Visualization and Histograming with COAST

Ralf Ulrich

Karlsruhe Institute of Technology

CORSIKA School 2010, Ooty, India

Please get: `coast-exercises-ooty.tar.gz`
from: `www-ik.fzk.de/~rulrich/coast/releases`

Outline

Requirements

Dependencies

For histograming	ROOT	root.cern.ch
Animations (optional)	gifsicle	packet: gifsicle/ungifsicle
	pstopnm/ppmtogif	packet: netpbm

CORSIKA

At least version v6.970 (On the USB-stick)

COAST + COAST-Interface

Version: v4r1

Get coast-v4r1.tar.gz and coast-interfaces-v4r1.tar.gz from

- http://www-ik.fzk.de/$\sim$rulrich/coast
- USB-sticks

Exercises

Get coast-exercises-ooty.tar.gz

Using COAST Together with CORSIKA

- Get COAST (e.g. from `http://www-ik.fzk.de/$\sim$rulrich/coast`)
- Define environment variable `COAST_DIR`
  (e.g. `export COAST_DIR=<path>/COASTinstall`)
- Add `${COAST_DIR}/lib` to `LD_LIBRARY_PATH`
- Define environment variable `COAST_USER_LIB` to
  select the user interface you want to use in CORSIKA.
  e.g. `export COAST_USER_LIB=<path>/coast-interfaces-v4r1/plot2D`
- Install COAST with `./configure; make install`
- Install CORSIKA with `./coconut` and option `ROOTRACK`
- Run CORSIKA

- `ROOTSYS` environment variable must be defined
  verify with: `echo $ROOTSYS`
- `${ROOTSYS}/lib` must be part of `LD_LIBRARY_PATH`
  verify with: `echo $LD_LIBRARY_PATH`

Preparation / Environment

Unpack source code

- Change to your favorite directory (e.g. ${HOME}/coast-exercise)
- Unpack tar.gz in this directory
  - `tar xzvf coast-v4r1.tar.gz`
  - `tar xzvf coast-interfaces-v4r1.tar.gz`
  - `tar xzvf corsika-6970.tar.gz`
  - `tar xzvf coast-exercises-ooty.tar.gz`

<1 min

Define environment (e.g. by: `source setEnvironment.[c]sh`)

Location of COAST installation

```
export COAST_DIR=<dir>/coast-v4r1-install
setenv COAST_DIR <dir>/coast-v4r1-install
```

Choice of COAST-Interface

```
export COAST_USER_LIB=<dir>/coast-interfaces-v4r1/plot2D
setenv COAST_USER_LIB <dir>/coast-interfaces-v4r1/Histogram
```

Add ${COAST_DIR}/lib to your LD_LIBRARY_PATH

```
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${COAST_DIR}/lib
setenv LD_LIBRARY_PATH ${LD_LIBRARY_PATH}:${COAST_DIR}/lib
```

Compilation / Installation

COAST + COAST-Interface

- `cd <dir>/coast-v4r1`
- `./configure`; `make install`

<3-4 min

CORSIKA

- `cd <dir>/corsika-6970`
- `./coconut`

<2-3 min

- Always accept the `[default]` offered by `coconut` (by pressing enter). Just change:
- If you work on a 64-bit system, select the 64-bit option of `coconut`

```
Compile in 32 or 64bit mode ?
1 - Force 32bit mode [DEFAULT]
2 - Use compiler default ('-m64' on a 64bit machine)
```

- Select: SIBYLL and GHEISHA ($\rightarrow$ fast)
- Options: Thinning (5), Slant (9) and ROOTRACK (q)
- Finish selection and start compilation (by pressing several `enter`s)

Visualization (2D)

set COAST_USER_LIB to <path>/coast-interfaces-v4r1/plot2D

(edit setEnvironment.[c]sh)

Customization of 2D Plots

Edit: `COAST2DConfig.config`

Option	Description	Default
output-extension	everything `TCanvas::SaveAs` understands (e.g. png, eps, root, jpg, ...)	png
bg-color	RGB code [0...1]	1.0 1.0 1.0
particle	name pid RGB	γ, e, μ, p, n, π
weight-boost-muons	visually increase weight of muons	1
weight-boost-hadrons	visually increase weight of hadrons	1
auto-range	automatically determine the horizontal range	1
xmin, xmax, ymin, ymax	range of coordinates to display in km	-0.05, 0.05, -0.1, 33
stretch_x	stretch shower in x-direction	50
pxl_per_km	output resolution	50

