Lowering the Cost of Climate Research

Energy Consumption vs Clock Speed for Various Application Profiles

Spencer Diamond, *Arizona State University, SIParCS*
Dave Hart, Rory Kelly, Ben Matthews, *NCAR*

July 27, 2021

This material is based upon work supported by the National Center for Atmospheric Research, which is a major facility sponsored by the National Science Foundation under Cooperative Agreement No. 1852977.

Motivation

- Lower power bill: utility costs for the NCAR Wyoming Supercomputing Center are about $1.4 million per year

- Every Joule used increases our environmental impact

Considerations and Scope

Energy Consumption Vectors

- Computation
- Idle Hardware
- Cooling
- File Systems
- Background Processes

Variables

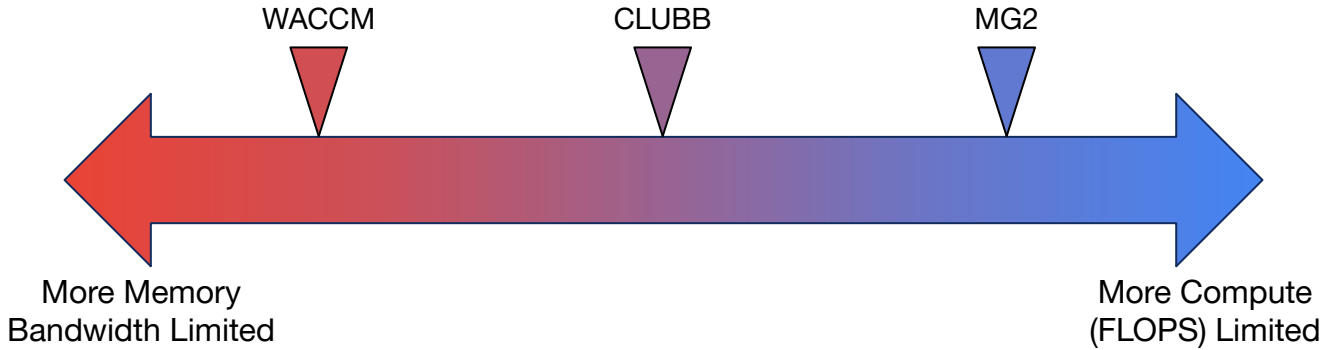
- Hardware
- Temperature
- Clock Speed
- Application Profile

Turbo (up to 2.301 MHz or more)
Rated (less than or equal to 2.3 MHz)
Slow (less than or equal to 1.2 MHz)

Methodology: Kernels

Three CESM2 Kernels and Sleep

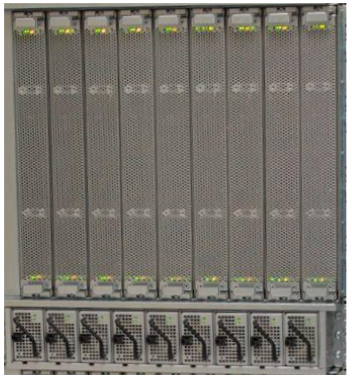
- WACCM_imp_sol_vector: represents a chemistry solver
- CESM2_MG2: represents microphysics calculations
- CESM2_CLUBB: represents a parameterization of clouds and turbulence
- Sleep: node does nothing



