# LorenzCycleToolkit: A Comprehensive Python Tool for Analyzing Atmospheric Energy Cycles

Danilo Couto de Souza [1], **Pedro Leite da Silva Dias** [1], **Matheus Bonjour Laviola da Silva** [2], **and Ricardo de Camargo** [1]

1 Institute of Astronomy, Geophysics and Atmospheric Sciences of the São Paulo University, Rua do Matão, 226, Cidade Universitária, 05508-090, São Paulo, Brazil **2** OceanPact, Rua da Glória, 122, Rio de Janeiro, 20241-180, Rio de Janeiro, Brazil

Summary

The `LorenzCycleToolkit` is a Python package designed for the calculation and analysis of the Lorenz Energy Cycle (LEC) in specified atmospheric regions. Originally proposed by Edward Lorenz in 1965, the LEC provides a comprehensive framework for understanding the transformation and distribution of energy within the atmosphere, particularly focusing on the zonal and eddy components of kinetic and available potential energy. This toolkit enables researchers to study these energy transformations in various atmospheric phenomena, thereby enhancing our understanding of atmospheric dynamics and improving weather and climate prediction models.

The `LorenzCycleToolkit` supports both Eulerian and semi-Lagrangian frameworks, allowing for flexibility in the analysis of atmospheric data. It can handle reanalysis or model data provided on isobaric levels, and outputs results in the CSV format for easy analysis. Additionally, the toolkit offers options for automatic plot generation, including spatial representations of the computational domain, time series, Hovmöller diagrams, and complete energy flux boxes for the LEC. This package is a valuable tool for meteorologists and climate scientists aiming to diagnose and improve the performance of numerical weather prediction models and to study the impacts of climate change on atmospheric energy processes.

Background

The Lorenz Energy Cycle (LEC), introduced by E. N. Lorenz (1955), is a framework to describe the processes of energy conversion and transfer within the Earth's atmosphere. It serves as a diagnostic tool for the general circulation of the atmosphere, providing insights into its dynamic and thermodynamic structures, and information about atmospheric flows and the transport of heat, moisture, and angular momentum. The LEC divides atmospheric energy into four main reservoirs: available potential energy (APE), kinetic energy (K), and their respective zonal mean (Z) and eddy (E) components. The LEC can indicate how energy is generated, converted, and dissipated within these reservoirs.
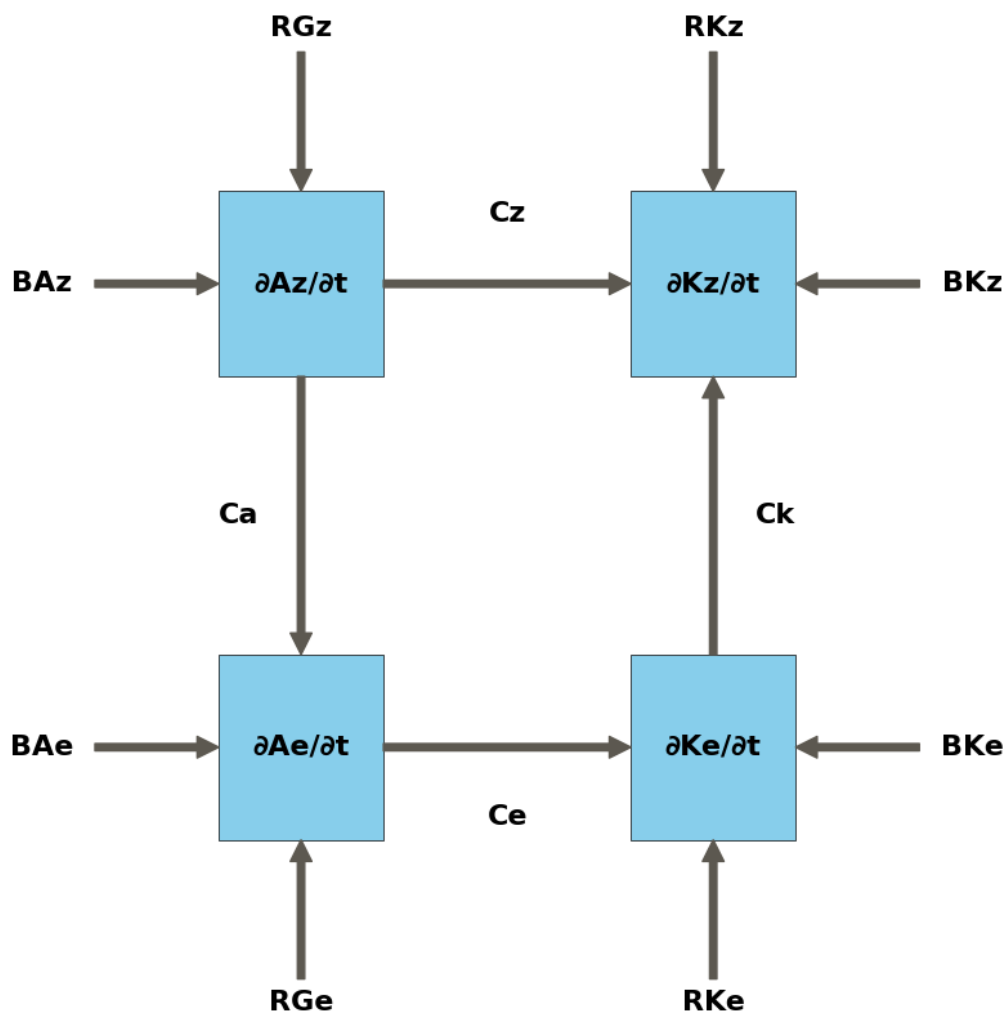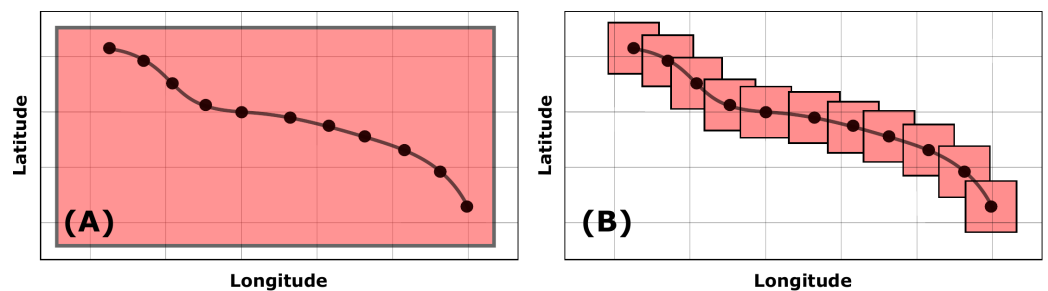
Figure 1: Schematic representation of the Lorenz Energy Cycle. The figure shows the four main energy reservoirs in the atmosphere: zonal available potential energy (A_Z), eddy available potential energy (A_E), zonal kinetic energy (K_Z), and eddy kinetic energy (K_E). The arrows indicate the energy transformations between these reservoirs: C_Z (conversion from A_Z to K_Z), C_E (conversion from A_E to K_E), C_A (conversion between A_Z and A_E), and C_K (conversion between K_Z and K_E). Additional terms include B (boundary fluxes), R (residuals), G (generation terms), and D (dissipation terms).

