Contents of DGvdH_SoilMoistureColdPools_Data
Aryeh Jacob Drager, last updated 10 October 2019

- **MATLABAnalysisScripts**
  - There are lots of Matlab scripts here. Some may not be necessary. The description that follows provides a guide for running the analysis.
  - In general, the following naming convention is used:
    - luzon040 = 95% soil moisture (DRENCHED-SOIL)
    - luzon041 = 75% soil moisture (WET-SOIL)
    - luzon042 = 50% soil moisture (MID-SOIL)
    - luzon043 = 25% soil moisture (DRY-SOIL)
    - luzon071 = 45% soil moisture
  - The general workflow for data analysis is as follows:
    1. Save data from the RAMS header files about the model base state and grids, as well as the output variable list, into Matlab's .mat format using `save_basestate_and_grids.m`
    2. Update `simulation_paths_and_timing.m` with simulation information (some file path information goes here; other information goes in `ramsread_h5_ColdPools_luzon_XXX.m`).
    3. Save raw data from the RAMS HDF5 model outputs, as well as derived variables (such as equivalent potential temperature) into Matlab's .mat format using some flavor of `ramsread_h5_ColdPools_luzon_XXX.m`. It is necessary to do this a few times: run simulation paths and timing, save raw data and derived variables, convert microphysical budget variables into 5-minute increments, save the smoothed density potential temperature field, and finally coarsen to the 250-m grid. Before coarsening to the 250-m grid, it is necessary to run `save_basestate_and_grids_coarsen.m`. The reason there are so many scripts of the form `ramsread_h5_ColdPools_luzon_XXX.m` is that I usually opt to run several processes simultaneously in order to save (wall) time. For example, `ramsread_h5_ColdPools_luzon_XXX_a.m` might save data from output times 2-13, `ramsread_h5_ColdPools_luzon_XXX_b.m` might save data from output times 14-25, et cetera. It is necessary to convert microphysical budget variables into 5-minute increments using 12-output time chunks starting with the output time corresponding to time XX:05.
    4. This process creates two sets of data for each simulation: one set on a 125-m grid, and one set on a 250-m grid. It is okay to delete the 125-m data set once the 250-m data set has been created in order to save storage space.
    5. Once the .mat files have been created, cold pools can be analyzed. The below steps create individual files for each cold pool containing snapshots of the model fields (at the vertical levels corresponding to `k_to_keep_cylindrical_polar` in `misc_setup_unified_pad_luzonXXX_001.m`, where $k = 2$ corresponds to the lowest above-ground model level):
       a. Determine the model fields you are desiring to save for each cold pool for potential compositing. The lists are specified in six separate files, one pair for each type of data:
          - 3D atmospheric variables: `fetch_fields_to_process.m`, `data_field_info_combined.m`
          - 2D variables that are specified separately for each "patch" (see RAMS/LEAF-3 documentation):

fetch_fields_to_process_np.m,
data_field_info_combined_np.m
- 2D variables: fetch_fields_to_process_sfc.m,
data_field_info_combined_sfc.m

b. Run initializing_cold_pools_3D_2Dstart_pad_luzonXXX_001.m with the following settings chosen in misc_setup_unified_pad_luzonXXX_001.m:
- need_to_save_full_domain_data = 1;
- need_to_find_cold_pools = 1;
- need_to_update_cylindrical_polar_data_settings = 0;
- need_to_compute_sdf = 0;
- need_to_save_polar_data_attempt_1 = 1;
- need_to_save_polar_data_attempt_2 = 0;
- need_to_save_radial_profiles = 0;
- need_to_save_radial_profiles_sfc = 0;
- need_to_save_radial_profiles_np = 0;
- need_to_save_radial_profiles_nonorm = 0;
- need_to_track_backward = 0;
- need_to_make_repairs = 0;

c. Run tracking_cold_pools_3D_2Dstart_pad_luzonXXX_001.m

d. Run initializing_cold_pools_3D_2Dstart_pad_luzonXXX_001.m with the following settings chosen in misc_setup_unified_pad_luzonXXX_001.m:
- need_to_save_full_domain_data = 0;
- need_to_find_cold_pools = 0;
- need_to_update_cylindrical_polar_data_settings = 0;
- need_to_compute_sdf = 0;
- need_to_save_polar_data_attempt_1 = 0;
- need_to_save_polar_data_attempt_2 = 1;
- need_to_save_radial_profiles = 1;
- need_to_save_radial_profiles_sfc = 1;
- need_to_save_radial_profiles_np = 1;
- need_to_save_radial_profiles_nonorm = 0;
- need_to_track_backward = 0;
- need_to_make_repairs = 0;

6. By the end of the previous step, each cold pool file contains a copy of the model fields centered on the cold pool center. This includes model fields calculated based on the cold pool center, such as the radial component of velocity. Each file also contains the azimuthally averaged fields for that cold pool. The next step is to create composites and calculate statistics by combining the data across files. This is done separately for cold pool area, normalized composites, and nonnormalized composites:
   a. Cold pool area: cold_pool_object_averaging_areas_exact_luzon_sm_001c.m
      - First, set compute_averages = 1 and run the script. This will compute the averages. Then, once this is done, set compute_averages = 0 and run the script to create plots.
   b. asdfkjhl

7. sadfslhk

- **RAMSModel**
  - ConradFiles
    - PBSScripts
      - Portable Batch System (PBS) scripts for running RAMS on the ONR Conrad machine.
      - Project number has been redacted.
      - See “MATLABAnalysisScripts” for file naming convention.
    - include.mk
      - Settings for compiling RAMS on the ONR Conrad machine.
      - Before compiling RAMS, must enter in the command line:
        - `module swap craype-target-native craype-haswell`
        - `module swap PrgEnv-cray PrgEnv-intel`
  - ModifiedRAMSSourceCode
    - rams_20180508_release_6.2.08/src/6.2.08/micro/mic_driv.f90
      - This version contains a simple bug fix (missing parentheses) for calculating the LATHEATFRZ diagnostic field; this bug was fixed in release 6.2.09 and does not affect the model integration.
    - rams_20180508_release_6.2.08/src/6.2.08/surface/ruser.f90
      - This code sets the NDVI to a uniform value of 0.6.
  - Namelists
    - Model settings for all simulations.
    - See “MATLABAnalysisScripts” for file naming convention.
    - See RAMS model documentation (available in rams_20180508_release_6.2.08.tar.gz; see “docs” directory) for information about namelist parameters.
  - rams_20180508_release_6.2.08.tar.gz
    - This tarfile contains the model and model documentation, including instructions for compiling on various platforms.
    - The latest version of RAMS (6.2.11 as of 10 October 2019) can be downloaded at the following link:
      - https://vandenheever.atmos.colostate.edu/vdhpage/rams.php