Ebm1D cheat sheet

André Paul (apaul@marum.de)

15 November 2018

1 Preparation

Copy all name list files and bash shell scripts to the `run` directory:

```
Ebm1D $ cd input
input $ cp *.nml ../run
input $ cd ../utils/bash
bash $ cp * ../../run
```

2 PD experiments

Change to the working directory, compile and run the forward model and save the results:

```
bash $ cd ../../run
run $ ./compile_forward.sh
run $ ./prgcost > printout.txt
run $ ./copy_results.sh PD0
```

Change to the `utils/python` directory and run the Python scripts `plot_surface_temperature.py` and `plot_surface_temperature_difference.py` to plot the simulated surface temperature in comparison to the observed surface temperature:

```
run $ cd ../utils/python
python $ python plot_surface_temperature.py &
python $ python plot_surface_temperature_difference.py &
```

Invoke TAF to generate the adjoint code for calculating the derivatives of the cost function with respect to the control variables.

```
python $ cd ../../run
run $ ./generate_adjoint.sh
```

Alternatively, change to the `src` *directory and copy the adjoint code provided for the PD1 experiment:*

```
python $ cd ../../src
src $ cp taf_ad_pd1.f90 taf_ad.f90
src % cp ebm1d_data_ad_pd1.f90 ebm1d_data_ad.f90
src % cp ebm1d_procedures_ad_pd1.f90 ebm1d_procedures_ad.f90
src $ cd ../../run
```

Compile and run a test program and open the printout using a text editor to compare the algorithmic derivatives to the finite-difference approximations:

```
run $ ./compile_test.sh
run $ ./prgtest > printout.txt
```

To optimize the values of the control variables, compile and run the adjoint model and save the results:

```
run $ ./compile_adjoint.sh
run $ ./prgopti > printout.txt
run $ ./copy_results.sh PD1
```

Change to the `utils/python` directory and plot and dicuss the time evolution of the cost function:

```
run $ cd ../utils/python
python $ python parse_cost.py
python $ python plot_cost.py &
```

Edit the Python scripts `plot_surface_temperature.py` and `plot_surface_temperature_difference.py` using a text editor (change the experiment name to `PD1`). Then run the scripts to plot the optimized surface temperature and its difference from the observation:

```
python $ python plot_surface_temperature.py &
python $ python plot_surface_temperature_difference.py &
```

3 LGM experiments

To use the optimal values of the control variables, open `Ebm1DParametersOut.nml` and `Ebm1DParametersIn.nml` in the `run` directory using a text editor and copy the following numbers from `Ebm1DParametersOut.nml` to `Ebm1DParametersIn.nml`:

```
hocn=60.8,
alw=209.2,
diff0=2.2E05,
diff2=−1.26,
diff4=0.32,
```

In addition, change the following parameters:

```
pyear=−21000.0,
co2ccn=200.0,
```

Furthermore, edit `numbmod.f90` in the `src` directory and set `n` (the number of control variables) to 1. Finally, make a copy of the LGM run parameters, regenerate the adjoint code *(alternatively, change to the* `src` *directory and copy the adjoint code provided for the LGM1 experiment, cf. the PD1 experiment)* and recompile the adjoint model, run it, save the results and inspect the simulated and reconstructed surface temperature anomalies using the Python script `plot_surface_temperature_anomaly.py`:

```
run $ cp LGM_RunParameters.nml RunParameters.nml
run $ ./generate_adjoint.sh
run $ ./compile_adjoint.sh
run $ ./prgopti > printout.txt
run $ ./copy_results.sh LGM1
run $ cd ../utils/python
python $ python plot_surface_temperature_anomaly.py &
```

Now edit `numbmod.f90` in the `src` directory to set the number of control variables `n` to 4. Regenerate the adjoint code *(alternatively, change to the* `src` *directory and copy the adjoint code provided for the LGM2 experiment, cf. the PD1 experiment)*, recompile and rerun the adjoint model as before and save its results:

```
python $ cd ../../run
run $ ./generate_adjoint.sh
run $ ./compile_adjoint.sh
run $ ./prgopti > printout.txt
run $ ./copy_results.sh LGM2
```

Edit the Python script `plot_surface_temperature_anomaly.py` using a text editor (change the experiment name to `LGM2`) and run it:

```
run $ cd ../utils/python
python $ python plot_surface_temperature_anomaly.py &
```

4 2xCO2 experiment

Edit `numbmod.f90` in the `src` directory to reset the number of control variables `n` to 5. Change the following parameters in `Ebm1DParametersIn.nml`:

```
pyear=0.0,
co2ccn=690.0,
```

Make a copy of the PD run parameters, compile the forward model, run it and save the results:

```
run $ cp PD_RunParameters.nml RunParameters.nml
run $ ./compile_forward.sh
run $ ./prgcost > printout.txt
run $ ./copy_results.sh 2xCO2
```