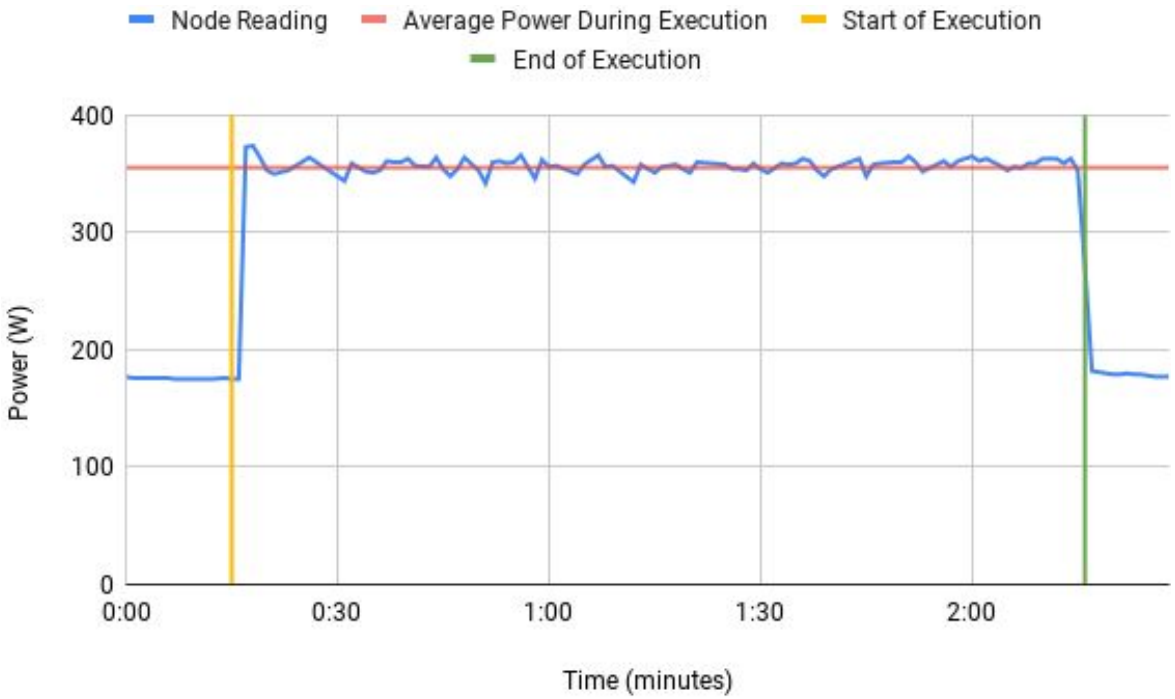
Methodology: Data streams

Data was collected from the node via command line call and from the power supply (PSU) via database

- Neither of these data streams can be calibrated, so confidence only comes from agreement between the two
- Node side data needs to be limited to single node jobs
- On Cheyenne, 9 PSUs feed a group of 36 nodes, so an otherwise empty group is needed to isolate the desired test