Try to e.g. increase the weight-boost-muons and weight-boost-hadrons to 50

Run CORSIKA

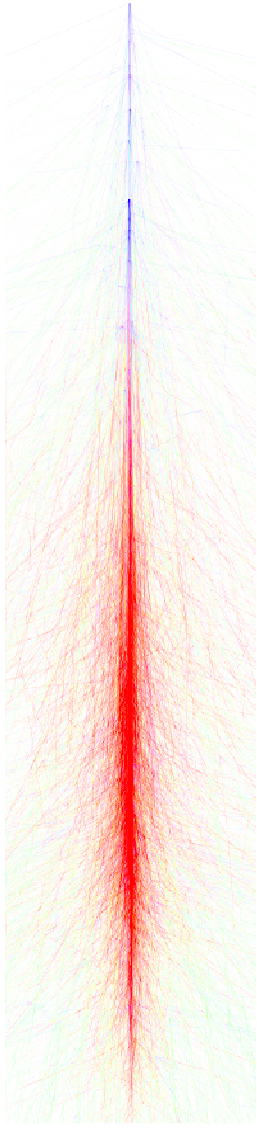
```
./corsika6970Linux_SIBYLL_gheisha < plot2D.inp
```

(The `plot2D.inp` CORSIKA steering card:
proton, 10^{14} eV, vertical, strong thinning)

<1 min

```
cd <dir>/corsika-6970/run
```

Output: `DAT000001_1.png`



Experiment with the `COAST2DConfig.config` and `plot2D.inp` files and `CORSIKA` options.

Histograming

set COAST_USER_LIB to <path>/coast-interfaces-v4r1/Histogram

(edit setEnvironment.[c]sh)

Customization of Histograming

Edit: `${COAST_USER_LIB}/TUser.cc`

Function: `void TPlotter::InitParticles()`

```
fParticles[3] = ParticleDef("electron", 4);
```

(e.g. γ:1, e^+:2, e^-:3, μ^+:5, μ^-:6, π^0:7, π^+:8, π^-:9, n:13, p:14, $\overline{p}$:15)

Function: `void TPlotter::InitHistograms(HistDef& hists)`

```
hists["2"] = new TProfile("hAngle", "angle",
                          6, -2.5, 2.5, "s"); //[lg(r/rm)]
hists["2"]->SetMarkerStyle(21);
hists["2"]->SetXTitle("log_{10}(r/r_{m})");
hists["2"]->SetYTitle("Theta   [deg]");
```

Function: `TPlotter::FillHistograms(...)`

```
((TProfile*)hists["2"])->Fill(log10(r/rm), theta/deg,
                              weight);
```

Don't forget: `cd ${COAST_USER_LIB}; make`

Run CORSIKA

```
cd <dir>/corsika-v6970/run
./corsika6970Linux_SIBYLL_gheisha < hist.inp
```

~2 min

(e.g. the `hist.inp` CORSIKA steering card: proton, 10^{17} eV, 20°)

Read output

```
root DAT000001_1.hist.root
TProfile* h = 0;
data_electron->SetBranchAddress("hAngle_electron", &h);
data_electron->GetEntry(10);
h->Draw();
```

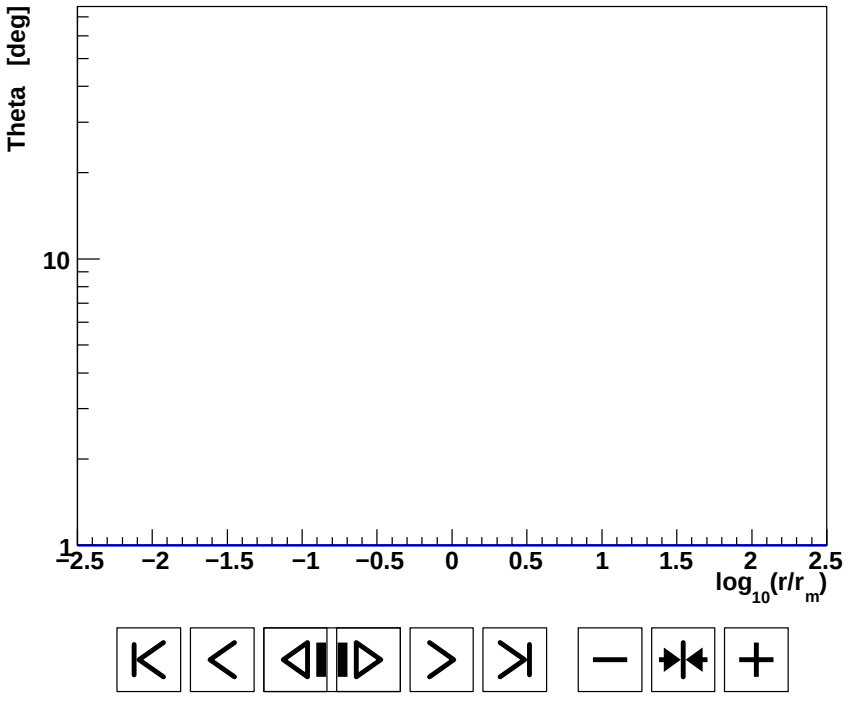
Generate animated histograms (OPTIONAL)

```
${COAST_USER_LIB}/MakeAnim DAT000001_1.hist.root
gifview DAT000001_1.hist_hAngle_electron.gif
```

