Phase 4: Vacuum-to-FRW Consistency and Scale Sanity

A corridor-style test of the Phase 3 global-amplitude mechanism

Origin Axiom Program

January 7, 2026

Abstract

Phase 4 tests whether the canonical global-amplitude mechanism defined in Phase 3 can be connected, in a structurally reasonable way, to FRW-style dynamics and vacuum-energy-like observables. The goal is to probe whether the non-cancellation floor that stabilises the toy vacuum can also support scale-sane behaviour when mapped into simple cosmological modules, without claiming a full cosmological fit or a theory of everything.

1 Introduction

Phase 4 sits between the mechanism-level vacuum work of Phase 3 and any future attempts to build a unified picture of vacuum, matter, and geometry. Its mission is narrow:

- take the canonical Phase 3 global-amplitude mechanism and non-cancellation floor as given;
- define simple, explicit mappings from the floor-enforced amplitude or residue into FRW-like dynamics and vacuum-energy-like observables; and
- study the resulting behaviour of the Origin-Axiom phase parameter θ in the corridor / ledger framework of Phase 0.

The guiding question is not whether we can reproduce the full ΛCDM model or fit precise cosmological parameters, but whether the Phase 3 mechanism can be made compatible with toy FRW modules in a way that is numerically stable, structurally sane, and expressible as a θ-filter in the sense of Phase 0.

Throughout this phase we distinguish carefully between:

- *binding* outputs, which may eventually define a Phase 4 θ-filter; and
- *non-binding* diagnostics and figures, which serve only as intuition and internal checks.

The present rung does not define any concrete mappings or filters. It only provides a minimal paper skeleton so that future rungs can add well-documented mechanisms and experiments without restructuring the front matter.

2 Mapping families: first pass (F1)

Phase 4 takes as input the Phase 3 global-amplitude mechanism: a toy vacuum with an unconstrained observable $A_0(\theta)$, a non-cancellation floor ε, and a floor-enforced amplitude $A(\theta) = \max(A_0(\theta), \varepsilon)$ defined on a grid $\theta \in [0, 2\pi)$. The present paper introduces a first, explicit mapping family, denoted **F1**, from this structure to a toy vacuum-energy-like scalar.

2.1 F1: direct scalar mapping from $A(\theta)$

The F1 family is intentionally simple. For a fixed Phase 3 vacuum configuration and floor ε (taken from the Phase 3 baseline diagnostics), we define a scalar

$$E_{\mathrm{vac}}(\theta) \;=\; \alpha\, A(\theta)^{\beta}, \tag{1}$$

where $\alpha > 0$ and $\beta > 0$ are explicit, configurable parameters. At this rung we adopt a conservative default, $\alpha = 1$ and $\beta = 2$, and focus on structural behaviour rather than numerical normalisation.

Operationally, Phase 4 reuses the Phase 3 baseline configuration `baseline_v1` and the floor ε recorded in `phase3/outputs/tables/mech_baseline_scan_diagnostics.json`. We then evaluate $A(\theta)$ and $E_{\mathrm{vac}}(\theta)$ on a uniform grid of $N_\theta = 2048$ points in $[0, 2\pi)$. The per-grid values and summary diagnostics are written to

`phase4/outputs/tables/phase4_F1_sanity_curve.csv`,

together with metadata describing the mapping parameters and the underlying Phase 3 diagnostics.

At this stage F1 is a *non-binding* mapping family: it does not yet define a θ-corridor or a Phase 4 θ-filter. Instead it serves as a concrete, auditable bridge between the Phase 3 mechanism and simple scalar observables that later rungs can connect to FRW-like toy modules and corridor construction.

3 Diagnostics and toy corridors

Phase 4 currently provides three layers of diagnostics built on the Phase 3 vacuum mechanism and the F1 mapping family. All of them are explicitly non-binding: they are used for internal checks and illustration, not as canonical θ-filters.

3.1 F1 sanity curve

The first layer is a direct sanity check of the F1 mapping. Using the Phase 3 baseline configuration and the quantile-based floor recorded in `phase3/outputs/tables/mech_baseline_scan_diagnostics.jso`
we evaluate the floor-enforced amplitude $A(\theta)$ on a uniform grid of $N_\theta = 2048$ points in $[0, 2\pi)$ and construct a toy vacuum-energy-like scalar

$$E_{\mathrm{vac}}(\theta) = \alpha A(\theta)^{\beta},$$

with $\alpha = 1$ and $\beta = 4$ at this rung. The script `phase4/src/phase4/run_f1_sanity.py` writes the per-grid values and metadata to

`phase4/outputs/tables/phase4_F1_sanity_curve.csv`,

which establishes that $E_{\mathrm{vac}}(\theta)$ is strictly positive, numerically well-behaved, and tied cleanly to the Phase 3 baseline diagnostics.