Applications of the LEC are diverse and significant in meteorology and climate science. It aids in diagnosing the performance of numerical weather prediction models by comparing simulated and observed energy cycles (Ulbrich & Speth, 1991). Moreover, it is used to study climate change impacts, such as shifts in energy transfer processes due to global warming (Li et al., 2007; Veiga & Ambrizzi, 2013). Understanding the LEC is also crucial for improving the representation of atmospheric processes in general circulation models (GCMs), which are essential for accurate climate projections (Boer & Lambert, 2008; E. Lorenz, 1967).

Statement of Need

`LorenzCycleToolkit` is a Python package designed for calculating the Lorenz Energy Cycle (LEC) in limited atmospheric regions, supporting both Eulerian and semi-Lagrangian frameworks. The formulations used are the same as in Michaelides et al. (1999). The Eulerian framework computes the LEC for a fixed region in the atmosphere (Dias Pinto & Da Rocha, 2011), requiring users to provide a "box limits" file containing the minimum and maximum latitudes

and longitudes for the desired computational domain. The semi-Lagrangian framework, on the other hand, uses a domain that changes position with time (Michaelides et al., 1999), which is especially useful for studying cyclonic systems. For this option, users must specify the central position of the domain for each timestep in a "track" CSV file, which can also specify the domain width and length. If these specifications are not provided, a default 15°×15° domain is used. The `LorenzCycleToolkit` also offers an option for users to interactively choose the domain position and size for each data timestep.

Figure 2: Schematic representation of the Eulerian and semi-Lagrangian frameworks. On the left (A), the Eulerian framework is shown with a fixed domain capturing the evolution of a cyclone as it moves through the grid. On the right (B), the semi-Lagrangian framework is illustrated, which tracks the cyclone with a moving domain that follows the cyclone's path. For each time step, the domain shifts to remain centered on the cyclone. The latitude and longitude axes are shown for spatial reference.

