paper.md

wrfup: A Python Package to Enhance Urban Climate Modeling in WRF

Author:
Jacobo Gabeiras
Univ. Grenoble Alpes, CNRS, Grenoble INP, LEGI, Grenoble, 38000, France
Email: Jacobo.Gabeiras-Penas@univ-grenoble-alpes.fr

Abstract

wrfup is a Python-based tool designed to improve urban climate modeling in the Weather Research and Forecasting (WRF) model. By dynamically calculating key urban canopy parameters, wrfup enhances the precision of urban weather simulations. The package facilitates ingestion of high-resolution urban morphology data directly into WRF's geo_em files, including crucial fields like **URB_PARAM** and **FRC_URB2D**. These parameters are vital for advanced urban modeling schemes such as SLUCM, BEP, and BEP+BEM. wrfup simplifies the workflow, offering an accessible and efficient method for preparing urban data for WRF simulations via terminal commands.

Here's an updated version that addresses the key points you mentioned, while incorporating the correct order and clarity of ideas:

Statement of Need

Accurately simulating urban climate and weather phenomena, such as the **urban heat island (UHI) effect**, altered wind patterns, and precipitation dynamics, is critical for improving local climate predictions in the context of urbanization. These phenomena can significantly influence public health, energy consumption, and urban planning. The **Weather Research and Forecasting (WRF)** model offers advanced urban parameterizations—**SLUCM (Single-Layer Urban Canopy Model)**, **BEP (Building Effect Parameterization)**, and **BEP+BEM (Building Energy Model)** [@bepplusbem]—which enable precise simulations of these urban effects. However, these parameterizations depend heavily on high-quality data about urban morphology, including building heights, building fraction and others.

One major challenge lies in obtaining detailed, city-specific urban data. High-resolution datasets like **LiDAR** provide accurate urban morphology data but are often difficult to process, complex to integrate into models, and not universally available, especially in developing cities. The **WUDAPT (World Urban Database and Access Portal Tools)** [@EUWUDAPT] project addresses this gap by offering the **Local Climate Zone (LCZ)** framework, which generalizes urban areas into distinct zones based on physical characteristics. The **LCZ Generator** [@lczgenerator] and **W2W (WUDAPT to WRF)** [@w2w] further enhance and facilitate the integration process of LCZ data into the WRF model. However, this approach relies on generalized table values rather than specific, detailed urban structure data, which limits the accuracy of the simulations.

wrfup offers a more tailored solution by allowing users to ingest real-world urban data directly into WRF's **geo_em** files. Unlike the LiDAR approach, which can be time-consuming and difficult to handle, wrfup provides a much faster workflow—achievable within minutes—and offers a higher level of accuracy compared to WUDAPT's generalized approach. By dynamically calculating critical urban morphological parameters, wrfup supports advanced urban parameterizations like SLUCM, BEP, and BEP+BEM, enabling detailed urban canopy simulations that reflect the specific characteristics of each city. Furthermore, wrfup is compatible with W2W, enhancing the capability to simulate urban environments in a way that is both accurate and accessible.

Software Description

Functionality

The tool works by calculating and ingesting the fields **URB_PARAM** and **FRC_URB2D**, which contain critical information for representing urban surfaces in the WRF model. **URB_PARAM** is a key field in the WRF urban models, containing information like the **Plan Area Fraction** and **Building Height Distribution**, while **FRC_URB2D** represents the fraction of urban land in each grid cell.

Code Structure

The **wrfup** package is structured around modules that handle downloading data, calculating necessary fields, and ingesting them into WRF's **geo_em** files.

- **Main Module**: The entry point of the package, which allows users to run commands from the terminal. It simplifies tasks such as ingesting the **URB_PARAM** and **FRC_URB2D** fields with one-line commands.
- **Download Module:** Automates the process of downloading urban morphology data for a given area of interest (AOI). Data from sources like the **World**

Settlement Footprint 3D (WSF3D) [@wsf3d] and **Urban Fraction** [@urban_fraction] datasets are used for these calculations.

- **Calculation Module**: Responsible for calculating urban parameters such as:
  - **Plan Area Fraction (LAMBDA_P)** stored in slice [90,:,:]
  - **Mean Building Height** stored in slice [91,:,:]
  - **Weighted Building Height** stored in slice [93,:,:]
  - **Frontal Area Fraction (LAMBDA_B)** stored in slice [94,:,:]
  - **Frontal Area Index** stored in slices [96-99,:,:] for different wind directions
  - **Building Height Distribution** stored in slices [117:132,:,:]
- **Utility Module**: Handles tasks such as cleaning temporary files, verifying geo_em file integrity, and managing the output of modified **geo_em** files.

Initial Data Requirements

To use **wrfup** effectively, the primary requirement is a **geo_em.dXX.nc** file for the specific inner domain of the WRF model. This file is typically generated by the WRF Preprocessing System (WPS) using **geogrid.exe**, following the standard procedure without needing any additional modifications to the **namelist.wps** file.

In cases where the **URB_PARAM** or **FRC_URB2D** fields are missing from the **geo_em** file, wrfup automatically creates these fields. It follows the required attributes and structure within the WRF framework to ensure compatibility with urban canopy models.

