

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

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

Dritëro M.

January 15, 2026

Abstract

This Phase 4 paper takes the Phase 3 amplitude output $\theta \mapsto A(\theta)$ as input and develops a structured, cosmology-facing diagnostic pipeline. The aim is not to identify a preferred θ^* or to claim contact with precision cosmology, but to test whether simple mappings from $A(\theta)$ to Friedmann–Robertson–Walker (FRW)-like quantities can be made numerically stable and transparently interrogated.

We focus on a baseline one-parameter family of mappings, $E_{\text{vac}}(\theta) = \alpha A(\theta)^\beta$, with fixed toy parameters $(\alpha, \beta) = (1, 4)$, and construct a five-layer diagnostic stack: (1) an *F1* sanity curve that probes basic properties of $E_{\text{vac}}(\theta)$; (2) *F1* shape diagnostics that define a purely toy “corridor” based on simple amplitude thresholds; (3) FRW-inspired toy checks that build $H^2(a; \theta)$ from $E_{\text{vac}}(\theta)$ and enforce basic consistency conditions (positivity, matter era, late acceleration); (4) a FRW viability scan that records which θ points satisfy these toy criteria and exports a mask for further use; and (5) a corridor-extraction step that identifies contiguous θ -bands where a given mapping passes the toy FRW checks. On top of this, we implement a Λ CDM-like probe and a stub for an external data probe, designed to handle the absence of bundled data as a structured negative result rather than an implicit success.

All of these constructions are explicitly non-binding: the *F1* mapping is not derived from fundamental physics, the resulting corridors are diagnostic rather than canonical, and no claim is made that any particular θ or mapping solves the cosmological constant problem or reproduces the observed Universe. The value of this phase lies in providing a transparent, reproducible interface between Phase 3 toy mechanisms and FRW-like summaries, together with a clear scaffold that future, more physically motivated mappings or data comparisons could use at higher rungs.

1 Introduction

Phase 3 of the origin–axiom programme constructed a concrete lattice mechanism for scanning a dimensionless parameter θ and produced a baseline diagnostic of the resulting amplitude landscape. In particular, it delivered a reproducible pipeline which, given a fixed mechanism configuration, yields a numerically stable *amplitude* $A(\theta)$, a global floor, and a summary JSON file with basic moments and quantiles of the scan. Phase 3 is deliberately agnostic about cosmology: it stops at a well-controlled $\theta \mapsto A(\theta)$ map together with a mechanism-facing diagnostic layer.

Phase 4 takes this Phase 3 output as a starting point and asks a deliberately modest but physically oriented question: *can we systematically stress-test the Phase 3 landscape through simple, Friedmann–Robertson–Walker (FRW) inspired summaries without over-claiming physical predictions or selecting a unique θ_* ?* The guiding philosophy, inherited from Phase 0, is to separate (i) the construction of reproducible mechanism- and mapping-level objects from (ii) any stronger claims about the structure of the real Universe.

At this rung we therefore introduce a single, explicit mapping family

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

denoted “F1”, which turns the Phase 3 amplitude $A(\theta)$ into a non-negative scalar $E_{\text{vac}}(\theta)$ intended to play the role of a toy vacuum-energy density. We adopt a simple baseline choice $\alpha = 1$, $\beta = 4$ and treat the mapping as a tunable, mechanism-facing object rather than as a physical law. On top of F1 we build a stack of FRW-inspired diagnostics: a toy FRW sanity check, a more structured FRW “viability” scan, an extraction of contiguous θ -corridors, and a broad Λ CDM-like probe, together with an explicit hook for future data-level tests.

Crucially, all of these layers are *non-binding*. Phase 4 does *not* attempt to fit cosmological data, does *not* claim a preferred value of θ , and does *not* promote any FRW diagnostic into a hard filter on the origin–axiom mechanism. Instead, the goal is to show, in a fully auditable way, that the Phase 3 landscape can be connected to simple cosmology-facing summaries and that both positive and negative outcomes (including empty or narrow corridors) are recorded and reasoned about explicitly. Subsequent phases are free to refine the mapping family, the FRW summaries, and the data interfaces, but they do so on top of the reproducible infrastructure established here.

