Euclidean ϕ^4 theory in 0+1 dimensions: correlators, spectrum, and PIMC

A theoretical and methodological note for `PIMC-anharmonic-oscillator`

Antonio Mosca

February 8, 2026

Abstract

This document accompanies the repository `PIMC-anharmonic-oscillator`. The target is deliberately narrow: start from the operator definition of the thermal problem, derive the Euclidean lattice measure that is actually sampled in the code, extract the spectral content of the Euclidean two-point function, and isolate the last fully analytic checkpoints (harmonic limit and weak-coupling trend). Beyond that boundary the model is genuinely non-Gaussian and one must sample the Euclidean weight nonperturbatively; the final part of the note explains, in the same order as the theory, how the repository implements Path Integral Monte Carlo (PIMC), the estimators, and the analysis pipeline.

Contents

1 From correlators to the quartic oscillator

A useful way to frame quantum dynamics (and, in QFT, the dynamics of fields) is to treat correlation functions as the primary objects: they are directly connected to linear response, spectral densities, and the extraction of energy scales. In Euclidean language one typically packages them through a source-dependent functional integral:

$$\mathcal{Z}[J] = \int \mathcal{D}\phi \ \exp\left\{ -S_E[\phi] + \int \mathrm{d}^{d+1}x \, J(x)\phi(x) \right\}$$

so that n-point functions follow from differentiating with respect to J and setting $J = 0$. When S_E is quadratic the measure is Gaussian: Wick factorization then collapses all higher correlators to the two-point function. The first nontrivial deformation of that situation is a quartic interaction, the ϕ^4 term, which makes the measure non-Gaussian and produces genuine connected contributions beyond pairwise Wick contractions.

The repository studies the extreme reduction of Euclidean ϕ^4 to 0+1 dimensions, where the field becomes a single coordinate $x(\tau)$ living only in Euclidean time. Equivalently, it is the finite-temperature quantum mechanics of the quartic anharmonic oscillator (units $\hbar = k_B = 1$, so $\beta = 1/T$):

$$\boxed{H = \frac{p^2}{2m} + V(x), \qquad V(x) = \frac{1}{2}m\omega^2x^2 + \lambda x^4, \qquad m > 0,\ \omega \geq 0,\ \lambda \geq 0} \tag{1}$$

This is the simplest interaction where: (i) the harmonic case $\lambda = 0$ is fully analytic and therefore a hard validation target for the entire numerical pipeline; and (ii) turning on λ forces one out of Gaussian territory in a controlled, yet nonperturbative, way.

The thermal objects of interest are the partition function and thermal expectation values:

$$Z(\beta) = \mathrm{Tr}\left(\mathrm{e}^{-\beta H}\right), \qquad \langle \mathcal{O} \rangle_\beta = \frac{1}{Z(\beta)} \, \mathrm{Tr}\left(\mathrm{e}^{-\beta H}\mathcal{O}\right)$$

The central observable throughout the repository is the Euclidean two-point function:

$$G(\tau) = \langle x(\tau)x(0) \rangle_\beta \,, \qquad x(\tau) = \mathrm{e}^{\tau H} x \, \mathrm{e}^{-\tau H}, \qquad 0 \leq \tau \leq \beta$$

together with local moments such as $\langle x \rangle_\beta$ and $\langle x^2 \rangle_\beta$. The reason is simple: in the low-temperature regime $G(\tau)$ contains clean information about the spectral gap $\Delta = E_1 - E_0$, and this gap is the most robust single-number summary of the dynamics across couplings.

2 From the trace to the Euclidean lattice measure

Everything that PIMC samples in this repository follows from rewriting the operator trace $Z(\beta) = \mathrm{Tr}(\mathrm{e}^{-\beta H})$ as a (discretized) Euclidean path integral.

The steps are standard, but I spell them out because they determine the precise lattice action implemented in `src/action.py`.