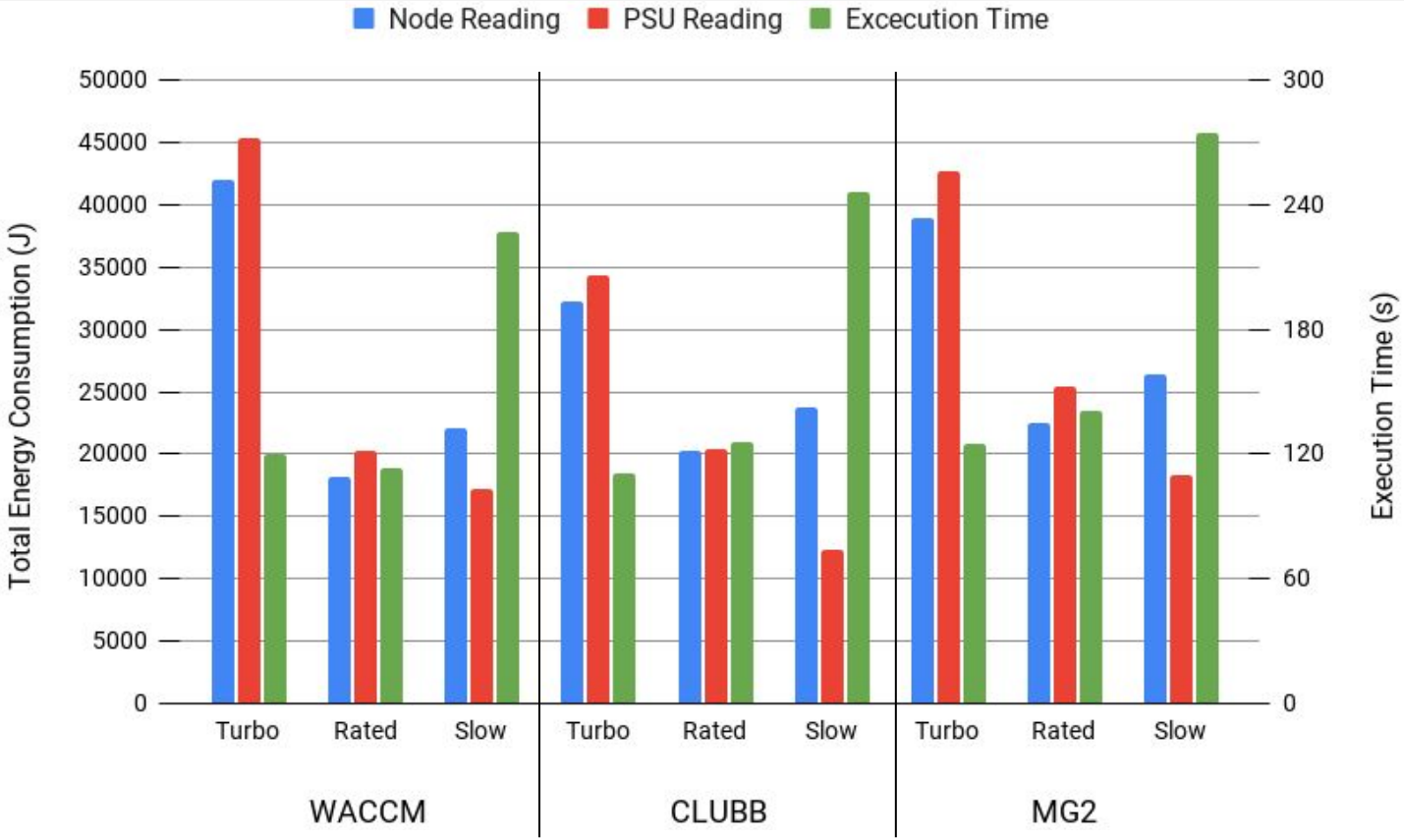
Methodology: Data Collection



In batch script:
1. Node name is recorded
2. Data collection command was run once a second in the background
3. Start timestamp is recorded
4. Kernel execution
5. End timestamp is recorded

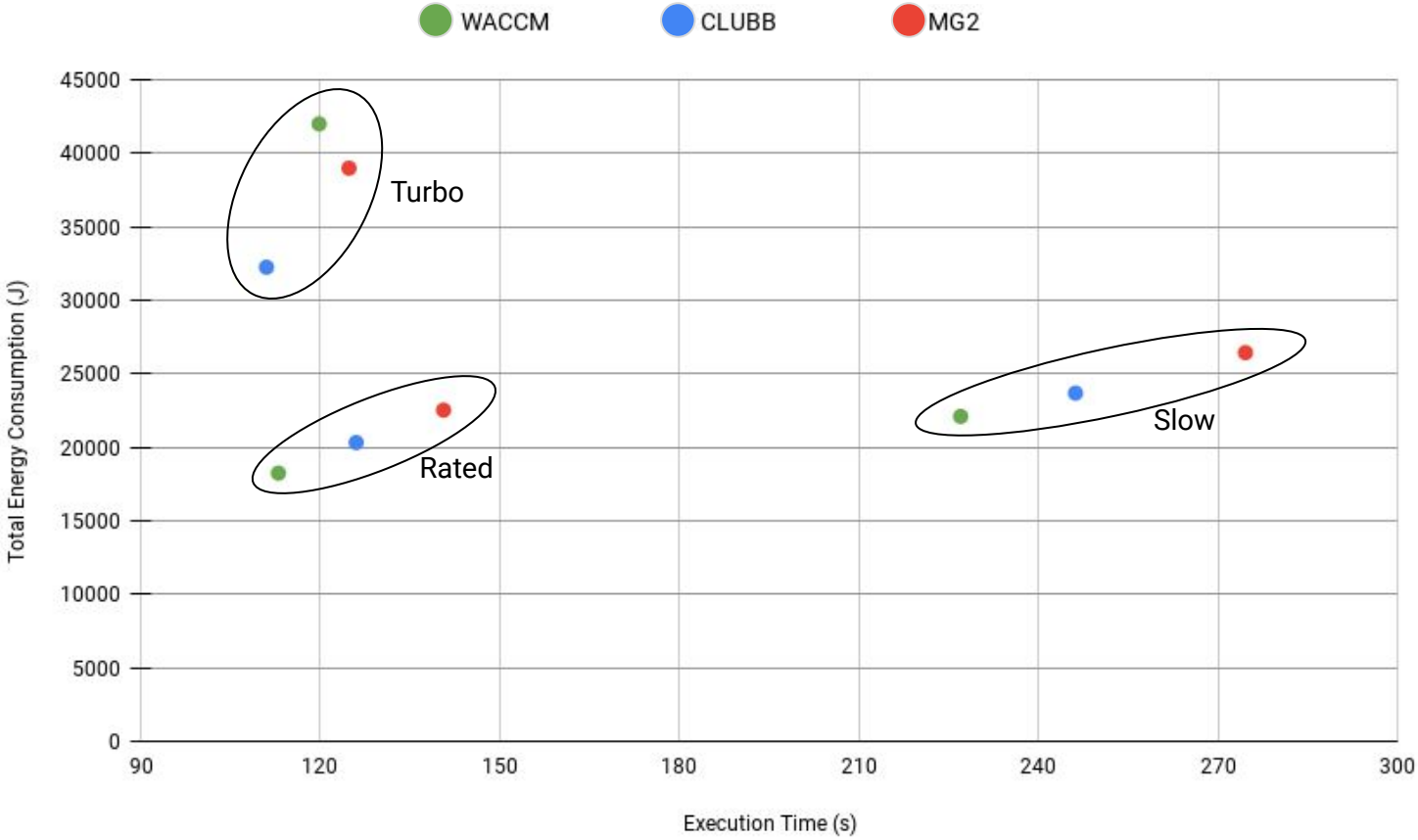
In Python script:
1. Node data is read from job output
2. PSU data is read from database
3. Data is normalized and averaged to get average power, then multiplied by execution time to get total energy
4. Total energy for both data streams and execution time are written to CSV