In the concrete baseline implementation reported in this paper, Phase 4 *does* instantiate a specific mapping family and a stack of FRW-inspired diagnostics. The F1 mapping takes the Phase 3 scalar as input and, via a simple normalisation, produces a toy $\Omega_\Lambda(\theta)$; the associated scripts implement FRW-facing sanity, viability, corridor, and Λ CDM-like probes as explicit, reproducible computations with per-grid masks and JSON summaries. What this rung does *not* yet provide is a binding Phase 4 θ -filter or any claim of a fit to observational data. All FRW-facing results are deliberately presented as non-binding worked examples and structured sanity checks, rather than as evidence for a preferred θ_* .

2 Phase 4 mappings: F1 baseline vacuum map and FRW lift

Phase 4 takes the Phase 3 baseline mechanism and turns it into a family of vacuum curves and FRW backgrounds that can be probed against toy diagnostics, Stage 2 hosts, and external data. In this section we define the F1 mapping family at a symbolic level and record how it is implemented in the code.

The starting point is the Phase 3 baseline diagnostics JSON `phase3/outputs/tables/mech_baseline_scan`. That file summarizes the baseline mechanism scan on a uniform θ grid of length N and reports, among other quantities, the unconstrained amplitude range and a reference “floor” value ϵ_{floor} that is used to avoid numerical pathologies in regions where the mechanism is effectively flat. Phase 4 does not change that baseline; it only wraps it into a more structured mapping family.

2.1 Symbolic F1 mapping definition

At a symbolic level, the F1 mapping family is a one-parameter deformation of the Phase 3 baseline amplitude table into a vacuum curve $E_{\text{vac}}(\theta)$ on a fixed grid. We denote by $f_{\text{baseline}}(\theta; \alpha, \beta)$ the rescaled baseline amplitude after applying the F1 hyperparameters (α, β) , and by ϵ_{floor} the numerical floor imported from Phase 3. The Phase 4 F1 vacuum curve is then

$$E_{\text{vac}}^{\text{F1}}(\theta; \alpha, \beta) = \max\left\{\epsilon_{\text{floor}}, f_{\text{baseline}}(\theta; \alpha, \beta)\right\}, \quad (2)$$

evaluated on the same N -point grid used in the Phase 3 diagnostics. In the current baseline configuration, the scripts fix $(\alpha, \beta) = (1, 4)$ and treat ϵ_{floor} and the grid bounds $[\theta_{\min}, \theta_{\max}]$ as imported from the Phase 3 JSON diagnostics.

Equation (2) is intentionally symbolic: the paper does not attempt to re-derive the precise interpolation or rescaling choices inside f_{baseline} . Those details are locked in the code and documented by the Phase 3 diagnostics JSON; the role of the paper is to show how the resulting $E_{\text{vac}}^{\text{F1}}(\theta)$ behaves once lifted into FRW backgrounds and probed by the diagnostics in Section 3.

2.2 Repository implementation and sanity curve

In the repository, the F1 mapping is implemented and exercised by the script `phase4/src/phase4/run_f1_sanity.py`. Running this script reads the Phase 3 baseline diagnostics JSON, constructs the F1 vacuum curve according to Equation (2), and writes two outputs under `phase4/outputs/tables/`:

- a CSV file `phase4_F1_sanity_curve.csv` containing per- θ columns for θ , $E_{\text{vac}}^{\text{F1}}(\theta)$, and any additional helper quantities used downstream;
- a JSON diagnostics file `phase4_F1_sanity_curve_diagnostics.json` summarizing basic statistics of the curve (minimum, maximum, mean, and standard deviation), together with the grid length N and the imported Phase 3 floor and unconstrained range.

The key point is that Phase 4 never reconstructs the baseline physics from scratch. Instead, it treats the Phase 3 baseline as a frozen numerical object and uses the F1 mapping family to explore how a single, coherent vacuum curve can support FRW backgrounds and external probes.

2.3 From F1 vacuum map to FRW backgrounds

The F1 vacuum curve $E_{\text{vac}}^{\text{F1}}(\theta)$ is then lifted into FRW backgrounds by a family of scripts under `phase4/src/phase4/` that are described in Section 3. For each grid point θ_i on the F1 curve, those scripts construct an effective Hubble history $H^2(a; \theta_i)$ with fixed (Ω_m, Ω_r) and an effective $\Omega_\Lambda(\theta_i)$ determined by $E_{\text{vac}}^{\text{F1}}$. The resulting FRW backgrounds are then subjected to a sequence of diagnostics: sanity checks on $H^2(a)$, toy “shape corridor” tests, FRW viability and late-acceleration masks, Λ CDM-like windows, and external-host and data-facing probes.

