

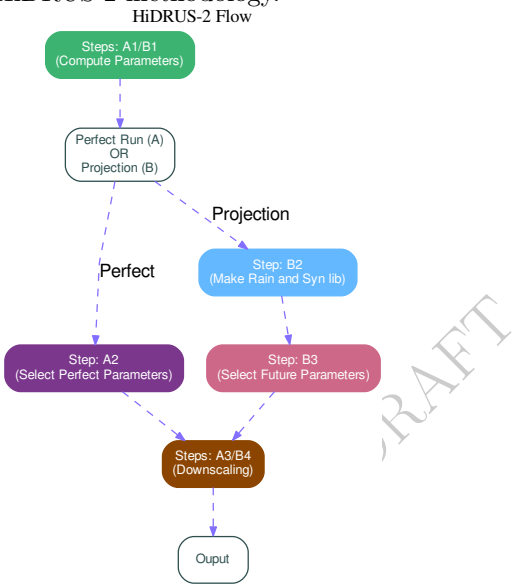
Running HiDRUS-2

HiDRUS-2 methodology has two configurations.

1. **perfect mode (A)** to downscale radar data with computed parameters. This is used for model validation, model bias & uncertainty estimation.

2. **projection mode (B)** to be run with GCM/RCM/reanalysis data.

The flow-digram shows important stages of the *perfect* and the *projection* modes of HiDRUS-2 methodology.



Compute Cascade Parameters

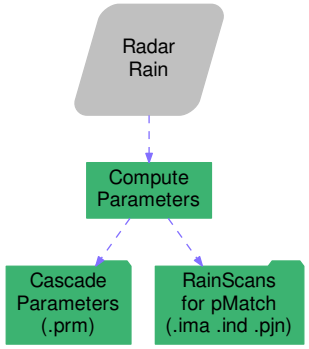
```
$mkcascparm <fPath> <rn>
```

fPath - path of the input directory, rn - Radar id. This will be suffixed to the output file name.

```
$./mkcascparm ~/data/ MLB
$cd ./c_parms_dbz
$ls
 20100108_MLB.prm
 ...
 ...
 20101224_MLB.prm
 ...
 RainPMatch_MLB.ima
 RainPMatch_MLB.ind
 RainPMatch_MLB.pjn
```

```
12 ...
```

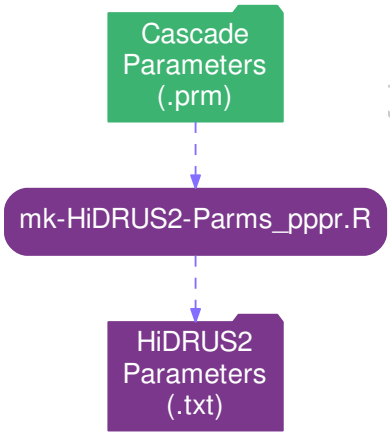
HiDRUS-2
Step A1 and B1: Compute Cascade Parameters



3 Perfect Run

Perfect Run - Make Parameters

HiDRUS-2
A2: Select Perfect Parameters

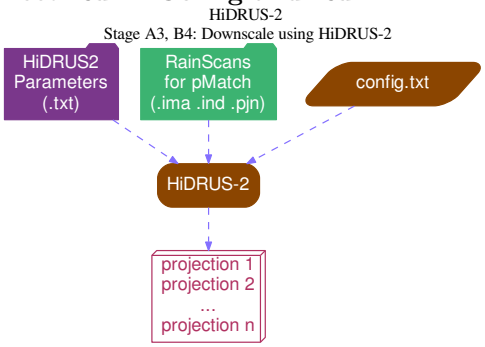



```
1 $vim mk-HiDRUS2-Parms_pppr.R
2
3 #set parameters directory
4 setwd("~/data/c_parms_dbz/")
5
6 #input file name pattern
7 fn_pat <- "_MLB.prm"
8
9 #desired output file name
10 ofName<-"simParm-2010.txt"
```

Run the script.

```
$Rscript mk-HiDRUS2-Parms_pppr.R
```

Perfect Run - Config and Run



Provide appropriate paths in config.txt file

```
$vim config.txt

outFilePathPrefix = /path/pSim2010 ;
pMatchPathPrefix = /path/RainPMatch_MLB ;
prmFileName = /path/simParm-2010.txt ;
```

```
$hidrus2 ./config.txt ens_id
```

If you give ens_id here then ens_id in config.txt will be overridden. This feature can be used to make several simulations with the same config file.