Average Total Energy Consumption and Average Execution Time by Job Type



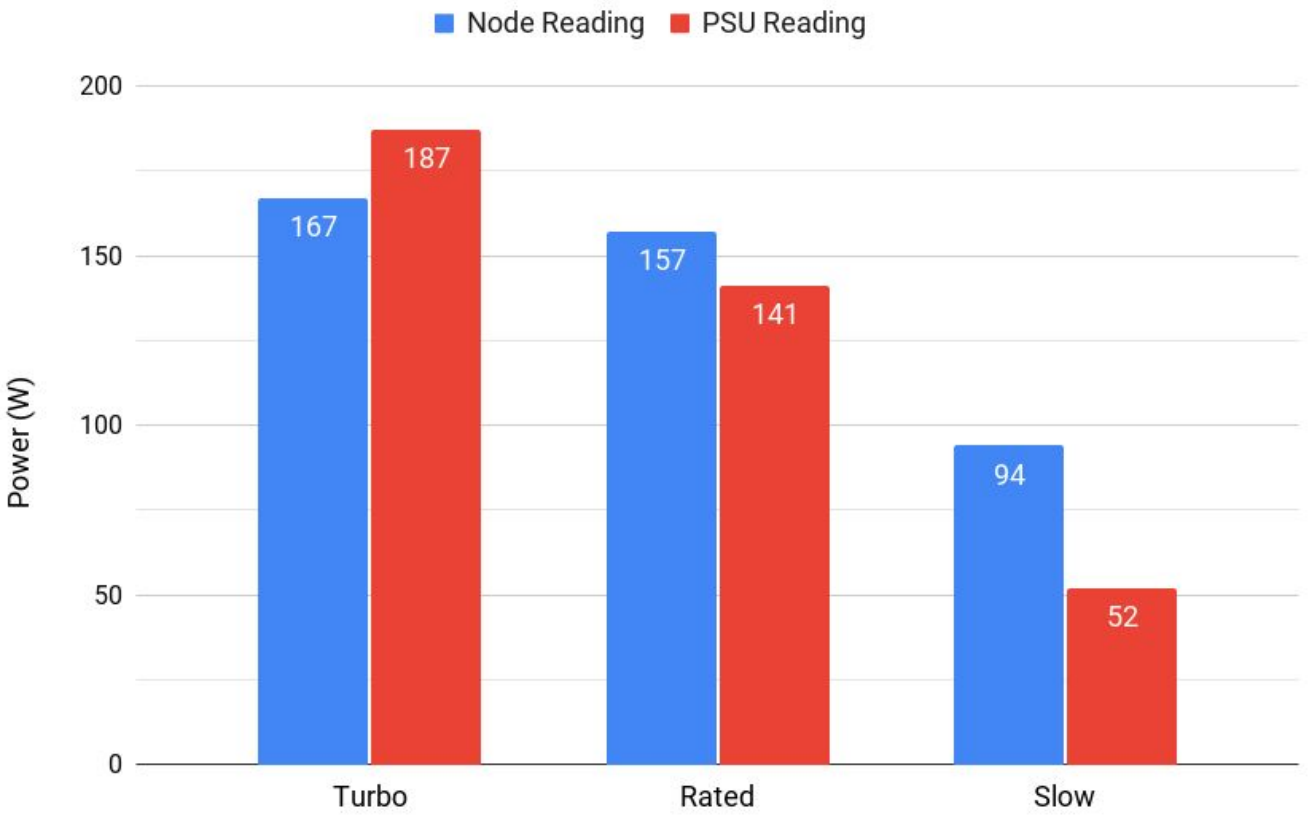
The two data streams have correlation coefficient R=0.8615

Average Total Energy Consumption vs Average Execution Time



Rated gives an average of a 45% decrease in energy consumption with an average 7% increase in execution time compared to Turbo

