PNNL-38466

GCAM Regional Tuning

A framework to tune GCAM parameters

September 2025

Pralit Patel
Katherine Calvin
Allen Fawcett
Gokul Iyer
Sash Lamba
Maggie Liu

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from
the Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062
www.osti.gov
ph: (865) 576-8401
fox: (865) 576-5728
email: reports@osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
or (703) 605-6000
email: info@ntis.gov
Online ordering: http://www.ntis.gov

PNNL-38466

GCAM Regional Tuning

A framework to tune GCAM parameters

September 2025

Pralit Patel
Katherine Calvin
Allen Fawcett
Gokul Iyer
Sash Lamba
Maggie Liu

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Abstract

The Global Change Analysis Model (GCAM) assumptions typically generate scenarios that are designed to be internally consistent and globally coherent. The gcamdata tool which facilitates the compilation of data sets and user assumptions is not well suited to tailoring to specific country or regional realities, sponsor requirements, or perform harmonization for model intercomparison needs. As described in this report, the GCAM Regional Tuning project develops a computational framework that enables users to adjust GCAM parameters, so model outputs match targeted outcomes at user-defined spatial, temporal, and sectoral resolutions. The framework integrates GCAM, gcamdata, and gcamwrapper with a set of flexible “tuning directives” and an iterative numerical solver. Users can define targets (e.g., technology shares in power generation, BEV uptake, sectoral service demands), select tuners that manipulate relevant GCAM parameters (e.g., share weights, cost adders, elasticities), and export tuned parameters as reusable GCAM XML inputs for future runs. We demonstrate the approach and document usage, diagnostics, and known limitations, and we outline potential future directions.

Summary

GCAM's globally consistent parameter assumptions[1] are often too idealized for harmonization with near-term, country-specific policy realities or sponsor specific datasets. The state of the art approach to developing these tuned parameters are labor intensive and often not readily transferable. Users need durable, easily repeatable tools to tune GCAM outputs to match targeted outcomes.

The approach described in this report provides a flexible framework which is widely adaptable to various versions of GCAM, sectors, and outcomes of interest. This framework asks users to provide external datasets. We then provide a library of tuning directives, which themselves can be easily extended, that identifies the parameters in GCAM which can affect the model towards the desired outcomes. Then finally identify and map GCAM outcomes to the given target dataset so the framework can evaluate how well a certain set of input parameters perform towards the desired goal. It will iteratively solve for parameter values that most closely align to the user-specified targets. Once found, the tuned parameters are exported as a standalone XML inputs which can be used in subsequent GCAM runs and in alternative scenarios.

We will guide readers through the experience of using the new GCAM Regional Tuning framework. This includes identifying the required software (GCAM[2], gcamwrapper[4], and gcamdata[3]) and setting up the workflow (installing R packages, identify a base configuration, and providing the tuning dataset and appropriate tuning directives). We then run through an example. We also highlight some of the limitations including insufficient parameter interactions, potential unintended outcomes, over-fitting, and solver sensitivity. Finally, the report closes with potential future directions such as expanding the library of tuning directives and maintaining a near-term tuned GCAM reference scenario.

Acknowledgments

This research was supported by the [**insert name of Initiative or Investment**], under the Laboratory Directed Research and Development (LDRD) Program at Pacific Northwest National Laboratory (PNNL). PNNL is a multi-program national laboratory operated for the U.S. Department of Energy (DOE) by Battelle Memorial Institute under Contract No. DE-AC05-76RL01830.

Acronyms and Abbreviations

BEV: Battery Electric Vehicle
DOE: U.S. Department of Energy
GCAM: Global Change Analysis Model
GDP: Gross Domestic Product
LDRD: Laboratory Directed Research and Development
PNNL: Pacific Northwest National Laboratory

Contents

1.0 Introduction

GCAM's core scenarios prioritize internal consistency and cross-region comparability, which is valuable for long-term, globally coherent analysis. However, users frequently need to align GCAM outputs with country-specific policies, sponsor projections, or emerging sector trends, especially in the near term. Examples include aligning power sector technology shares, transport technology adoption, sectoral service demands, resource production profiles, agricultural yields, and GDP paths. Historically, adjusting GCAM to match these targets required one-off scripts, manual edits to XML input files, and other ad hoc procedures that were difficult to maintain and transfer. The GCAM Regional Tuning project addresses this by providing computational tools and a repeatable process for tuning GCAM parameters to match desired outcomes, while documenting the process and output parameters for reuse and future scenarios.

2.0 Regional Tuning Framework and Methods

2.1 Concept and Design Principles

The GCAM Regional Tuning framework is designed for flexibility, supporting a wide variety of model regions (existing or new), sectors, and targets with minimal constraints on user-defined directives. It emphasizes reusability by producing tuned parameter XMLs that can be re-applied across scenarios and GCAM versions, and by enabling users to quickly re-run parameter tuning on newer GCAM versions as the model continues to develop. The approach prioritizes transparency through comprehensive documentation of parameter changes, iteration logs, and diagnostics to help users understand and trust results. Finally, it is grounded in practicality, building on GCAM's existing software ecosystem (gcamdata, gcamwrapper) to query, modify, run, and export parameter changes.

