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 $\theta^\star$ 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, $E_{\rm vac}(\theta) = \alpha\, A(\theta)^\beta$, 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 $E_{\rm vac}(\theta)$; (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 $E_{\rm vac}(\theta)$ 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 ΛCDM-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 $\theta_\star$?* 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

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

denoted "F1", which turns the Phase 3 amplitude $A(\theta)$ into a non-negative scalar $E_{\rm vac}(\theta)$ 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 ΛCDM-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 ΛCDM-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 $\theta_\star$.

2 Mapping family and scalar $E_{\rm vac}(\theta)$

Phase 3 furnishes a fixed, mechanism-level map $\theta \mapsto A(\theta)$ together with a global floor and a summary diagnostics file. Concretely, a baseline configuration of the Phase 3 mechanism produces a table of amplitudes $A(\theta_j)$ on a uniform θ-grid and writes, among other outputs, a JSON file

$$\texttt{phase3/outputs/tables/mech\_baseline\_scan\_diagnostics.json}, \tag{2}$$

which records the global minimum and maximum, basic moments, and quantiles of the scan. Phase 3 itself stops at this level: it does not interpret $A(\theta)$ as a cosmological quantity.

Phase 4 introduces a simple, explicit mapping family that turns the Phase 3 amplitude into a non-negative scalar intended to play the role of a toy vacuum-energy density:

$$E_{\rm vac}(\theta) \;=\; \alpha\, A(\theta)^{\beta}, \qquad \alpha > 0,\ \beta > 0. \tag{3}$$

We refer to this family as "F1". The two parameters α and β are treated as tunable, mechanism-facing hyperparameters: they shape the relative weighting of amplitude variations across the θ-grid but do not correspond to any claimed physical constants.

In this Phase 4 baseline rung we adopt the concrete choice

$$\alpha = 1, \qquad \beta = 4, \tag{4}$$

motivated by three simple considerations:

1. $A(\theta)$ is already normalised at the mechanism level, so it is natural to absorb any overall scale into a separate FRW toy-model normalisation rather than into α;

2. an even power β guarantees that $E_{\rm vac}$ remains non-negative wherever $A(\theta)$ is defined;

3. a quartic choice $\beta = 4$ accentuates contrasts between low- and high-amplitude regions without being numerically extreme on the Phase 3 baseline scan.

These points are *pragmatic* rather than physical: they select a well-behaved default for the present rung while keeping the door open to future exploration of alternative exponents and normalisations.

Given a fixed Phase 3 amplitude table, the Phase 4 script `phase4/src/phase4/run_f1_sanity.py` constructs the scalar curve $E_{\mathrm{vac}}(\theta)$ according to Eq. (5) with $\alpha = 1$ and $\beta = 4$, reusing the global minimum and quantile-based floor reported in `mech_baseline_scan_diagnostics.json`. It writes:

- a per-θ CSV `phase4/outputs/tables/phase4_F1_sanity_curve.csv` containing θ and $E_{\mathrm{vac}}(\theta)$; and
- a JSON summary `phase4/outputs/tables/phase4_F1_sanity_curve_diagnostics.json` with global extrema, basic moments, and configuration metadata.

This establishes a reproducible, one-dimensional scalar landscape on top of the Phase 3 mechanism. All subsequent Phase 4 diagnostics— including the toy corridor, FRW-inspired sanity checks, FRW viability scan, corridor extraction, and ΛCDM-facing probes—are built *on this F1 scalar*, and their outputs always reference `phase4_F1_*` tables rather than the raw Phase 3 amplitudes.

Throughout this rung, F1 is treated as a *diagnostic mapping* rather than a physical law. In particular, we do not claim that $E_{\mathrm{vac}}(\theta)$ is the actual vacuum energy of the real Universe; instead, we use it as a controlled, reproducible proxy for asking whether the Phase 3 landscape can coherently feed into FRW-like summaries without generating obviously pathological behaviour (e.g. negative or wildly oscillatory $H^2(a;\theta)$) across the θ-grid. Future work is free to vary α, β, or even the functional form of the mapping, but such variations will sit on top of the baseline pipeline documented here.