All of those steps depend on the F1 mapping only through the numerical curve and diagnostics written by `run_f1_sanity.py`. This separation is deliberate: it allows the mapping layer in Equation (2) to be swapped or refined in future work without changing the overall FRW and external-host machinery, as long as the same output format and grid structure are respected.

3 Phase 4 diagnostics: vacuum map, corridors, and FRW-facing probes

Given the F1 mapping family defined in Section 2, the Phase 4 diagnostics are designed to answer a sequence of concrete questions: (i) Is the F1 vacuum curve numerically well behaved on the full θ grid and consistent with the Phase 3 baseline diagnostics? (ii) Does the vacuum curve admit a simple “shape corridor” that can be used as a toy, non-binding reference band in θ ? (iii) When the curve is lifted into FRW backgrounds, for what fraction of the grid do we obtain sensible, late-accelerating cosmologies? (iv) Within that FRW-viable set, is there a window of θ where the background looks Λ CDM-like in a coarse sense? (v) How does that window overlap Stage 2 external hosts and external data-facing probes? This section records how those questions are implemented in the repository and which tables and masks connect the mapping layer to the rest of the program.

3.1 Baseline diagnostics and F1 sanity sweep

The starting point is the Phase 3 baseline diagnostics JSON `phase3/outputs/tables/mech_baseline_scan_diagnostics.json`, which summarizes the unconstrained amplitude distribution and the numerical floor ϵ_{floor} used in Equation (2). The Phase 4 script `phase4/src/phase4/run_f1_sanity.py` reads that JSON, constructs the F1 vacuum curve on the Phase 3 grid, and writes: (i) the per- θ CSV `phase4_F1_sanity_curve.csv` and (ii) the JSON diagnostics `phase4_F1_sanity_curve_diagnostics.json`. The JSON records, among other quantities, the minimum, maximum, mean, and standard deviation of $E_{\text{vac}}^{\text{F1}}(\theta)$, the

grid length N , and the imported Phase 3 floor and unconstrained range. These numbers are used throughout the Phase 4 text whenever we summarize the raw F1 vacuum behaviour.

3.2 Toy shape corridor on the F1 vacuum curve

As a first, deliberately non-binding diagnostic, Phase 4 defines a toy “shape corridor” on the F1 vacuum curve. The goal is to mark a simple band in θ where the vacuum energy sits close to its global minimum, without claiming any direct physical interpretation. The script `phase4/src/phase4/run_f1_shape_diagnostics.py` consumes `phase4_F1_sanity_curve.csv`, computes the global minimum $E_{\text{vac},\min}$ and standard deviation σ of the curve, and defines a toy corridor by $E_{\text{vac}}^{\text{F1}}(\theta) \leq E_{\text{vac},\min} + k_\sigma \sigma$, with a fixed k_σ specified in the script (currently $k_\sigma = 1$). The script writes: (i) `phase4_F1_shape_diagnostics.json`, summarizing the global range of $E_{\text{vac}}^{\text{F1}}$ and the fraction of grid points that satisfy the toy corridor condition; and (ii) `phase4_F1_shape_mask.csv`, a per- θ CSV that marks which grid points lie inside the corridor. The paper only uses this corridor as a visual and conceptual aid when discussing the shape of the F1 vacuum; it is not used as a hard constraint in any FRW or external-host diagnostic.

3.3 Toy FRW backgrounds and viability mask

The next diagnostic lifts the F1 vacuum curve into FRW backgrounds by treating $E_{\text{vac}}^{\text{F1}}(\theta)$ as the source of an effective $\Omega_\Lambda(\theta)$ on a fixed matter–radiation background. The script `phase4/src/phase4/run_f1_frw` reads `phase4_F1_sanity_curve.csv`, fixes (Ω_m, Ω_r) and H_0 to a simple flat- Λ CDM baseline inspired by Planck Collaboration 2018 constraints [1], and constructs $H^2(a; \theta)$ on a grid in the scale factor a . It then applies a set of purely internal sanity checks: the presence of a radiation/matter era, a late-accelerating phase, and the absence of obvious numerical pathologies in $H^2(a)$. The outputs are: (i) `phase4_F1_frw_toy_diagnostics.json`, which reports the fraction of θ values that pass the FRW sanity checks and the corresponding age range for the toy Universe; and (ii) `phase4_F1_frw_toy_mask.csv`, a per- θ CSV that marks FRW-viable grid points. From this point on, all FRW-facing diagnostics in Phase 4 are restricted to the FRW-viable subset of the grid.