3.2 F1 shape diagnostics and toy corridor

The second layer probes the *shape* of $E_{\mathrm{vac}}(\theta)$ by defining a toy, non-binding corridor in the space of scalar values. The script `phase4/src/phase4/run_f1_shape_diagnostics.py` constructs a mask based on

$$E_{\mathrm{vac}}(\theta) \leq E_{\mathrm{vac,min}} + k_\sigma \sigma, \quad k_\sigma = 1,$$

where $E_{\mathrm{vac,min}}$ and σ are the global minimum and standard deviation of the F1 curve. This is intentionally weak: it retains a substantial fraction of the θ-grid while discarding the largest excursions.

The script writes a summary diagnostics file `phase4/outputs/tables/phase4_F1_shape_diagnostics.json`

and a per-grid mask

`phase4/outputs/tables/phase4_F1_shape_mask.csv`,

which later rungs and toy modules can reuse. At this stage the mask is explicitly described as a *toy corridor*: it does not define a canonical $\theta_\star$ and is not elevated to a Phase 4 filter.

3.3 FRW-inspired toy diagnostics

The third layer introduces a minimal FRW-inspired toy diagnostic that treats $E_{\text{vac}}(\theta)$ as a source for a *dimensionless* vacuum term in a simple Hubble-like quantity. From the sanity curve we define

$$\tilde{E}_{\text{vac}}(\theta) = \frac{E_{\text{vac}}(\theta)}{\langle E_{\text{vac}} \rangle}$$

and rescale it to a toy $\Omega_\Lambda(\theta)$ with mean ≈ 0.7. Together with fixed, dimensionless matter and radiation parameters $\Omega_m = 0.3$ and $\Omega_r = 0$, we evaluate

$$H^2(a;\theta) = \Omega_r a^{-4} + \Omega_m a^{-3} + \Omega_\Lambda(\theta)$$

on a scale-factor grid $a \in [0.1, 1]$. No physical units are assigned and no claim is made that these numbers match observations.

The script `phase4/src/phase4/run_f1_frw_toy_diagnostics.py` writes:

- a JSON summary `phase4/outputs/tables/phase4_F1_frw_toy_diagnostics.json`, recording global extrema and moments of E_{vac}, $\Omega_\Lambda(\theta)$, the chosen Ω-parameters, and a simple FRW-sanity fraction; and
- a per-grid mask `phase4/outputs/tables/phase4_F1_frw_toy_mask.csv`, containing θ, $\Omega_\Lambda(\theta)$, and a Boolean indicator of whether $H^2(a;\theta)$ stays positive and within a bounded variation factor over the scale-factor grid.

In the current baseline configuration (with $\Omega_m = 0.3$, $\Omega_r = 0$, $\langle \Omega_\Lambda \rangle \approx 0.7$, and a variation bound of order 10 on the ratio of maximal to minimal H^2 across the scale-factor grid), the FRW-sanity mask happens to be empty: the corresponding diagnostics report a very small FRW-sanity fraction (`frac_sane` ≈ 0). This is recorded as a *toy-level negative result* for this particular normalisation and sanity criterion. It is not used to define a Phase 4 filter and does not by itself constrain θ; its role is purely to illustrate how the F1 scalar can feed into FRW-like checks and to demonstrate how Phase 4 handles empty-corridor outcomes in a structured way.

4 Limitations and outlook

Phase 4 is intentionally narrow in scope. Even once the mappings and diagnostics are implemented, the phase will not claim:

- a full derivation of cosmological parameters;
- a proof that the Origin Axiom is realised in nature; or
- a unique mechanism for connecting vacuum structure to FRW dynamics.

Instead, the goal is to provide a clean yes-or-no style test for a specific question:

> Can the Phase 3 global-amplitude mechanism support scale-sane FRW-like behaviour, in at least one simple mapping family, without producing a degenerate or empty θ-corridor?

If the answer is "no" for all tested mapping families, Phase 4 will record this as a structured negative result, signalling that either the Phase 3 mechanism or the mapping strategy needs revision before further unification attempts.