2.2 Architecture

Our workflow begins with a specific GCAM[2] instance—compiled code linked with gcamwrapper[4], a gcamdata[3] run that has produced XML inputs (ideally with a cache of intermediate results useful for sectoral or fuel mapping), and a baseline configuration to anchor the tuning. External datasets and targets are then mapped to GCAM definitions (sectors, fuels, technologies, regions, years). Tuning directives ("tuners") are small, focused modules that can be applied individually or in combination for multi-objective tuning; they identify which GCAM parameters to change (e.g., share weights, cost adders, elasticities), apply those changes to a running GCAM instance, query outputs and compute discrepancies relative to user targets, provide an interface to export tuned parameters as XML, and, where appropriate, establish guardrails and error checks. A numerical solver iteratively adjusts parameters through these tuners until model outputs meet targets within tolerance, supporting common methods such as nleqslv and extendable to others like Broyden-type approaches. Finally, the framework exports tuned parameters as XML input files that can be included in subsequent GCAM runs.

3.0 Supported Tuners

The framework currently includes a library of eight tuning directives described below. These tuners were selected to align with typical GCAM user needs. In many cases these can be used directly, and a user merely needs to provide the target data. In other cases, users can extend these to suite their specific needs. The tuning directives are written using the R6 class standard which allows users a formal way to extend functionality for only the methods required. For instance, adjust the queries that retrieve GCAM outputs to be more targeted or use an alternative aggregation scheme.

- Transportation service output tuner (TRN_TUNER): Targets transportation technology service output by adjusting those same technology's share weights.
- Final demand tuner (FD_TUNER): Targets any final service demands by adjusting its income elasticities.
- Resource production tuner (RES_TUNER): Targets resource production by adjusting a non-energy cost adder. Note, this tuner requires an additional GCAM input file (which may be generated by the tuner, however a user will need to update their configuration file) which adds a suitable non-energy object which the tuner can freely change.
- Electricity / Renewable generation tuner (RENEW_TUNER): Targets electricity generation by technology by adjusting a non-energy cost adder. Note, this tuner requires an additional GCAM input file (which may be generated by the tuner, however a user will need to update their configuration file) which adds a suitable non-energy object which the tuner can freely change.
- Electricity generation tuner (ELEC_TUNER): Targets electricity generation by subsector (absolute generation targets) by adjusting the share weights of those same subsectors.
- Electricity generation share tuner (ELEC_SHARE_TUNER): Targets electricity generation by subsector (percent of total generation) by adjusting the share weights of those same subsectors.
- GDP tuner (GDP_TUNER): Adjusts GDP trajectories directly.
- Yield tuner (YIELD_TUNER): Adjusts agricultural and forestry crop yields directly.

4.0 How to Use the Tool

4.1 Prerequisites

To use the gcam-tuner[5] framework, you need a working GCAM installation with a baseline scenario configuration. The gcamwrapper R package[4] must be linked to that corresponding version of GCAM and installed such that using the standard R library function call can load it. The gcamwrapper package will be used to run GCAM and query results from a running instance. In addition, the gcamdata R package[3] must also be installed so that it can be used to build and export tuned XML inputs and potentially to retrieve data and mappings which may be needed to map user sector or fuel definitions to GCAM's. Users will also use the provided run script and example target files for each tuner type, and supply target datasets mapped to GCAM definitions (regions, sectors, technologies, units, and years).

4.2 Quick Start Workflow

1. Download or clone the gcam-tuner package from https://github.com/JGCRI/gcam-tuner
2. Load the package and helpers.
3. Set user-specified constants:
   - FIRST_FUTURE_YEAR: The first future year to start tuning. If using share-weight-based tuners, this must not be the actual first future year in GCAM.
   - YEAR_LIMIT: The last year to include in tuning; targets beyond this year are ignored.
   - QUERIES: A list of GCAM query names to retrieve model results. Ensure all are available in the designated query file.
4. Choose tuners:
   - Activate desired tuners by specifying paths to their target files (leave empty to disable).
   - Target files should follow the provided column structure and units.
5. Run the tuning framework. Under the hood the following steps occur. Note this process could take up to several hours, depending on the type and number of tuning targets. Diagnostics messages are produced as the algorithm proceeds.
   - Initialize and run the base GCAM scenario:
     1. Execute the base run and write results to the XML database with scenario name "Original".
   - Create tuners and run checks:
     1. Initialize each selected tuner and validate target specifications.

2. If no tuners are specified, the script exits after this step.

- Extract results from the “Original” scenario:

    1. Query and save results relevant to the active tuners (for comparison and solver initialization).

- Run the tuners:

    1. The solver adjusts parameters until outputs match targets.

    2. Iteration logs print comparisons at regular intervals. Note: the printed iteration names may not exactly match internal solver iterations.