Perfect Run - Excerpt of a PBS Job Script

```
#!/bin/bash
# Example submission of PBS jobs in a loop
NUMBERS=$(seq 1 100)
for NUM in ${NUMBERS}
do
  NAME=h2pp${NUM}
  echo "Submitting: ${NAME}"
  PBS="#!/bin/bash\n\
##PBS -N ${NAME}\n\
##PBS -l walltime=10:00:00\n\
##PBS -l mem=500MB\n\
##PBS -l wd\n\
../hidrus2 ./config.txt ${NUM}"
        echo -e ${PBS} | qsub
        echo "done."
done
```

Perfect Run - Output

In output directory

```
$ls
pSim2010_001_tStat.asc pSim2010_055_tStat.asc
pSim2010_002_tStat.asc pSim2010_056_tStat.asc
...
pSim2010_001.nc pSim2010_055.nc
pSim2010_002.nc pSim2010_056.nc
...
```

The files "$*\_tStat.asc$" contains domain average properties for all rainy time-steps.

```
$head pSim2010_055_tStat.asc
time pm in_dbz_fmean in_dbz_fstd in_rain_fmean in_rain_fstd in_rain_cmean
in_rFrac in_east in_south out_dbz_fmean out_dbz_fstd out_rain_fmean
out_rain_fstd out_rain_cmean out_rFrac
1200018240 1200018273 1.83 7.65 0.52 3.93 8.6 3 0.06 2.0 0.2
1.83 7.64 0.52  3.92 8.62  0.06
```

Perfect Run - Output

```
$ncdump -h pSim2010_055.nc
netcdf pSim2010_055 {
dimensions:
        lat = 230 ;
        lon = 256 ;
        time = UNLIMITED ; // (23000 currently)
variables:
        float lat(lat) ;
    lat:units = "degrees_north" ;
        float lon(lon) ;
    lon:units = "degrees_east" ;
        int64 time(time) ;
    time:units = "seconds since 1970-01-01 00:00:00 UTC" ;
        float rain(time, lat, lon) ;
    rain:units = "mm/hr" ;
    rain:_FillValue = -999.f ;
    rain:long_name = "instantaneous rainrate" ;
    rain:_description = "perfect_simulations" ;
// global attributes:
                :HiDRUS = "developed at Monash University and Bureau of Meteorology." ;
                :creator_name = "Bhupendra A. Raut and Alan W. Seed" ;
                :creator_webpage = "www.baraut.info" ;
                :_institution = "School of Earth Atmosphere and Environment, Monash University" ;
                :_acknowledgment = "Bureau of Meteorology and CRC for Water Sensitive Cities" ;
                :_project = "B1.1: Cities as Water Supply Catchments  Urban Rainfall in a changing c
                :_project_leaders = "Christian Jakob, Michael J. Reeder" ;
                :date_created = "Tue Sep 29 11:53:59 2015" ;
```

```
}
```

4 Projection Run

Projection Run - Make Rain Events Library

```
$vim mk_rainEvents_lib.R
```

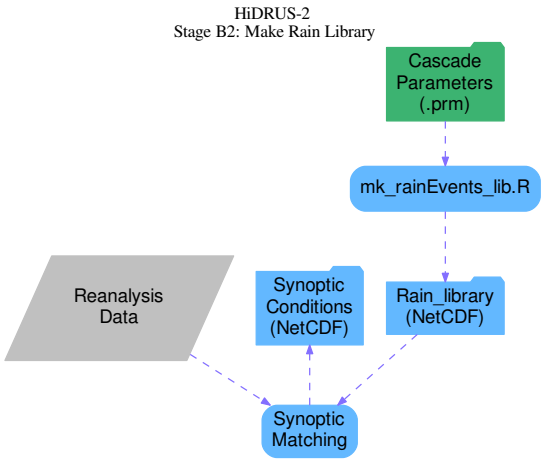
```
#set working directory to parameters directory
setwd("/home/bhupendra/data/c_parms_dbz/")
ofileName<-"RainLib_2010_MLB"  #no extension

#--------- Some initialisation settings----------------#
window <- 240    #number of point in sampled ts
cut_off=0.1      #mean area rain mm/hr
duration=5       #min duration of rain in tSteps
nprms=65         # number of parameters
fn_pat <- "_MLB.prm"    #file name pattern
```