Start from:

$$Z(\beta) = \mathrm{Tr}\Big(\mathrm{e}^{-\beta(T+V)}\Big), \qquad T = \frac{p^2}{2m}$$

and discretize Euclidean time into N slices:

$$a = \frac{\beta}{N}, \qquad \tau_i = i\,a, \qquad i = 0, 1, \ldots, N-1$$

Using the (first-order) Trotter product formula:

$$\mathrm{e}^{-\beta(T+V)} = \lim_{N\to\infty} \Big(\mathrm{e}^{-a(T+V)}\Big)^N = \lim_{N\to\infty} \big(\mathrm{e}^{-aT}\mathrm{e}^{-aV}\big)^N$$

one defines the finite-N approximant:

$$Z_N = \mathrm{Tr}\big(\mathrm{e}^{-aT}\mathrm{e}^{-aV}\big)^N$$

whose discretization error vanishes as $a \to 0$ at fixed β.

Insert N resolutions of the identity in the position basis:

$$\mathbf{1} = \int_{-\infty}^{+\infty} \mathrm{d}x_i\, |x_i\rangle\langle x_i|$$

and use cyclicity of the trace to enforce periodicity $x_N \equiv x_0$. This gives:

$$Z_N = \int \prod_{i=0}^{N-1} \mathrm{d}x_i \;\prod_{i=0}^{N-1} \langle x_{i+1}|\mathrm{e}^{-aT}|x_i\rangle\; \mathrm{e}^{-aV(x_i)}$$

The only nontrivial ingredient is the kinetic kernel $\langle x_{i+1}|\mathrm{e}^{-aT}|x_i\rangle$. Insert momentum states:

$$\mathbf{1} = \int \frac{\mathrm{d}p}{2\pi}\, |p\rangle\langle p|, \qquad \langle x|p\rangle = \mathrm{e}^{\mathrm{i}px}$$

so that, with $\Delta x_i = x_{i+1} - x_i$:

$$\begin{aligned}\langle x_{i+1}|\mathrm{e}^{-aT}|x_i\rangle &= \int \frac{\mathrm{d}p}{2\pi}\, \exp\Big(-a\,\frac{p^2}{2m}\Big)\; \mathrm{e}^{\mathrm{i}p(x_{i+1}-x_i)} \\ &= \int \frac{\mathrm{d}p}{2\pi}\, \exp\Big[-\frac{a}{2m}p^2 + \mathrm{i}p\,\Delta x_i\Big]\end{aligned}$$

Completing the square:

$$\begin{aligned}-\frac{a}{2m}p^2 + \mathrm{i}p\,\Delta x_i &= -\frac{a}{2m}\left[p^2 - 2\left(\mathrm{i}\frac{m}{a}\Delta x_i\right)p\right] \\ &= -\frac{a}{2m}\left(p - \mathrm{i}\frac{m}{a}\Delta x_i\right)^2 - \frac{m}{2a}(\Delta x_i)^2\end{aligned}$$

the Gaussian integral over the shifted variable yields:

$$\int \frac{\mathrm{d}p}{2\pi}\, \exp\Big[-\frac{a}{2m}\left(p - \mathrm{i}\frac{m}{a}\Delta x\right)^2\Big] = \sqrt{\frac{m}{2\pi a}}$$

hence:

$$\langle x_{i+1}|\mathrm{e}^{-aT}|x_i\rangle = \sqrt{\frac{m}{2\pi a}}\;\exp\Big[-\frac{m}{2a}(x_{i+1}-x_i)^2\Big]$$

Substituting back into Z_N one arrives at:

$$Z_N = \Big(\frac{m}{2\pi a}\Big)^{N/2}\int\prod_{i=0}^{N-1}\mathrm{d}x_i\;\exp\big(-S_E[\boldsymbol{x}]\big), \qquad \boldsymbol{x} = (x_0,\dots,x_{N-1})$$

with the discretized Euclidean action:

$$\boxed{S_E[\boldsymbol{x}] = \sum_{i=0}^{N-1}\Big[\frac{m}{2a}(x_{i+1}-x_i)^2 + a\,V(x_i)\Big]\,, \qquad x_N \equiv x_0} \tag{2}$$

This is exactly the convention implemented by `euclidean_lattice_action` in `src/action.py`. The potential $V(x) = \frac{1}{2}m\omega^2x^2+\lambda x^4$ is implemented in `src/potential.py` as `potential_energy`, with derivative (force) `potential_force`.