- Extract results from the tuned scenario:

    1. Query and save tuned results for the active tuners.

    2. Save tuning parameter values by iteration; plots are generated and saved.

- Export tuned parameters:

    1. Use gcamdata to create an XML pipeline and add the tuned parameters.

    2. Convert and save the XML; include it as an input in future GCAM runs to reproduce tuned behavior.

5.0 Inputs and Outputs

5.1 Inputs

Apart from the pre-requisites, primary user inputs include Target files (CSV-like with specified headers and units) and an associated tuning directive. In some cases, users will need to extend tuning directives beyond those available in the Regional Tuning Framework library. In this case a user may need to provide new gcamwrapper style queries to facilitate identifying parameters to change in GCAM and/or GCAM outputs which can be transformed and compared against the Target file data. Optionally additional GCAM inputs may be required for cost-based tuners (resource, renewable).

5.2 Outputs

The primary output of the regional tuning framework is a standalone XML input file (one or more) which contain the solved set of GCAM parameters that most closely match the users target data. However additional outputs of interest could include Iteration logs, plots comparing targets vs. GCAM outputs, and parameter traces by iteration (e.g., share weights, cost adders, elasticities). These outputs can help a user check on the progress of the algorithm and understand why it ultimately chose the solution parameters that were selected.

6.0 Practical Guidance for tuning

When tuning, aim for targets that are “in the right ballpark” (refrain from reducing solution tolerance) as overly tight targets can induce solver instability or non-convergence; begin with moderate targets and expand scope gradually (for example, tune one sector first, then add others). Use guardrails by tracking sector prices, final demands, and supply–demand balances to avoid unintended consequences, and run simple consistency checks after each iteration (such as energy balance and affordability metrics). For error diagnostics, if the solver stalls or reports issues like “Jacobian singular” or “ill-conditioned,” print the solver message after tuning (e.g., print(out$message)) and verify target coverage (years, regions, units) and parameter ranges. Revisit FIRST_FUTURE_YEAR and share-weight assumptions where relevant, and, if needed, reduce dimensionality by tuning fewer targets simultaneously and provide better initial guesses.

7.0 Known Limitations and Challenges

Tuning introduces risks of parameter interactions and unintended consequences because multiple parameters can influence the same outcome (for example, share weights and cost adders in power generation). Thus, changes in one parameter can ripple into others and cause unexpected shifts. Over-fitting is a concern when tuning narrowly to a single scenario or near-term period, potentially degrading behavior in other scenarios or over longer time-horizons; cross-scenario checks are advisable. When targets are strongly coupled across sectors (e.g., power, transport, final demand), the solver may require careful initialization or dampening. Sensitivity to initial guesses and constraints means poor starting values or unrealistic targets can stall the solver or yield non-physical parameter values. Resolution constraints also arise because some GCAM sectoral detail is coarse relative to country-specific data, making proxies necessary when exact alignment is not feasible. Solver issues such as singular or ill-conditioned Jacobians can occur with complex, highly nonlinear targets, in which case approximate derivatives or alternative solver strategies may be needed. Careful data mapping from country datasets to GCAM definitions is essential, as misalignment can produce apparent tuning “errors” that are actually data issues. Finally, certain tuners require additional GCAM input files (e.g., for resource and renewable cost tuners).

8.0 Potential Future Directions

Future work will expand a curated library of tuning directives by maintaining a repository of commonly used tuners and associated target datasets that can be easily re-applied as GCAM evolves, while establishing standardized guardrail metrics—such as sector prices, final demands, and supplydemand balances—for routine vetting of tuned results. We will collect and analyze tuning cases to perform meta-analysis that reveals common parameter ranges, stability, and transferability across regions and scenarios, and extend the solver to support multi-objective tuning (e.g., matching technology shares and emissions trajectories) with explicit trade-off diagnostics. Diagnostics will be enhanced through better reporting of sensitivity, parameter identifiability, and uncertainty, along with automated detection of ill-conditioned problems and suggested remedies. Broader model development will pursue finer sectoral detail and add new technologies or policy levers where proxies are currently needed. We will also explore distributing tuned scenarios as part of GCAM defaults, maintained over time with version-aware tuning directives and datasets, and improve workflow through automated documentation of tuned parameters and results, tighter integration with continuous integration pipelines, and robust version control.

9.0 References

[1] Joint Global Change Research Institute. “GCAM Documentation”. Zenodo, June 3, 2025. https://doi.org/10.5281/zenodo.15581183.

[2] Ben Bond-Lamberty, Pralit Patel, Joshua Lurz, Page Kyle, kvcalvin, Steve Smith, abigailsnyder, et al. “Jgcri/gcam-core: GCAM 8.2”. Zenodo, June 3, 2025. https://doi.org/10.5281/zenodo.15581174.

[3] Joint Global Change Research Institute (JGCRI). August 17, 2022 “gcamdata.” https://jgcri.github.io/gcamdata/.

[4] Joint Global Change Research Institute (JGCRI). n.d. “gcamwrapper.” Accessed September 25, 2025. https://github.com/JGCRI/gcamwrapper/

[5] Joint Global Change Research Institute (JGCRI). n.d. “gcam-tuner” Accessed September 25, 2025. https://github.com/JGCRI/gcam-tuner/

Pacific Northwest
National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354

1-888-375-PNNL (7665)

www.pnnl.gov