At this stage the F1 family should be read strictly as a structural, toy choice. The exponents and normalisation $(\alpha, \beta) = (1, 4)$ are selected for numerical convenience and to accentuate contrast in the Phase 3 amplitude; they are *not* proposed as a physically motivated relation between the toy amplitude and any real vacuum energy density. Any future, data-facing use of the Phase 4 pipeline would be expected to revisit and possibly replace this mapping.

$$E_{\mathrm{vac}}^{(\mathrm{F1})}(\theta) = E_{\mathrm{vac}}^{\mathrm{baseline}}(\theta; \alpha = 1, \beta = 4) \tag{5}$$

The Phase 4 diagnostics are deliberately layered. The goal is to place the F1 vacuum mapping, the FRW lift, and the external-host bridge on a common θ-grid, and to make every intermediate table and mask *explicitly* reproducible from scripts in the repository. This section summarises what each diagnostic script does, what it reads, what it writes, and how the resulting masks line up.

Throughout, we work with the baseline F1 mapping `F1_baseline_v1` and its associated Phase 3 diagnostics. All tables and JSON files referenced below live under `phase4/outputs/tables/` unless otherwise stated.

(1) F1 “sanity” curve: lifting Phase 3 to Phase 4.

The starting point for the Phase 4 layer is the Phase 3 mechanism baseline scan diagnostics in `phase3/outputs/tables/mech_baseline_scan_diagnostics.json`. These diagnostics encode the one-dimensional θ-grid, the baseline unconstrained mechanism values, and simple summary statistics such as

- the minimum and maximum unconstrained mechanism values,
- quartiles of the unconstrained distribution,
- the fraction of grid points that are at the imposed `epsfloor` bound.

The script `phase4/src/phase4/run_f1_sanity.py` turns this Phase 3 baseline information into a Phase 4 F1 “sanity” curve. It constructs a uniform θ-grid with `n_grid = 2048` points spanning $[0, 2\pi)$, evaluates the F1 vacuum mapping $E_{\rm vac}(\theta)$ on that grid, and writes a CSV table

`phase4_F1_sanity_curve.csv`

with at least the columns

- `theta` (the grid in θ),
- `E_vac` (the mapped vacuum scale),
- a small set of diagnostic fields (e.g. the baseline `epsfloor` threshold carried through from Phase 3).

It also writes a JSON diagnostics file `phase4_F1_sanity_curve_diagnostics.json` that records the key summary statistics of the curve, including

- `theta_min`, `theta_max`,
- `E_vac_min`, `E_vac_max`,
- `E_vac_mean`, `E_vac_std`,
- a copy of the Phase 3 baseline diagnostics used.

In the baseline configuration used here, the F1 sanity curve respects the Phase 3 `epsfloor` threshold and simply transports the Phase 3 amplitude structure into the Phase 4 layer without any additional tuning.

(2) F1 “shape corridor” on the F1 sanity curve.
The next diagnostic is a toy “shape corridor” on the F1 vacuum curve. The script `phase4/src/phase4/run_f` reads the sanity-curve CSV `phase4_F1_sanity_curve.csv`, computes summary statistics of $E_{\rm vac}(\theta)$, and defines a simple corridor as

$$E_{\rm vac}(\theta) \leq E_{\rm vac}^{\rm min} + k_\sigma \, \sigma_{E_{\rm vac}},$$

with $k_\sigma = 1$ in the baseline runs. This is deliberately an *internal* diagnostic: it does not encode any observational constraint, only the shape of the F1 vacuum map itself.

The script writes

- a JSON summary `phase4_F1_shape_diagnostics.json` containing, for example,
  - `E_vac_min`, `E_vac_max`,
  - `E_vac_mean`, `E_vac_std`,
  - `corridor_fraction` (the fraction of grid points in the toy shape corridor),
  - `corridor_theta_min`, `corridor_theta_max`.
- a per-θ mask CSV `phase4_F1_shape_mask.csv` with columns
  - `theta`,
  - `E_vac`,
  - `in_toy_corridor` (Boolean).