3.4 FRW corridors and principal band in θ

Given the FRW-viable mask, the script `phase4/src/phase4/run_f1_frw_corridors.py` identifies connected FRW-viable corridors in θ and highlights a principal band that will be used as the main Phase 4 FRW window. It reads the per- θ FRW viability information, groups consecutive viable grid points into intervals, and records: (i) `phase4_F1_frw_corridors.json`, which summarizes the fraction of the grid that is FRW-viable, the number of corridors, and the θ span of the principal band; and (ii) `phase4_F1_frw_corridors.csv`, which lists the individual corridor intervals and their basic properties. The principal corridor is the object that is later compared to Stage 2 host kernels and external data-facing windows.

3.5 Toy FRW viability and Λ CDM-like probe

To connect the F1 FRW backgrounds to more familiar cosmologies, Phase 4 introduces a coarse Λ CDM-like probe. The script `phase4/src/phase4/run_f1_frw_lcdm_probe.py` takes the FRW-viable mask as input, fixes target values for $(\Omega_m, \Omega_\Lambda, H_0)$ and a broad target age window, and asks: for which FRW-viable θ does the background age and $\Omega_\Lambda(\theta)$ fall within those tolerances? The corresponding outputs are: (i) `phase4_F1_frw_lcdm_probe.json`, recording the fraction of FRW-viable grid points that are deemed Λ CDM-like, together with the θ and age ranges of that set; and (ii) `phase4_F1_frw_lcdm_probe_mask.csv`, a per- θ CSV marking which FRW-viable points also fall into the Λ CDM-like window. This probe is intentionally coarse: it is a structured

Figure 1: Phase 4 F1 FRW shape-probe summary (placeholder).

diagnostic, not a fit to any particular dataset. Its role is to identify a window in θ where the toy FRW background is compatible with a broad, textbook Λ CDM picture.

3.6 External data-facing probe

Finally, Phase 4 provides a hook for direct comparison with an external FRW distance dataset. The script `phase4/src/phase4/run_f1_frw_data_probe.py` is wired to read a binned FRW distance table from `phase4/data/external/frw_distance_binned.csv`, compute a simple χ^2 misfit for each FRW-viable θ , and mark grid points whose χ^2 per degree of freedom lies below a configurable threshold. In the current baseline configuration, the data file is a stub and no θ passes the data-quality mask, but the numerical plumbing and output formats are in place. The outputs are: (i) `phase4_F1_frw_data_probe.json`, which reports whether any usable data were found, the number of FRW-viable grid points tested, and the fraction that pass the χ^2 threshold (zero in the present stub); and (ii) `phase4_F1_frw_data_probe_mask.csv`, a per- θ CSV marking FRW-viable points that are also data-consistent under the chosen threshold. Once a real external dataset is wired in, this probe becomes the natural entry point for Phase 4 to talk directly to observations.

3.7 Shape-probe summary figure (placeholder)

For readability, the paper includes a single shape-probe figure that is meant to summarize the overlap of three key masks: FRW viability, the toy shape corridor, and the Λ CDM-like window. The underlying plot is generated by a Python helper under `phase4/src/phase4/` that consumes the masks described above and writes per- θ summary tables; the figure itself is currently represented in the source as a semantic placeholder: The label `fig:phase4_F1_frw_shape_probe` is used elsewhere in the Phase 4 text to refer to this summary. Replacing the placeholder with the actual plot only requires regenerating the figure and updating the local `\includegraphics` call, without touching any of the numerical diagnostics described above.

4 Limitations and scope of Phase 4

Phase 4 is intentionally modest in scope. Its purpose is to demonstrate how a simple mapping family and a small stack of diagnostics can be wired, in a fully reproducible way, on top of the Phase 3 vacuum mechanism. It is *not* a proposal for a definitive θ -corridor, a preferred θ_* , or a calibrated cosmological model. In this section we make explicit what the current Phase 4 construction does and does not claim.

4.1 Structural vs. physical content

The F1 mapping family, the toy corridor, and the FRW-inspired module are all *structural* devices. They are designed to test whether the Phase 3 amplitudes can be converted into simple scalar observables that admit the kind of corridor and probe machinery laid out in Sec. 2. The construction deliberately holds fixed a number of choices that, in a full physical proposal, would need to be re-opened and justified on their own terms.

