T&C-2D: distributed modeling

Material:

- **Source Code and Pre-Processing Tools:**
  - T&C source code
  - Scripts for DEM pre-processing
  - demo parameter file
- **Input Data**:
  - Files containing CO_2 air concentration.
- **Reference Materials:**
  - Parameter Lookup File: *T&C_Variables_LIST_PlotScale.pdf*
  - Technical Reference: *T&C model technical reference.pdf*
- **Visualization and Validation Tools:**
  - GRAPHY_TEC and GRAPHY_TS files for results visualization and model validation.
- **Supporting Data:**
  - Flux tower data from the first exercise.

Objective: This exercise is designed to help students learn how to prepare input files and execute a basic demo study using the T&C-2D model (Fatichi et al., 2012). It provides guidance on the essential steps, including pre-processing DEM input data, reviewing critical model settings, understanding the model's inputs and outputs, and running the simulation in a virtual catchment.

1. Prepare the digital elevation input file

In addition to the meteorological input data prepared in the previous exercise, spatial information of the study area is crucial for running the T&C-2D model. To begin, navigate to the folder **DEM_Preprocessing** and open the file **PreprocessDEMexample.m**. This script allows the pre-processing of raw digital elevation model (DEM) data to calculate various geomorphological and hydrological properties (e.g., soil texture, slope, water accumulation area) using the TopoToolbox. It also includes vegetation and soil spatial information. For now, the script loads a virtual catchment from Remondi et al. (2019) (‘CV_domain.mat’, see Figure 1) and generates the necessary topographic file to run the T&C-2D model. If you wish to customize the study area, you can replace the provided DEM with data from your own study area, along with its corresponding soil and vegetation spatial information. This will allow you to generate the necessary spatial input file for running the T&C-2D model.

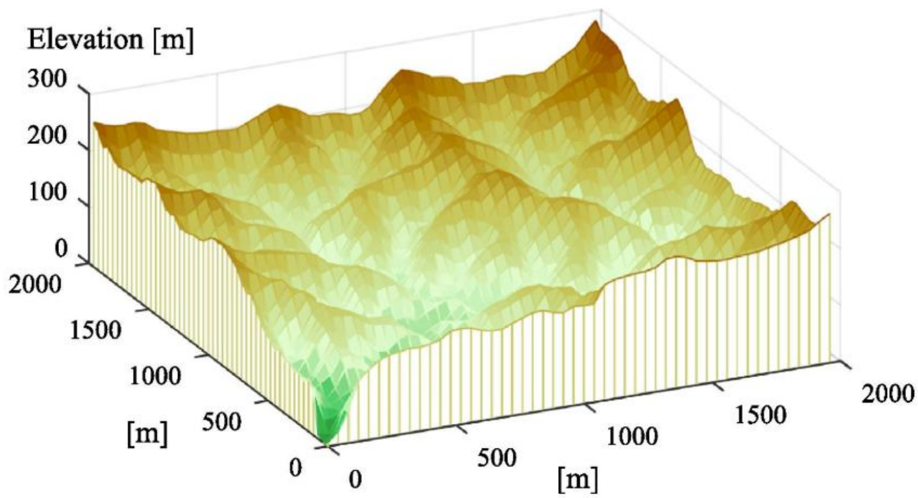

Figure 1. 3D surface showing the provided virtual catchment (figure from Remondi et al., 2019).

Reflect on the study site and the catchment characteristics:

- *What is the size of the topography? How many computation cells does it have?*
- *What is the spatial resolution?*
- *How many types of vegetation are there?*
- *What are the soil properties in the domain? (It can be spatially distributed if soil texture map is available, such as from Gupta et al., 2024)*

2. Understand the case study

As for the 1D application, understanding the case study is essential before running simulations. Open the file **PARAMETER_ALL_demo.m**, which contains most of the vegetation properties used in the model. In this file, many parameters are defined with a size of 2 instead of 1, reflecting multiple land cover types (see Figures 2 and 3).

```
zatm =[30.0 3.0 ] ; %% Reference Height
zatm = zatm(ANSWER);
```

Figure 2. An example of how 2D T&C treat spatially vary inputs.