Appendix A: Phase 4 claims table (draft)

Table 1 summarises the intended Phase 4 claims. At this rung all entries are draft and non-binding.

Table 1: Draft Phase 4 claims. Binding status will be updated once the phase is complete and audited.

ID	Binding?	Summary
C4.1	no	Existence of at least one explicit mapping from the Phase 3 global amplitude or residue into an FRW-like or vacuum-energy-like observable with numerically stable behaviour.
C4.2	no	Existence of a non-empty, non-trivial θ-corridor for at least one such mapping.
C4.3	no	Structured negative result if all tested mappings yield empty or pathological corridors.

Appendix B: Reproducibility notes (draft)

At this rung, Phase 4 builds on the Phase 3 toy vacuum mechanism and its non-cancellation floor, introduces the F1 mapping family, and adds internal diagnostics and a toy, non-binding corridor. This appendix summarises how to rebuild the Phase 4 artifacts used in the paper.

Directory structure

Phase 4 lives in the top-level directory `phase4/` with the following relevant subdirectories:

- `phase4/paper/`: LaTeX sources for the Phase 4 paper (including this appendix).
- `phase4/src/phase4/`: Python source for the F1 mapping family and diagnostics.
- `phase4/outputs/`:
  - `outputs/paper/`: built Phase 4 PDF;
  - `outputs/tables/`: CSV/JSON artifacts from F1 diagnostics.
- `phase4/artifacts/`: canonical Phase 4 PDF artifact exported by the gate script.

Upstream Phase 3 prerequisites

Phase 4 assumes that the Phase 3 baseline scan and binding-certificate artifacts have been generated at least once. In practice this can be ensured by running, from the repository root:

```
bash scripts/phase3_gate.sh --level A
```

and, if needed, explicitly running the Phase 3 scripts:

```
python phase3/src/phase3_mech/run_baseline_scan.py
python phase3/src/phase3_mech/run_binding_certificate.py
```

These commands populate:

- `phase3/outputs/tables/mech_baseline_scan.csv`
- `phase3/outputs/tables/mech_baseline_scan_diagnostics.json`
- `phase3/outputs/tables/mech_binding_certificate.csv`
- `phase3/outputs/tables/mech_binding_certificate_diagnostics.json`

The F1 mapping family reads `mech_baseline_scan_diagnostics.json` to reuse the quantile-based floor and grid configuration.

Phase 4 F1 mapping and diagnostics

With the Phase 3 prerequisites in place, the current Phase 4 diagnostics can be rebuilt via:

```
python phase4/src/phase4/run_f1_sanity.py
python phase4/src/phase4/run_f1_shape_diagnostics.py
```

These scripts write:

- `phase4/outputs/tables/phase4_F1_sanity_curve.csv` (per-θ values of $E_{\mathrm{vac}}(\theta)$ and associated metadata);
- `phase4/outputs/tables/phase4_F1_shape_diagnostics.json` (summary statistics and toy corridor definition);
- `phase4/outputs/tables/phase4_F1_shape_mask.csv` (per-θ Boolean mask indicating membership in the toy corridor).

At this rung these diagnostics are explicitly non-binding: they do not define a canonical θ-filter, but they are used in the Phase 4 paper to illustrate the behaviour of the F1 mapping.

Building the Phase 4 paper and artifact

The Phase 4 paper and its canonical PDF artifact can be rebuilt from the repository root via:

```
bash scripts/phase4_gate.sh --level A
```

This gate script runs a Snakemake workflow in `phase4/workflow/Snakefile` that invokes `latexmk` on `phase4/paper/main.tex` and copies the resulting PDF to:

- `phase4/outputs/paper/phase4_paper.pdf`
- `phase4/artifacts/origin-axiom-phase4.pdf`

The gate currently covers the Phase 4 paper only; F1 diagnostics are rebuilt by the explicit Python commands above.

Planned extensions

The design note `phase4/FRW_TOY_DESIGN.md` specifies a minimal FRW-inspired toy module that may be implemented in later rungs. If such a module is added, this appendix will be extended to include:

- the relevant Python scripts and configuration files;
- any additional JSON/CSV outputs; and
- updated gate levels if FRW-like diagnostics become part of the canonical Phase 4 artifact.

Until then, reproducibility for Phase 4 consists of:

1. regenerating the Phase 3 baseline and binding-certificate artifacts;
2. regenerating the Phase 4 F1 diagnostics; and
3. rebuilding the Phase 4 paper and artifact via the Level A gate.

References