Workflow Overview

The entire process for preparing and modifying urban canopy parameters for the Weather Research and Forecasting (WRF) model using **wrfup** is handled through a single terminal command. This command allows the user to update the **geo_em.d0X.nc** file by calculating and ingesting the necessary urban parameters, depending on the specified target (either **URB_PARAM** or **FRC_URB2D**).

The execution involves the following:

```
wrfup geo_em.d0X.nc URB_PARAM --work_dir YOUR_DIRECTORY
```

or

```
wrfup geo_em.d0X.nc FRC_URB2D --work_dir YOUR_DIRECTORY
```

The tool updates the **geo_em** file, calculating the urban canopy parameters and generating a new modified output file (e.g., **geo_em_URB_PARAM.nc** or **geo_em_FRC_URB2D.nc**), ready for subsequent steps in the WRF modeling workflow.

This streamlined approach eliminates the need for manual data handling and complex processing, allowing researchers to quickly integrate real-world building data into WRF simulations.

Yes, the **Integration in WRF's Preprocessing** section is a great place to include the workflow that highlights how **wrfup** fits into the WRF model pipeline.

Here's an updated version of the section including the workflow:

Integration in WRF's Preprocessing

The **wrfup** tool is designed to work seamlessly with the Weather Research and Forecasting (WRF) Preprocessing System (WPS). The workflow integrates **wrfup** with WRF's preprocessing tools, ensuring that the calculated urban canopy parameters are correctly incorporated into the **geo_em.d0X.nc** files.

1. First, **geogrid** generates the initial **geo_em.d0X.nc** file that contains the necessary geographical data for WRF.
2. Using **wrfup**, the urban canopy parameters are calculated and added to the file. If the **URB_PARAM** or **FRC_URB2D** fields are not present in the file, **wrfup** creates them following the required attributes of the WRF framework. The output file, such as **geo_em_URB_PARAM.d0X.nc**, contains the detailed urban morphology required for accurate simulations.
3. This file can be renamed as **geo_em.d0X.nc** or passed directly to **metgrid** for further processing.
4. The **metgrid** tool integrates the updated **geo_em** file with meteorological data, preparing the input files for the WRF simulation.
5. Finally, these preprocessed files are used by the **real** and **wrf** executables to run the actual weather or climate simulation, incorporating precise urban morphology data into the model.

Here's a visual representation of the integration process:

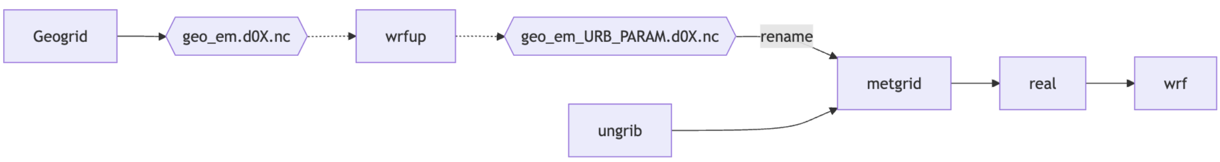

Figure 1. Workflow illustrating how wrfup integrates building data into the WRF model. The tool calculates urban parameters such as Plan Area Fraction, Mean Building Height, and Total Building Area, and modifies the geo_em.dXX.nc files for use with Metgrid and WRF simulations.

Here’s a refined version of the **Important Notes** section:

Important Notes

When using **wrfup** to modify **geo_em** files and integrate urban morphological data into WRF simulations, there are several key considerations to keep in mind:

- **Compatibility with WRF Versions**: Ensure that your WRF version supports the **URB_PARAM** and **FRC_URB2D** fields. **wrfup** requires **WPS version 3.8 or higher** for full compatibility with these urban fields and urban parameterization schemes like **SLUCM**, **BEP**, and **BEP+BEM**. Earlier versions may not properly integrate the urban parameters needed for accurate simulations.

- **Handling Large Datasets**: For large urban areas or high-resolution data, it's essential to ensure your system has adequate memory and processing power. Processing large cities or dense urban regions may require significant computational resources. In such cases, you may need to work in smaller sections or reduce the resolution of the data to ensure smoother processing.

- **Creating and Naming Output Files**: After modifying the **geo_em** files, it is important to adhere to WRF’s naming conventions. For example, rename **geo_em_modified.nc** back to **geo_em.dXX.nc** before running the **metgrid.exe** step in WPS. Incorrect file names may cause issues in the preprocessing stages, leading to potential errors during the WRF simulation.

- **Inspecting Updated URB_PARAM Fields**: It is recommended to validate the correctness of the updated **URB_PARAM** fields by using visualization tools like **xarray** or **ncview**. This step ensures that the calculated urban morphological parameters are correctly written into the **geo_em** files before proceeding with further WRF preprocessing steps.

References

- **Marconcini, M., Esch, T., Metz, A., et al.** (2021). *World Settlement Footprint 3D - A first three-dimensional survey of the global building stock*. ResearchGate. Link

- **Pablo d'Entremont, et al.** (2022). *Urban Fraction Data*. Zenodo. Link

- **Matthias Demuzere, Daniel Argüeso, Andrea Zonato, Jonas Kittner** (2022). *W2W: A Python package that injects WUDAPT's Local Climate Zone information in WRF*. Journal of Open Source Software, 7(76), 4432. Link