In Figure 3, the variable **ANSWER** represents the land cover type, which is loaded from your digital elevation input file (as 'ksv' or 'VEG_CODE'). Correspondingly, there are multiple sets of vegetation parameters, determined by the value of **ANSWER**. For example:

- If **ANSWER = 1**, the land is 100% covered by vegetation, consisting only of high vegetation. The root depth of high vegetation at its 95th percentile is 400 mm, while the root depth for low vegetation is set to 0.
- If **ANSWER = 2**, the land is also 100% covered by canopy but consists only of low vegetation. In this case, the root depth at its 95th percentile is 250 mm.

This parameterization enables the model to differentiate vegetation properties based on the land cover type, enhancing the realism and accuracy of simulations. The setting of ANSWER depends on the landcover types in the domain and can be changed with flexibility, but it is suggested to set only one vegetation type in one cell (Ccrown=1).

```
switch ANSWER
     %case {1,2,3,8,9}
    case {1}
        %%%% LAND COVER PARTITION  100  Sitka Spruce
        Cwat = 0; Curb = 0.0 ; Crock = 0.0;
        Cbare = 0.0; Ccrown = [1.0];
        cc=length(Ccrown);%% Crown area
        II = [ 1 0 ]>0;
     %case 6
    case {2}
        %%%% LAND COVER PARTITION 100 GRassland
        Cwat = 0; Curb = 0.0 ; Crock = 0.0;
        Cbare = 0.0;
        % Ccrown = [0 1];
        Ccrown = [1.0];
        cc=length(Ccrown);%% Crown area
        II = [0 1]>0;
    otherwise
        disp('INDEX FOR SOIL VEGETATION PARAMETER INCONSISTENT')
        return
end

ZR95_H = [400 0 ]; %% [mm]
ZR95_L = [0 250 ]; %% [mm]
ZR50_H = [NaN NaN];
ZR50_L = [NaN NaN ];
ZRmax_H = [NaN NaN ];
ZRmax_L = [NaN NaN ];
ZR95_H =ZR95_H(II); ZR50_H =ZR50_H(II); ZRmax_H =ZRmax_H(II);
ZR95_L =ZR95_L(II); ZR50_L =ZR50_L(II); ZRmax_L =ZRmax_L(II);
```

Figure 3. Interpretation of parameters settings in 2D T&C.

3. Understand the structure of the model

The file **GO_TeC_Demo.m** is the main script to run the model. Open it to explore its coding structure. Most of the code is self-explanatory, and this document highlights only some key elements:

- **Directories and files**. In the current example, the directories and files are properly set and the model can be simply run. You can define the folder title to save the result in 'TITLE_SAVE', and remember to make sure all the input files are in the correct working directories in your future work.
- **tstore**. tstore defines the time steps at which spatial average maps are saved. The model calculates spatial maps for various variables (e.g., temperature, leaf area index (LAI)) at every time step. The value of tstore specifies the interval over which these variables are averaged. For example, tstore = [40] means each cell will have an average value of variables (e.g., temperature, LAI) between the start of the simulation and time step 40. This will result in spatial maps where the values are averaged over the time steps specified by the intervals in tstore.
- **Lapse rate (grad)**. Lapse rate is used in T&C-2D to spatially distributed meteorological forcing – for example, the temperature will decrease with increasing elevation. The rates can be also derived from two meteorological stations if they are available near the study area.
- **DeltaGMT, Lon, Lat**. Update these fields to reflect the location information of the meteorological station you are using.
- **ms_max**. Ensure that the value of ms_max matches the corresponding value in the parameter file.
- **VEG_CODE**. This represents the landcover or vegetation cover used in the model. In the case study, it has been simplified to include only two species, as shown in Figure 4.

```
%%% 1 WOOD
%%% 2 grass and shrub
%%% simplified to only two vegetaion cover
VEG_CODE(VEG_CODE==1)=1;
VEG_CODE(VEG_CODE==2)=2;
VEG_CODE(VEG_CODE==3)=2;
VEG_CODE(VEG_CODE==4)=2;
ksv=reshape(VEG_CODE,num_cell,1);
Ccrown_OUT =[ 1  ;  1 ];  %% Ccrown fraction for PFT
EVcode = [ 1 2];  %% code of each PFTs
```