In particular, Phase 4:

- fixes the Phase 3 baseline vacuum (including the grid, ε -floor, and diagnostic summary) and does not explore alternative vacua or hyperparameters;

- fixes the F1 mapping family and its parameters (here $\alpha = 1$, $\beta = 4$) rather than treating them as fit parameters or scanning over a broader family;
- defines a toy shape corridor using a simple $E_{\text{vac}}(\theta)$ -based threshold, chosen for readability and reproducibility, not for optimality;
- translates the vacuum curve into FRW-like quantities with a highly simplified background, intended to expose failure modes rather than to compete with standard cosmological analyses.

None of these structural choices are claimed to be unique or optimal. They are scaffolding for Phase 4; any future attempt to argue for a specific θ_* or for phenomenological consequences would need to revisit and justify them.

4.2 FRW-facing limitations

The FRW-facing layers of Phase 4 (toy sanity checks, the viability scan, corridor construction, and the Λ CDM-like probe) are deliberately simple. They fix Ω_m , Ω_r , and H_0 by hand, work with a single baseline mapping family, and use broad box-shaped windows in age and Ω_Λ rather than a likelihood-based treatment. Radiation, baryons, curvature, and observational systematics are all treated in an approximate or implicit way, if at all.

As such, the FRW module is best read as a structured plausibility check on the F1 mapping and on the Phase 3 mechanism, not as a competitive cosmological fit. In particular:

- the Λ CDM-like probe is defined by generous age and Ω_Λ windows rather than by a full parameter inference against data;
- the shape probe and corridor overlap summaries are qualitative diagnostics of how the vacuum curve behaves when pushed through a simple FRW calculator, not precision constraints;
- the FRW data probe is wired into the pipeline but, in the baseline repository configuration, operates on a stub external dataset and records that no real data have been applied.

The numerical fractions and bands reported in the Phase 4 diagnostics and paper should therefore be read as internally consistent checks *within this specific scaffold*, not as evidence for or against any particular cosmological model.

4.3 External host and flatness limitations

The Stage 2 external host layer and its flatness-based kernel provide an additional structural bridge between the Phase 4 FRW module and an external source of FRW background grids. Even here, however, the construction remains deliberately conservative.

The external host kernel used in Phase 4:

- is defined by a near-flatness criterion on Ω_{tot} together with broad age windows, rather than by a full joint likelihood over background parameters;
- is restricted to a small 12-point kernel that matches the Phase 4 FRW viable corridor and the toy shape corridor; and
- is summarised by simple bands in θ , Ω_Λ , age, and a single mechanism scalar mech_baseline_A0, rather than by a higher-dimensional posterior.

The external host kernel is thus a consistency and alignment device: it checks that a simple external FRW grid and the internal Phase 4 construction agree on a small region of parameter space. It is not a claim that the kernel defines an observationally selected θ -interval, nor that it captures all relevant uncertainties in the external constructions it mirrors.

4.4 Data-facing limitations and future rungs

Finally, Phase 4 does not perform any real data assimilation. The FRW data-probe rung is implemented as a reproducible stub: the code expects a binned distance dataset in a standard format but, in the baseline repository, operates on a placeholder file and records that no external data are present. This is intentional. It keeps the artifact self-contained, avoids bundling proprietary or versioned datasets, and prevents accidental over-interpretation of the toy FRW outputs as data-driven constraints.

Any future phase that seeks to confront data would need to:

- specify concrete datasets and preprocessing steps;
- define likelihoods and nuisance-parameter treatments for those datasets;
- revisit the mapping-family assumptions and FRW backgrounds used here; and
- articulate clear, limited claims (and non-claims) about what a given rung does or does not establish.

Those tasks lie outside the scope of the present Phase 4 artifact. Here, the goal is narrower: to show that the Phase 3 vacuum mechanism can be wired, in a disciplined way, into simple FRW-inspired probes and into a small external host bridge, while keeping all assumptions and limitations explicit.

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.

A Reproducibility notes for Phase 4

This appendix records the concrete steps required to reproduce the Phase 4 diagnostics and the main paper artefact from a fresh clone of the repository. The emphasis is on (i) identifying the precise scripts and tables that underpin the claims in the main text and (ii) clarifying the logical flow from Phase 3 outputs to Phase 4 FRW-facing summaries.