This corridor plays a purely structural role: it lets us track how strongly the later FRW and external-host layers overlap with a region where the vacuum map is “low” relative to its global spread, without imposing any physical interpretation on that choice.

(3) FRW lift and viability diagnostics.

We then lift the F1 vacuum curve into a toy FRW background. The script `phase4/src/phase4/run_f1_frw_` reads `phase4_F1_sanity_curve.csv` and interprets $E_{\mathrm{vac}}(\theta)$ as a proxy for $\Omega_\Lambda(\theta)$ via a simple rescaling. In the baseline configuration we fix

- $\Omega_m = 0.3$,
- $\Omega_r = 0$,
- and match the mean of $\Omega_\Lambda(\theta)$ to a target value $\Omega_{\Lambda,\mathrm{target}} = 0.7$.

The script then integrates a flat FRW background for each θ-grid point on a scale-factor grid $[a_{\min}, a_{\max}] = [0.5, 1]$ with `n_a = 128`, and checks a set of FRW “sanity” and viability conditions. These include:

- the existence of a matter-dominated era,
- late-time acceleration,
- a smoothly varying $H^2(a)$ over the sampled interval,
- an age that lies inside a generous window (here $[10, 20]$ Gyr).

The script writes

- a JSON diagnostics file `phase4_F1_frw_toy_diagnostics.json` summarising the global statistics, including
  - `age_Gyr_min`, `age_Gyr_max`,
  - `frac_has_matter_era`,
  - `frac_has_late_accel`,
  - `frac_smooth_H2`,
  - `frac_age_window_ok`,
  - `frac_viable`.
- a per-θ mask CSV `phase4_F1_frw_toy_mask.csv` with columns such as
  - `theta`,
  - `E_vac`,
  - `omega_lambda`,
  - `age_Gyr`,
  - Boolean flags recording matter era, late-time acceleration, FRW-sanity, and age-window membership.

From this toy FRW lift we then construct a more compact FRW-viability mask. The script `phase4/src/phase4/run_f1_frw_viability_mask.py` reads the toy diagnostics and writes

- a JSON summary `phase4_F1_frw_viability_diagnostics.json` (used as the backbone of the Phase 4 FRW viability statements),

- a mask CSV `phase4_F1_frw_viability_mask.csv` combining the per-θ flags into a single `frw_viable` Boolean.

In the baseline configuration used here, roughly half of the θ-grid is FRW-viable under these broad criteria.

(4) FRW corridors in θ.
Given the FRW-viability mask, we next group θ-points into contiguous corridors. The script `phase4/src/phase4/run_f1_frw_corridors.py` takes `phase4_F1_frw_viability_mask.csv` as input and identifies contiguous runs of grid points with `frw_viable = 1`. It writes

- a JSON file `phase4_F1_frw_corridors.json` summarising the number of corridors, the size of each corridor, and in particular the "principal" corridor (the one with the largest number of FRW-viable points),
- a CSV table `phase4_F1_frw_corridors.csv` that records, for each corridor,
  - `theta_min`, `theta_max`,
  - `n_points`,
  - the mean Ω_Λ on that corridor,
  - the index of the corridor.

In the baseline run, the principal FRW corridor covers the same θ-interval that later supports the external-host kernel: roughly $\theta \in [0.43, 3.54]$, with an internal mean $\Omega_\Lambda \simeq 1.3$ under the toy mapping used here.

(5) A broad ΛCDM-like FRW probe.
To relate the toy FRW lift to a more familiar late-time background, we introduce a broad ΛCDM-like probe. The script `phase4/src/phase4/run_f1_frw_lcdm_probe.py` again reads `phase4_F1_frw_viability_mask.csv` and selects FRW-viable grid points that satisfy both

- $|\Omega_\Lambda(\theta) - \Omega_{\Lambda,\text{target}}| \le \Delta\Omega_\Lambda$, with $\Omega_{\Lambda,\text{target}} = 0.7$ and $\Delta\Omega_\Lambda = 0.1$,
- $|t_\text{age}(\theta) - t_\text{target}| \le \Delta t$, with $t_\text{target} = 13.8$ Gyr and $\Delta t = 1.0$ Gyr.