Figure 4. Simplification of vegetation cover in the GO_Tec_Demo file

Some initial values are given in the GO file, for example in Figure 5.

```
%%% Evergreen
LAI_Htm1(ksv==1,1)=3.98;
B_Htm1(ksv==1,1,1)= 361.6316;
B_Htm1(ksv==1,1,2)= 980.8489;
B_Htm1(ksv==1,1,3)= 514.8147;
B_Htm1(ksv==1,1,4)= 53.6371;
B_Htm1(ksv==1,1,5)= 11.4537;
B_Htm1(ksv==1,1,6)= 1.4086e+04;
B_Htm1(ksv==1,1,7)= 14.7795;
AgeL_Htm1(ksv==1,1)=921.7089;
SAI_Htm1(ksv==1,1)=0.20;
hc_Htm1(ksv==1,1)=19;

%%% grass
LAI_Ltm1(ksv==2,1)=2.4;
B_Ltm1(ksv==2,1,1)= 68.5080;
B_Ltm1(ksv==2,1,2)= 0;
B_Ltm1(ksv==2,1,3)= 385.6041;
B_Ltm1(ksv==2,1,4)= 287.1935;
B_Ltm1(ksv==2,1,7)= 133.6411;
dflo_Ltm1(:,1)=0;
AgeL_Ltm1(:,1)=208;
SAI_Ltm1(ksv==2,1)=0.001;
hc_Ltm1(ksv==2,1)=0.16;
```

Figure 5. Initial values of plant properties in the GO_Tec_Demo file.

- **OPT_SoilBiogeochemistry**. This controls the soil biogeochemistry module. Ensure that it is set to 0 to keep the module turned off.
- **Time iteration loop**. The loop for time iteration begins with: t = 2:N_time_step
- **Parallel computation of 1D T&C**. Parallel computation is initiated with: parfor (ij = 1:num_cell). This requires the Parallel Computing Toolbox, and allows MATLAB to efficiently run 1D T&C in each cell and then run a water routing module to redistribute soil water. Within this loop, **PARAMETERS_ALL_demo** uses the value of **ksv** as the **ANSWER** to determine the vegetation parameters.
- **ROUTING_MODULE**. At each time step, the ROUTING_MODULE routes water after the completion of all 1D simulations.
- **OUTPUT_MANAGER_PAR_BG_pix**. The command `run('OUTPUT_MANAGER_PAR')` writes the simulation results in your result saving folder.

4. Ready to run the simulation

Ensure you are in the correct working directory, as shown in Figure 6. Before running the model, verify that the simulation's start and end dates are properly set, and double-check the model's latitude and longitude settings and time period for simulation, as illustrated in Figure 7. Once everything is correctly configured, the model is ready to run.

Figure 6. Correct working path

```
%%%% Set the start and end day
dateNum1 = datenum(2005, 5, 1, 0, 0, 0);
dateNum2 = datenum(2005, 10, 3, 23, 0, 0);
index1 = find(Date >= dateNum1, 1, 'first');
index2 = find(Date >= dateNum2, 1, 'first');
x1=index1+hot_spot; %%%%
x2= index2;
```

Figure 7. An example to set up the simulation period

After the simulation is completed, you can find the results in the folder you specified during setup. Note that the variables in the MATLAB workspace represent the state of these variables at the last time step of the simulation. Be prepared for the runtime to vary depending on the length of the simulation and the complexity of the setup. For instance, as shown in Figure 8, simulating approximately five months took almost four hours to complete.

```
Iter:
        3743

COMPUTATIONAL TIME [h]
    3.9442

 COMPUTATIONAL TIME [s/cycle]
    3.7935

IdleTimeout has been reached.
Parallel pool using the 'Processes' profile is shutting down.
```

Figure 8. An example of running time report for a simulation lasting approximately 5 months.

5. Checking the results

To quickly inspect the simulation results, you can use the **GRAPHY_TEC** and **GRAPHY_TS** files. These scripts provide an efficient way to visualize the outputs:

- **GRAPH_TEC**. This file is used to plot spatial maps. Open the file, load the corresponding .mat file, and run the script to generate spatial visualizations.
- **GRAPH_TS**. This file is used to plot time series of spatially averaged variables. These variables are averaged over the entire domain at each time step. Open the file, load the relevant .dat file, and plot the time series of the variables of interest.