Average Instantaneous Power Readings on Sleeping Nodes



Idling at slow uses 43.5% less power than idling at turbo

Discussion/Results

This data implies that running jobs at the rated clock speed provides the best energy efficiency with minimal effect on execution time. (Whether or not this is an effective way to reduce supercomputing costs remains to be seen)

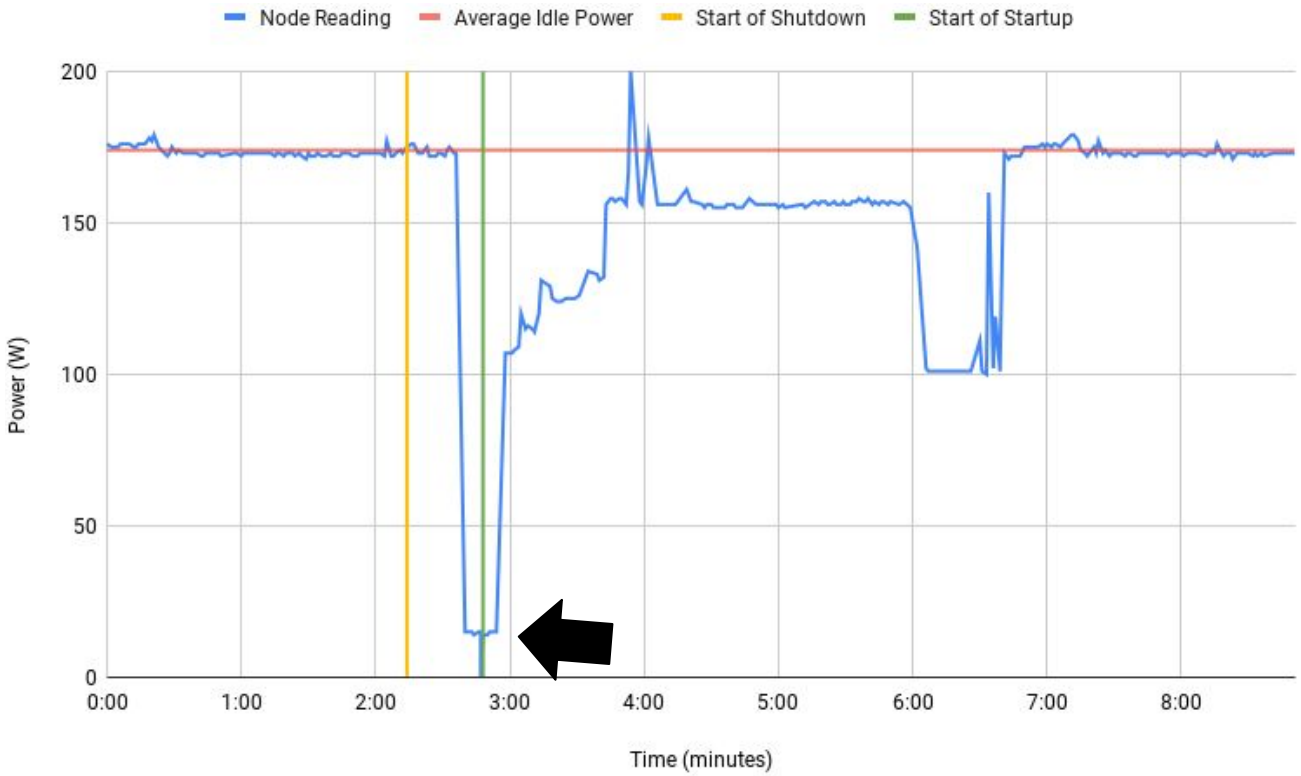
Another major implication of these results is that downclocking idle nodes in between jobs could have a significant impact on the total energy consumption of the system. From this data, we can roughly estimate that idling nodes at slow rather than turbo would carry a savings on the order of tens of thousands of dollars per year.

Continuing Research

- Other hardware
- Actual applications
- Jobs running on more than one node
- Effect of I/O on energy efficiency

- How does shutting down unused nodes scale to larger sections of the system
- Is it still viable to turn the nodes off if they are idling at lower clock speeds

- What's causing this huge drop in power



Acknowledgements

Mentors: Dave Hart, Rory Kelly, Ben Matthews

SIParCS & NESSI: AJ Lauer, Virginia Do, Jerry Cyccone, Max Cordes Galbraith, all the SIParCS and NESSI interns

Thank you to NCAR for hosting this internship and to NSF for providing funding for this research

Special thanks to: Mick Condy, Sidd Ghosh, Richard Valent, Storm Knight, Joseph Mendoza, John Dennis