B.1. Prerequisites

Phase 4 is designed to sit on top of the Phase 3 mechanism. The following assumptions are therefore required before running the Phase 4 scripts:

- The repository is checked out at a Phase 4-compatible commit and a Python environment satisfying the documented dependencies (NumPy, SciPy, Matplotlib, etc.) is available.
- The Phase 3 baseline scan has been executed, producing the amplitude table and diagnostics JSON `phase3/outputs/tables/mech_baseline_scan_diagnostics.json` as in the Phase 3 paper.
- The `phase3_gate` script (or its Phase 3 equivalent) has been run to ensure that the baseline mechanism artefacts are present and consistent.

In a typical workflow one would run

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

before entering the Phase 4 stage, so that all Phase 3 tables are present and up to date.

B.2. Phase 4 mapping and diagnostics scripts

Phase 4 introduces a single mapping family (F1) and layered diagnostics built on top of it. The core scripts are:

(1) F1 scalar construction and basic diagnostics.

- **Script:** `phase4/src/phase4/run_f1_sanity.py`
- **Inputs:** Phase 3 diagnostics JSON `phase3/outputs/tables/mech_baseline_scan_diagnostics.json` (for the amplitude range and quantile-based floor) and the corresponding amplitude table.
- **Outputs:**
 - `phase4/outputs/tables/phase4_F1_sanity_curve.csv`, containing θ and $E_{\text{vac}}(\theta)$ for the F1 mapping with $\alpha = 1$, $\beta = 4$;
 - `phase4/outputs/tables/phase4_F1_sanity_curve_diagnostics.json`, recording global extrema, basic moments, and configuration metadata.

(2) F1 shape diagnostics and toy corridor.

- **Script:** `phase4/src/phase4/run_f1_shape_diagnostics.py`
- **Inputs:** `phase4_F1_sanity_curve.csv`.
- **Outputs:**
 - `phase4/outputs/tables/phase4_F1_shape_diagnostics.json`, a JSON summary of the F1 scalar, including a toy one- σ -style corridor definition;
 - `phase4/outputs/tables/phase4_F1_shape_mask.csv`, a per- θ CSV with Boolean `in_toy_corridor` flags.

(3) FRW-inspired toy diagnostics.

- **Script:** `phase4/src/phase4/run_f1_frw_toy_diagnostics.py`
- **Inputs:** `phase4_F1_sanity_curve.csv` and the chosen toy parameters Ω_m , Ω_r , a target mean $\langle \Omega_\Lambda \rangle$, and a scale-factor window $[a_{\min}, a_{\max}]$.
- **Outputs:**
 - `phase4/outputs/tables/phase4_F1_frw_toy_diagnostics.json`, which records global extrema and moments of $E_{\text{vac}}(\theta)$ and the toy $\Omega_\Lambda(\theta)$, along with a simple FRW-sanity fraction;
 - `phase4/outputs/tables/phase4_F1_frw_toy_mask.csv`, a per- θ mask with $\Omega_\Lambda(\theta)$ and a Boolean FRW-sanity indicator. In the baseline configuration the FRW toy is treated as a non-binding diagnostic only.

(4) FRW viability scan.

- **Script:** `phase4/src/phase4/run_f1_frw_viability.py`
- **Inputs:** `phase4_F1_sanity_curve.csv` and a choice of FRW parameters (Ω_m , Ω_r , H_0) as well as an age window $[t_{\min}, t_{\max}]$.
- **Outputs:**
 - `phase4/outputs/tables/phase4_F1_frw_viability_diagnostics.json`, recording basic statistics of the FRW histories across the θ -grid, the fraction of points with a matter-dominated era, late-time acceleration, smooth $H^2(a; \theta)$, and an age within the specified window;
 - `phase4/outputs/tables/phase4_F1_frw_viability_mask.csv`, a per- θ mask with columns `theta`, `E_vac`, `Omega_Lambda(theta)`, the inferred age in Gyr, Boolean flags (`has_matter_era`, `has_late_accel`, `smooth_H2`), and a combined `frw_viable` flag.

(5) Corridor extraction on the FRW-viable subset.