These quick checks allow you to verify the trends, patterns, and spatial distributions of key variables like temperature, soil moisture, or LAI. Adjustments can be made if the results deviate significantly from expectations. Some example results are presented in Figure 9.

During the exercise session:

- Run the simulation for a period of ~3 days (it may take half an hour on your laptop).
- Based on your simulation results. Try the GRAPHY_TEC and GRAPHY_TS functions, get similar plot as Figure 9. ***Note that it is necessary to change the files to load in these two functions, you need to open the script and edit. If it takes too long in your computer to get simulation results, several demo simulation results are also provided*.**

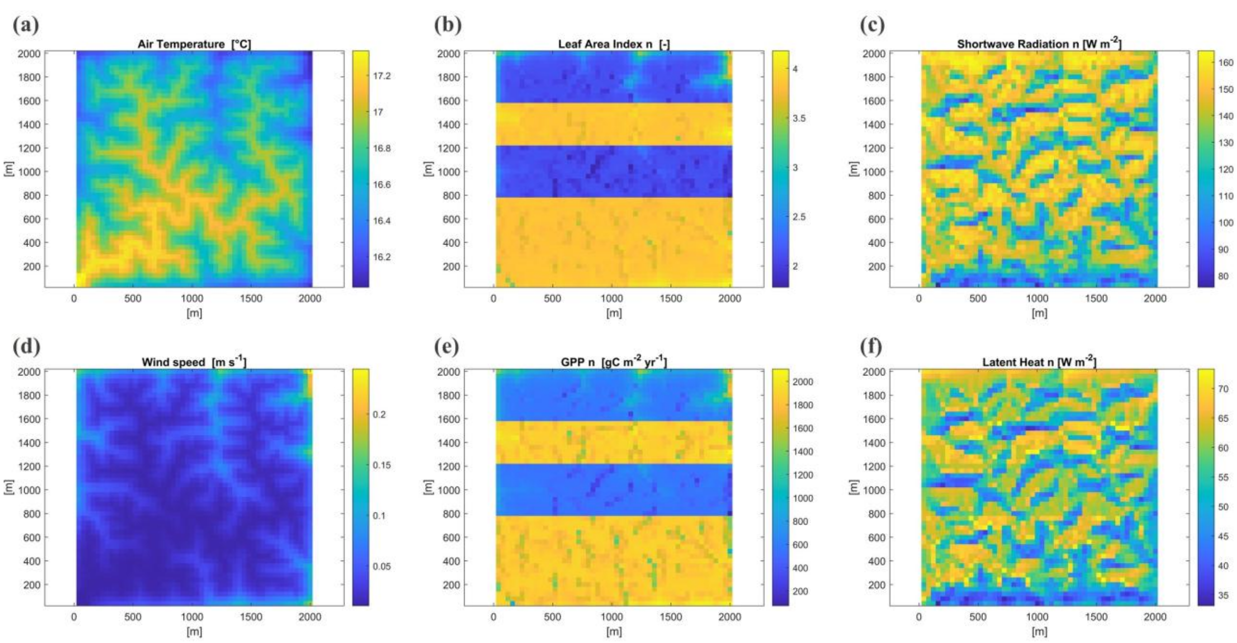

Figure 9. Example results from the 5 months simulation. (a) Air temperature. (b) Leaf area index. (c) Shortwave radiation. (d) Wind speed. (e) Gross primary productivity. (f) Latent heat.

Find out more

Laboratory of Catchment Hydrology and Geomorphology: https://www.epfl.ch/labs/change/

T&C GitHub source code : https://github.com/simonefatichi/TeC_Source_Code

References

Fatichi, S., Ivanov, V. Y., & Caporali, E. (2012). A mechanistic ecohydrological model to investigate complex interactions in cold and warm water-controlled environments: 2. spatiotemporal analyses. *Journal of Advances in Modeling Earth Systems, 4*(2) doi:https://doi.org/10.1029/2011MS000087

Remondi, F., Botter, M., Burlando, P., & Fatichi, S. (2019). Variability of transit time distributions with climate and topography: A modelling approach. *Journal of Hydrology*, 569, 37-50.

Gupta, S., Hasler, J. K., & Alewell, C. (2024). Mining soil data of Switzerland: New maps for soil texture, soil organic carbon, nitrogen, and phosphorus. *Geoderma Regional*, 36, e00747.