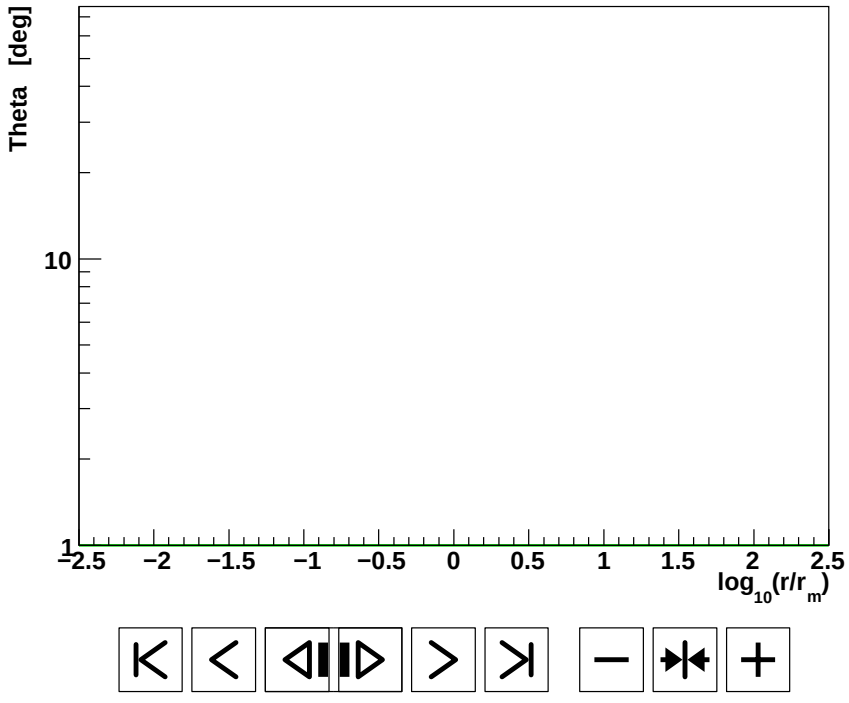
1 min

Result

electrons



muons



⇒ Powerful tool for studying the internal structure of air showers
⇒ Easy to use/customize
⇒ Very flexible

CORSIKA File Reading

CORSIKA Reader Skeleton

- See `${COAST_USER_LIB}/CorsikaRead` for an example.
- Auto-detect of thinning
- This example generates one `ROOT TTree` with the particles for each observation level found in the data file.

Try:

```
cd <dir>/coast-interfaces-v4r1/CorsikaRead; make
./CorsikaPlotter <dir>/coast-exercises-ooty/DAT000001
root DAT000001_1.root
data_1->Draw("y:x")
```

```
crsRead::MCorsikaReader cr(fname, 3);

crs::MRunHeader Run;
while (cr.GetRun(Run)) {

  crs::MEventHeader Shower;
  while (cr.GetShower(Shower)) {

    crs::TSubBlock Data;
    while (cr.GetData(Data)) {

      switch (Data.GetBlockType()) {
          case crs::TSubBlock::ePARTDATA:
          {
            const crs::MParticleBlock& ParticleData = Data;
            crs::MParticleBlock::ParticleListConstIterator iEntry;
            for (iEntry = ParticleData.FirstParticle();
                 iEntry != ParticleData.LastParticle();
                 ++iEntry) {

              if (iEntry->IsParticle()) {
                crs::MParticle iPart(*iEntry);
                const int id    = iPart.GetParticleID();
                const int level = iPart.GetObservationLevel();
                const double e  = iPart.GetKinEnergy();
                const double x  = iPart.GetX();
                const double y  = iPart.GetY();
                ...
```

The End