For Monte Carlo purposes the overall prefactor $(m/2\pi a)^{N/2}$ is irrelevant: it cancels in normalized expectation values, so the operative weight is simply $\exp(-S_E)$. In the continuum limit $N\to\infty$ (with $a\to 0$ and β fixed), Eq. (2) converges to:

$$S_E[x] \;=\; \int_0^\beta \mathrm{d}\tau\;\Big[\frac{m}{2}\dot{x}(\tau)^2 + \frac{1}{2}m\omega^2x(\tau)^2 + \lambda x(\tau)^4\Big]\,, \qquad x(\beta) = x(0)$$

where periodicity directly reflects the trace.

At this stage, the quantum thermal problem has been converted into a finite-dimensional integral:

$$\langle\mathcal{O}\rangle_\beta = \frac{\int\Big(\prod_{i=0}^{N-1}\mathrm{d}x_i\Big)\;\mathcal{O}(\boldsymbol{x})\,\mathrm{e}^{-S_E[\boldsymbol{x}]}}{\int\Big(\prod_{i=0}^{N-1}\mathrm{d}x_i\Big)\;\mathrm{e}^{-S_E[\boldsymbol{x}]}}$$

For $\lambda = 0$ this is a Gaussian integral and can be done analytically; for $\lambda > 0$ it is non-Gaussian and one must sample it.

3 Euclidean correlators and the spectral gap

The Euclidean correlator is defined operatorially by:

$$G(\tau) = \frac{1}{Z(\beta)}\,\mathrm{Tr}\Big(\mathrm{e}^{-\beta H}\,x(\tau)\,x(0)\Big)\,, \qquad x(\tau) = \mathrm{e}^{\tau H} x\,\mathrm{e}^{-\tau H}$$

Insert a complete energy eigenbasis, $H|n\rangle = E_n|n\rangle$, twice:

$$\begin{aligned} G(\tau) &= \frac{1}{Z}\sum_m \mathrm{e}^{-\beta E_m}\langle m|\mathrm{e}^{\tau H} x\,\mathrm{e}^{-\tau H} x|m\rangle \\ &= \frac{1}{Z}\sum_{m,n} \mathrm{e}^{-\beta E_m}\,\mathrm{e}^{\tau E_m}\,\mathrm{e}^{-\tau E_n}\,\langle m|x|n\rangle\langle n|x|m\rangle \end{aligned}$$

Using $\langle m|x|n\rangle\langle n|x|m\rangle = |\langle m|x|n\rangle|^2$:

$$\boxed{G(\tau) = \frac{1}{Z(\beta)}\sum_{m,n} \mathrm{e}^{-\beta E_m}\,\mathrm{e}^{-\tau(E_n-E_m)}\,|\langle m|x|n\rangle|^2\,, \qquad 0 \le \tau \le \beta} \tag{3}$$

This identity is the exact bridge between Euclidean data and the spectrum. Two immediate consequences are worth keeping explicit because they guide the analysis code:

(i) Periodicity and reflection symmetry. From the trace and the definition domain one has $G(\tau) = G(\beta-\tau)$ (up to the usual subtleties at coincident points). The repository optionally enforces this symmetry at the level of measured correlators using `symmetrize_periodic_correlator` in `src/analysis.py`; when enabled, bootstrap replicas are symmetrized consistently.