Projection Run - Make Rain Events Library

```
$Rscript mk_rainEvents_lib.R
$ls
RainLib_2010_MLB.nc RainLib_2010_MLB.pdf
```

The cascade parameters for each event will be stored in the NetCDF file.

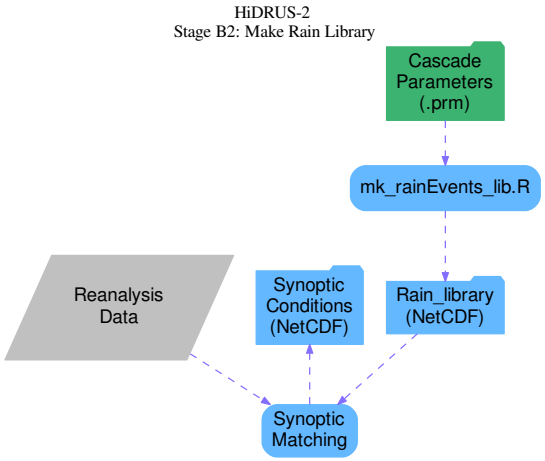
```
$ncdump -h RainLib_2010_MLB.nc
dimensions:
    id = UNLIMITED ; // (1546 currently)
    prms = 65 ;
    tstep = 240 ;
variables:
    int id(id) ;
    id:long_name = "identity number
    for the event" ;
        double ts_prms(id, tstep, prms) ;
    ts_prms:long_name = "cascade parameters" ;
```



Projection Run - Make Synoptic Conditions Library

At this stage auxiliary data can be stored for the back-identification of events in future projections. This may include

- Weather Regimes
- vertical profiles of U, V, humidity etc.
- MSLP patterns,
- accumulation of rain from Rain-gauge, TRMM

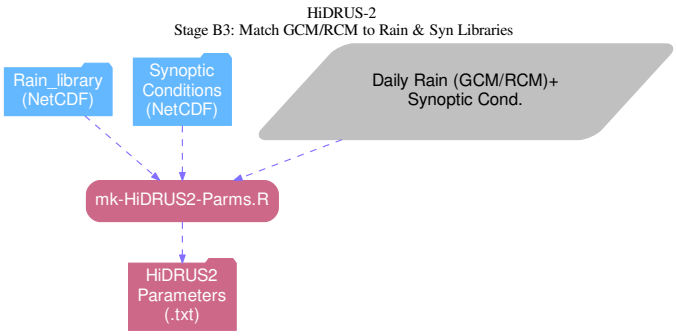


Projection Run - Select Future Parameters

- An example script mk-HiDRUS2-Parms.R is provided for selecting future days.

- Users will have to use their matching criterion for the past and future days.

```
$cdo fldmean -sellonlatbox,143.5,146,-36.5,-39 ifile.nc ofile.nc
```

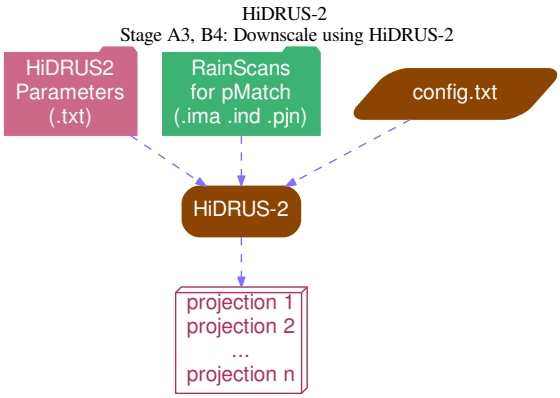


Projection Run - Config and Run

- Running HiDRUS-2 for projection is same as for the perfect run.
- An optional argument "parmFilePath" can be used to provide parameters file path. This will over-write the path of the parameter file in config.txt
- This allows us to run HiDRUS-2 for several input parameter files without modifying the config file.