- **Script:** `phase4/src/phase4/run_f1_frw_corridors.py`
- **Inputs:** `phase4_F1_frw_viability_mask.csv`.
- **Outputs:**
 - `phase4/outputs/tables/phase4_F1_frw_corridors.json`, a JSON summary of contiguous “corridors” of FRW-viable θ -values, including a principal corridor with its θ -extent and basic statistics of $\Omega_\Lambda(\theta)$;
 - `phase4/outputs/tables/phase4_F1_frw_corridors.csv`, a per-corridor CSV listing the θ -ranges and sizes.

(6) Λ CDM-like probe.

- **Script:** `phase4/src/phase4/run_f1_frw_lcdm_probe.py`
- **Inputs:** `phase4_F1_frw_viability_mask.csv` and a broad target window for Ω_Λ (e.g. 0.7 ± 0.1) and the age of the Universe (e.g. 13.8 ± 1.0 Gyr), with the same FRW parameters as the viability scan.
- **Outputs:**

- `phase4/outputs/tables/phase4_F1_frw_lcdm_probe.json`, which records the fraction of FRW-viable grid points that fall into this broad Λ CDM-like window, together with the θ -extent and the ranges of $\Omega_\Lambda(\theta)$ and age across the selected subset;
- `phase4/outputs/tables/phase4_F1_frw_lcdm_probe_mask.csv`, a per- θ mask augmenting the FRW-viability columns with a Boolean `lcdm_like` flag.

In the baseline configuration this probe is explicitly non-binding: it is used to demonstrate the existence of θ -regions with broadly Λ CDM-like FRW histories, not to select a unique θ_* or to fit observational data.

(7) Joint shape/FRW probe.

- **Script:** `phase4/src/phase4/run_f1_frw_shape_probe.py`
- **Inputs:** `phase4_F1_shape_mask.csv`, `phase4_F1_frw_viability_mask.csv`, and `phase4_F1_frw_lcdm_pr`
- **Outputs:**
 - `phase4/outputs/tables/phase4_F1_frw_shape_probe.json`, which reports the fractions and θ -ranges associated with the toy shape corridor, the FRW-viable subset, the Λ CDM-like subset, and their intersections;
 - `phase4/outputs/tables/phase4_F1_frw_shape_probe_mask.csv`, a joined mask that carries the shape, FRW-viability, and Λ CDM-like flags on a common θ -grid.

Figure ?? in the main text is derived from these joined tables.

(8) Data-facing probe (stub).

- **Script:** `phase4/src/phase4/run_f1_frw_data_probe.py`
- **Inputs:** `phase4_F1_frw_viability_mask.csv` and an optional external binned-distance data file `phase4/data/external/frw_distance_binned.csv`.
- **Outputs:**
 - `phase4/outputs/tables/phase4_F1_frw_data_probe.json`, a diagnostics file recording whether external data were found, the number of data points, and—if present—summary χ^2 statistics;
 - `phase4/outputs/tables/phase4_F1_frw_data_probe_mask.csv`, a per- θ mask with a Boolean `data_ok` flag.

In the baseline repository checkout no external FRW distance data are bundled, so the script records `data_available = false` and sets `data_ok = 0` for all grid points. This is an explicit, structured negative result rather than an implicit limitation: it demonstrates how data-level tests will be integrated in later work without making any present data-driven claims.

B.3. Recommended execution flow

Putting the pieces together, a minimal end-to-end reproduction of the Phase 4 artefact proceeds as follows:

1. Run the Phase 3 gate at an appropriate level, e.g.

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

to ensure the baseline mechanism scan and diagnostics are present.

2. Execute the Phase 4 scripts in the logical order described above, from F1 sanity (`run_f1_sanity.py`) through shape diagnostics, FRW toy, FRW viability, FRW corridors, the Λ CDM-like probe, the joint shape/FRW probe, and finally the optional data-facing stub.
3. Regenerate the Phase 4 paper and canonical artefact via

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

which runs `latexmk` on `phase4/paper/main.tex` and copies the resulting `main.pdf` to `phase4/outputs/paper/phase4_paper.pdf` and `phase4/artifacts/origin-axiom-phase4.pdf`.

The claims made in the main text and in the Phase 4 claims table are deliberately modest: they refer only to the existence and behaviour of these diagnostics and corridors *within this pipeline*. No attempt is made to extract precise cosmological parameter values or to promote any diagnostic into a binding filter on the origin–axiom mechanism.

References

- [1] Planck Collaboration. Planck 2018 results. vi. cosmological parameters. *Astronomy and Astrophysics*, 641:A6, 2020.