(ii) Low-temperature dominance and cosh structure. If $\beta(E_1 - E_0) \gg 1$, the Boltzmann weights suppress $m \neq 0$ in Eq. (3) and the correlator is dominated by the ground sector:

$$G(\tau) \approx \sum_{n\ge 1} |\langle 0|x|n\rangle|^2 \left[\mathrm{e}^{-(E_n-E_0)\tau} + \mathrm{e}^{-(E_n-E_0)(\beta-\tau)}\right]$$

If, within a fit window, a single excitation dominates, this reduces to the two-sided exponential model:

$$\boxed{G(\tau) \simeq A\left(\mathrm{e}^{-\Delta\tau} + \mathrm{e}^{-\Delta(\beta-\tau)}\right)\,, \qquad \Delta = E_1 - E_0} \tag{4}$$

The repository uses exactly Eq. (4) for gap extraction via `fit_gap_cosh` (weighted fit with `scipy.optimize.curve_fit`) and propagates uncertainty by refitting bootstrap correlators (`bootstrap_gap_from_correlators`).

We can now obtain an **effective mass estimator**.

On the lattice $\tau_k = ka$ and $G_k \equiv G(\tau_k)$. If Eq. (4) holds locally, then:

$$\frac{G_{k-1} + G_{k+1}}{2G_k} = \cosh(\Delta a)$$

which suggests the local estimator:

$$\boxed{\Delta_{\mathrm{eff}}(\tau_k) = \frac{1}{a}\,\mathrm{arcosh}\left(\frac{G_{k-1} + G_{k+1}}{2G_k}\right)\,, \qquad k = 1, \ldots, N-2} \tag{5}$$

This is implemented in `effective_mass_from_correlator` with two practical safeguards: (i) when the arcosh argument is noisy and dips below 1, the code can either clip it to $1 + \varepsilon$ (default behavior) or mark the point invalid as `NaN` (`invalid_as_nan=True`); and (ii) one can optionally apply an SNR mask using correlator error bars (`snr_min`).

A stable plateau of $\Delta_{\mathrm{eff}}(\tau)$ is the empirical signature that Eq. (4) is a good effective description in that Euclidean-time window; the fit window defaults to fractions of β via `default_fit_window_indices` (default 0.2β to 0.5β).

4 Analytic checkpoints: harmonic limit and weak-coupling trend

The purpose of the analytic layer in this project is not to "solve" the model—that would miss the point of PIMC—but to pin down reference formulas against which the numerical pipeline can be validated.

Harmonic case ($\lambda = 0$, $\omega > 0$)

For $\lambda = 0$ the model is Gaussian. Write the Hamiltonian in terms of ladder operators:

$$H_0 = \omega\left(a^\dagger a + \frac{1}{2}\right), \qquad [a, a^\dagger] = 1, \qquad E_n^{(0)} = \omega\left(n + \frac{1}{2}\right)$$

The partition function is a geometric series:

$$Z_0(\beta) = \sum_{n=0}^{\infty} \mathrm{e}^{-\beta\omega(n+1/2)} = \frac{\mathrm{e}^{-\beta\omega/2}}{1 - \mathrm{e}^{-\beta\omega}} = \frac{1}{2\sinh(\beta\omega/2)}$$

To obtain the Euclidean correlator, use:

$$x = \sqrt{\frac{1}{2m\omega}}\,(a + a^\dagger), \qquad a(\tau) = \mathrm{e}^{-\omega\tau} a, \qquad a^\dagger(\tau) = \mathrm{e}^{\omega\tau} a^\dagger$$

so:

$$x(\tau) = \sqrt{\frac{1}{2m\omega}}\left(\mathrm{e}^{-\omega\tau} a + \mathrm{e}^{\omega\tau} a^\dagger\right)$$