possible run commands

```
$hidrus2 ./config.txt
$hidrus2 ./config.txt ens_id
$hidrus2 ./config.txt ens_id parmFilePath
```



Projection Run - PBS Job Script

```
from popen2 import popen2
import time

yesno='no'
yesno = raw_input("Type␣'YeSs'␣to␣submit␣the␣HiDRUS␣jobs.\t")

if(yesno!='YeSs'):
    quit()

#input for the command
cfile="./config.txt"
ens=range(1, 101)
pfile_prefix="./h2parms_erai_2008-14_"
```

Projection Run - PBS Job Script (Contd.)

```
# Loop over your jobs
for ens_id in ens:
    #make input parameters file name
    pfile=pfile_prefix+ str(ens_id)+ ".txt"

    # Customize your options here
    job_name = "h2era_"+str(ens_id)
    walltime = "10:00:00"
    mem="500MB"
    qtype = "normal"

    command = "./hidrus2␣%s␣%d␣%s" %(cfile, ens_id, pfile)
```

Projection Run - PBS Job Script (Contd.)

```
 job_string = """#!/bin/bash
␣␣␣␣#PBS␣-N␣%s
␣␣␣␣#PBS␣-q␣%s
␣␣␣␣#PBS␣-l␣walltime=%s
␣␣␣␣#PBS␣-l␣mem=%s
␣␣␣␣#PBS␣-P␣k10
␣␣␣␣#PBS␣-l␣wd
␣␣␣␣#PBS␣-m␣a
␣␣␣␣#PBS␣-M␣bhupendra.raut@monash.edu
%s""" % (job_name, qtype, walltime, mem, command)

    # Open a pipe to the qsub command.
    output, input = popen2('qsub')
```

```
# Send job_string to qsub
input.write(job_string)
input.close()

#Print job and the system response as it's submitted
print "\tsubmiting... "+job_name
print output.read()
time.sleep(0.5)
```

Projection Run - PBS Job Script (Contd.)

```
#Print job and the system response as it's submitted
print "\tsubmiting... "+job_name
print output.read()
time.sleep(0.5)
```

5 List of Programs and Scripts

List of Programs and Scripts

1. **mkCascParm_dbz** : This C++ program computes STEPS parameters required to produce HiDRUS simulations.
2. **mk_rainEvents_lib.R** : This script creates library of rain-events and their parameters.
3. **add_mslp_uv_rainlib.ncl** : This script is an example of adding relevant synoptic data to the rainlib for the better identification of the future days.
4. **mk-HiDRUS2-Parms.R** : This script taken in rain library files produced by above scripts and GCM/RCM data and assigns rainfall events to future days. Prints all the required parameters in an ascii file.
5. **hidrus2_dbz** : This is the main simulation program which takes in the file generated by the *mk-HiDRUS2-Parms.R* and produces downscaled rainfall realisation. You can run this program several time with the same input to get ensemble of realizations.
6. **nc2nc_ents** : This program extracts the rainfall time-series at a given location from 100s of HiDRUS simulations and writes it in a netcdf file.

References

References

[1] Neill E Bowler, Clive E Pierce, and Alan W Seed. Steps: A probabilistic precipitation forecasting scheme which merges an extrapolation nowcast with downscaled nwp. *Quarterly Journal of the Royal Meteorological Society*, 132(620):2127–2156, 2006.

[2] Alan W Seed, Clive E Pierce, and Katie Norman. Formulation and evaluation of a scale decomposition-based stochastic precipitation nowcast scheme. *Water Resources Research*, 49(10):6624–6641, 2013.

[3] Alan W Seed, R Srikanthan, and Merab Menabde. A space and time model for design storm rainfall. *Journal of Geophysical Research: Atmospheres (1984–2012)*, 104(D24):31623–31630, 1999.

[4] AW Seed, C Draper, R Srikanthan, and M Menabde. A multiplicative broken-line model for time series of mean areal rainfall. *Water Resources Research*, 36(8):2395–2399, 2000.