The program can handle any reanalysis or model data provided on isobaric levels. Users must adjust the "namelist" file, specifying the names for each variable and coordinate in the input file. The requirements for the input file are that it must be in NetCDF format (Rew & Davis, 1990), the grid must be structured, and it must have isobaric levels as vertical coordinates. These specifications are common in reanalysis, climate, and weather datasets.

The program outputs the results for each term in comma-separated values (CSV) files, generating one CSV file for each term. For energy, conversion, and generation terms, it also provides results for each vertical level, enabling two-dimensional analysis. Additionally, the program can be configured by the user to automatically generate plots. These plots include spatial representations of the computational domain defined for the analysis, time series of the system's central minimum pressure and relative vorticity at 850 hPa, time series of each LEC term, Hovmöller diagrams, and complete energy flux boxes for the LEC.

Acknowledgements

External Libraries Used

The NetCDF data is handled by Xarray (Hoyer & Hamman, 2017), with Dask optimizing the handling of large datasets (Daniel, 2019). This combination allows for efficient slicing of large data chunks, better memory management, and improved performance when dealing with extensive data in NetCDF files. Pandas DataFrames (Reback et al., 2020) are used for storing output results and saving them as CSV files. Numpy is utilized for angle conversion

and computation procedures (Harris et al., 2020). For handling meteorological constants and variable units, the program uses the MetPy package (May et al., 2022).

References

Boer, G. J., & Lambert, S. (2008). The Energy Cycle in Atmospheric Models. *Climate Dynamics*, *30*, 371–390. https://doi.org/10.1007/s00382-007-0303-4

Daniel, J. (2019). *Data Science with Python and Dask*. Simon; Schuster. ISBN: 9781638353546

Dias Pinto, J. R., & Da Rocha, R. P. (2011). The Energy Cycle and Structural Evolution of Cyclones over Southeastern South America in Three Case Studies. *Journal of Geophysical Research: Atmospheres*, *116*(D14). https://doi.org/10.1029/2011JD016217

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N. J., & others. (2020). Array Programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Hoyer, S., & Hamman, J. (2017). xarray: ND Labeled Arrays and Datasets in Python. *Journal of Open Research Software*, *5*(1), 10–10. https://doi.org/10.5334/jors.148

Li, L., Ingersoll, A. P., Jiang, X., Feldman, D., & Yung, Y. L. (2007). Lorenz Energy Cycle of the Global Atmosphere Based on Reanalysis Datasets. *Geophysical Research Letters*, *34*(16). https://doi.org/10.1029/2007GL029985

Lorenz, E. (1967). The Nature and Theory of the General Circulation of the Atmosphere. *World Meteorological Organization*, *161*. ISBN: 9788873954736

Lorenz, E. N. (1955). Available Potential Energy and the Maintenance of the General Circulation. *Tellus*, *7*(2), 157–167. https://doi.org/10.3402/tellusa.v7i2.8796

May, R. M., Goebbert, K. H., Thielen, J. E., Leeman, J. R., Camron, M. D., Bruick, Z., Bruning, E. C., Manser, R. P., Arms, S. C., & Marsh, P. T. (2022). MetPy: A Meteorological Python Library for Data Analysis and Visualization. *Bulletin of the American Meteorological Society*, *103*(10), E2273–E2284. https://doi.org/10.1175/BAMS-D-21-0125.1

Michaelides, S. C., Prezerakos, N. G., & Flocas, H. A. (1999). Quasi-Lagrangian Energetics of an Intense Mediterranean Cyclone. *Quarterly Journal of the Royal Meteorological Society*, *125*(553), 139–168. https://doi.org/10.1002/qj.49712555309

Reback, J., McKinney, W., Van Den Bossche, J., Augspurger, T., Cloud, P., Klein, A., Hawkins, S., Roeschke, M., Tratner, J., She, C., & others. (2020). pandas-dev/pandas: Pandas 1.0.5. *Zenodo*. https://doi.org/10.5281/zenodo.3898987

Rew, R., & Davis, G. (1990). NetCDF: An Interface for Scientific Data Access. *IEEE Computer Graphics and Applications*, *10*(4), 76–82. https://doi.org/10.1109/38.56302

Ulbrich, U., & Speth, P. (1991). The Global Energy Cycle of Stationary and Transient Atmospheric Waves: Results from ECMWF Analyses. *Meteorology and Atmospheric Physics*, *45*, 125–138. https://doi.org/10.1007/bf01029650

Veiga, J. A. P., & Ambrizzi, T. (2013). A Global and Hemispherical Analysis of the Lorenz Energetics Based on the Representative Concentration Pathways Used in CMIP5. *Advances in Meteorology*, *2013*(1), 485047. https://doi.org/10.1155/2013/485047