Thermal averages satisfy:

$$\left\langle a^\dagger a\right\rangle_\beta = n_B, \qquad \left\langle a a^\dagger\right\rangle_\beta = n_B + 1, \qquad n_B = \frac{1}{\mathrm{e}^{\beta\omega} - 1}$$

and all other quadratic contractions vanish. A direct contraction gives:

$$\begin{aligned} G_{\text{harm}}(\tau) &= \langle x(\tau) x(0)\rangle_\beta \\ &= \frac{1}{2m\omega}\left[(n_B + 1)\mathrm{e}^{-\omega\tau} + n_B \mathrm{e}^{\omega\tau}\right] \end{aligned}$$

It is convenient to rewrite it in a manifestly $\tau \leftrightarrow \beta - \tau$ symmetric form:

$$\boxed{G_{\text{harm}}(\tau) = \frac{1}{2m\omega}\,\frac{\cosh\big(\omega(\beta/2 - \tau)\big)}{\sinh(\beta\omega/2)}} \tag{6}$$

The repository uses Eq. (6) as an exact reference through `harmonic_correlator` in `src/analysis.py`. At $\tau = 0$ one obtains the exact second moment:

$$\left\langle x^2\right\rangle_{\text{harm}} = G_{\text{harm}}(0) = \frac{1}{2m\omega}\coth(\beta\omega/2)$$

implemented as `harmonic_x2`. In the low-temperature limit $\beta\omega \gg 1$, $G_{\text{harm}}(\tau) \to \frac{1}{2m\omega}\mathrm{e}^{-\omega\tau}$ and the gap is exactly $\Delta = \omega$; this is the sharpest benchmark for the effective-mass plateau and the cosh fit.

First-order quartic correction

Turning on λ makes the measure non-Gaussian; nevertheless, first-order stationary perturbation theory gives a controlled weak-coupling trend for the spectrum. Write:

$$H = H_0 + \lambda x^4, \qquad E_n = E_n^{(0)} + \lambda \left\langle n|x^4|n\right\rangle_0 + O(\lambda^2)$$

With $x = \frac{1}{\sqrt{2m\omega}}(a + a^\dagger)$, define $A = a + a^\dagger$ so that:

$$x^4 = \frac{1}{4m^2\omega^2} A^4$$

A compact way to evaluate diagonal matrix elements is to use the normal-ordering identity:

$$A^4 =: A^4 : +6 : A^2 : +3,$$

which follows from repeated use of $[a, a^\dagger] = 1$ (equivalently, from Wick's theorem for a single mode).

Since $\left\langle n\right| : A^2 : \left|n\right\rangle = 2n$ and $\left\langle n\right| : A^4 : \left|n\right\rangle = 6n(n-1) + 12n$ (only terms with equal numbers of a and $a^\dagger$ survive), one finds the standard result:

$$\left\langle n|A^4|n\right\rangle = 3(2n^2 + 2n + 1), \qquad \left\langle n|x^4|n\right\rangle_0 = \frac{3}{4m^2\omega^2}(2n^2 + 2n + 1)$$

Therefore:

$$E_n = \omega \left(n + \frac{1}{2}\right) + \lambda \frac{3}{4m^2\omega^2}(2n^2 + 2n + 1) + O(\lambda^2)$$

and the corresponding first-order shift of the fundamental gap is:

$$\boxed{\Delta \equiv E_1 - E_0 = \omega + \frac{3\lambda}{m^2\omega^2} + O(\lambda^2)} \tag{7}$$

This formula is not used as an input anywhere in the code; it serves as a weak-coupling sanity trend (the quartic term stiffens the potential and increases the gap). At moderate coupling, perturbation theory quickly becomes quantitatively unreliable (the series is asymptotic), which is precisely where direct sampling of the Euclidean measure becomes the right tool.

5 PIMC in practice

At finite N, Eq. (2) defines a probability density on $\mathbb{R}^N$:

$$\pi_N(\boldsymbol{x}) = \frac{1}{Z_N}\,\mathrm{e}^{-S_E[\boldsymbol{x}]}$$

which is strictly positive for the real potential in Eq. (1); there is no sign problem in this model. PIMC constructs a Markov chain whose stationary distribution is π_N, and estimates observables as sample means over stored configurations after thermalization.