The output consists of

- a JSON summary `phase4_F1_frw_lcdm_probe.json` that records, among other quantities,
  - `n_grid`, `n_frw_viable`,
  - `n_lcdm_like`,
  - `lcdm_like_fraction`,
  - the θ-range and age/Ω_Λ-range of the selected window.
- a mask CSV `phase4_F1_frw_lcdm_probe_mask.csv` which augments the FRW-viability columns with a Boolean `lcdm_like` flag.

In the current baseline configuration, this ΛCDM-like window selects a small but non-zero subset of the FRW-viable θ-grid. It is intentionally broad and remains a toy-level probe: it is not a fit to observational data and does not single out a preferred $\theta_\star$. Instead, it demonstrates that the Phase 3 $\to$ Phase 4 pipeline can admit FRW histories that are, in a coarse sense, compatible with late-time ΛCDM-like cosmology.

(6) Joined F1/FRW shape probe.
Finally, we introduce a joined F1/FRW "shape probe" that ties together

- the F1 toy shape corridor (phase4_F1_shape_mask.csv),
- the FRW-viability mask (phase4_F1_frw_viability_mask.csv),
- the ΛCDM-like FRW window (phase4_F1_frw_lcdm_probe_mask.csv)

on a common θ-grid. The script phase4/src/phase4/run_f1_frw_shape_probe.py reads these inputs and writes

- a per-θ joined mask phase4_F1_frw_shape_probe_mask.csv with columns
  - theta,
  - E_vac,
  - omega_lambda,
  - age_Gyr,
  - in_toy_corridor,
  - frw_viable,
  - lcdm_like,
  - and a combined shape/FRW flag (e.g. shape_and_lcdm).
- a JSON diagnostics summary phase4_F1_frw_shape_probe.json that reports fractions such as
  - frac_frw_viable,
  - frac_in_toy_corridor,
  - frac_lcdm_like,
  - frac_shape_and_viable,
  - frac_shape_and_lcdm,

  together with the corresponding θ-ranges.

To provide a compact visual summary, the script phase4/src/phase4/plot_f1_frw_shape_probe.py builds a Figure showing $\Omega_\Lambda(\theta)$ with overlays for the toy shape corridor, FRW-viable region, and ΛCDM-like window. The resulting plot is stored as

outputs/figures/phase4_F1_frw_shape_probe_omega_lambda_vs_theta.png

and referenced in the text as Figure **??**.

At the present baseline, the joint shape/FRW probe exercises the shape and FRW-viability machinery, but does *not* yet attempt a genuine comparison to observational data. The Phase 4 data-probe stub, implemented in phase4/src/phase4/run_f1_frw_data_probe.py, is wired to look for an external binned FRW distance dataset at

phase4/data/external/frw_distance_binned.csv,

with columns z, mu, and sigma_mu. In the baseline repository configuration this file is an explicit stub; the current Phase 4 layer exercises the data-probe code path but records data_available = false, n_data_points = 0, and n_data_ok = 0 in phase4_F1_frw_data_probe.json. This keeps the FRW/data interface explicit while avoiding any accidental dependence on an unversioned external dataset.

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

3 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 $\theta_\star$, or a calibrated cosmological model. In this section we make explicit what the current Phase 4 construction does and does not claim.

3.1 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. **??**. 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 $\theta_\star$ or for phenomenological consequences would need to revisit and justify them.

3.2 FRW-facing limitations

The FRW-facing layers of Phase 4 (toy sanity checks, the viability scan, corridor construction, and the ΛCDM-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 ΛCDM-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.

3.3 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.

3.4 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 $E_{\mathrm{vac}}(\theta)$ 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 $E_{\mathrm{vac}}(\theta)$ 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) ΛCDM-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 ΛCDM-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 ΛCDM-like FRW histories, not to select a unique $\theta_\star$ 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 ΛCDM-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 ΛCDM-like flags on a common θ-grid.

Figure 1 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 ΛCDM-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