A single run is defined by an immutable `RunConfig` (module `src/config.py`), built from: `PotentialParams` (m, ω, λ), `LatticeParams` (β, N) with $a = \beta/N$, and `SamplerParams` (backend and hyperparameters).

The high-level orchestration lives in `src/experiment.py`: `run_single_case` executes one chain plus analysis and persistence; `run_lambda_scan` repeats `run_single_case` over a list of λ values (with deterministic seed strides); `run_experiment` is a thin wrapper used by scripts and tests.

Samplers: local red-black Metropolis and global HMC

Two update strategies are implemented in `src/sampler.py` under `run_pimc`.

Red-black (or even-odd) Metropolis (`method="metropolis_rb"`). Because the action in Eq. (2) couples only nearest neighbors, even sites are conditionally independent given the odd sites (and vice versa) when N is even. The code therefore uses checkerboard half-sweeps (even indices, then odd indices) and vectorizes each half update. The proposal is:

$$x_i' = x_i + \eta, \qquad \eta \sim \mathcal{U}[-w, w]$$

and the acceptance probability is $\min(1, \mathrm{e}^{-\Delta S_i})$, with the *exact* single-site action difference:

$$\begin{aligned}\Delta S_i = \frac{m}{2a}\Big[&(x_i' - x_{i-1})^2 + (x_{i+1} - x_i')^2 - (x_i - x_{i-1})^2 - (x_{i+1} - x_i)^2\Big]\\ &+ a\big[V(x_i') - V(x_i)\big]\end{aligned}$$

implemented as `local_delta_action` in `src/action.py`. The sampler enforces N even for this backend (otherwise the exact even/odd decoupling with periodic boundaries is lost); the code raises a clear error and suggests either choosing even `n_slices` or switching to HMC.

Hybrid Monte Carlo (`method="hmc"`). HMC produces global proposals by introducing auxiliary momenta p_i and defining the fictitious Hamiltonian:

$$\mathcal{H}(\boldsymbol{x}, \boldsymbol{p}) = S_E[\boldsymbol{x}] + \sum_{i=0}^{N-1} \frac{p_i^2}{2M_{\mathrm{HMC}}}$$

with algorithmic mass M_{HMC} (parameter `hmc_mass`). Each proposal draws Gaussian momenta with variance M_{HMC}, integrates Hamilton's equations in fictitious time using leapfrog (`hmc_step_size`, `hmc_n_leapfrog`), then accepts with a Metropolis test based on the Hamiltonian defect $\Delta\mathcal{H}$. Leapfrog requires the full gradient of the Euclidean action:

$$\frac{\partial S_E}{\partial x_i} = \frac{m}{a}\big(2x_i - x_{i-1} - x_{i+1}\big) + a\,\frac{\partial V}{\partial x_i}, \qquad \frac{\partial V}{\partial x} = m\omega^2 x + 4\lambda x^3$$

implemented as `action_gradient` in `src/action.py`. The full action `euclidean_lattice_action` is used for the acceptance step. This backend does not require even N and is the default choice in the notebook pipeline.

Tuning policy. A practical point that matters for reproducibility: proposal adaptation is restricted to thermalization sweeps only. During thermalization, the code adjusts the active control parameter (`proposal_width` for Metropolis, `hmc_step_size` for HMC) with a smooth multiplicative controller based on recent acceptance rates. After thermalization, tuning is frozen and production proceeds with fixed dynamics, so standard Markov-chain estimators apply without ambiguity.

Correlator estimator and uncertainty propagation

Given stored configurations with shape (N_{stored}, N), the translationally averaged correlator for a single configuration $\boldsymbol{x}$ is:

$$g_k(\boldsymbol{x}) = \frac{1}{N}\sum_{i=0}^{N-1} x_i\, x_{i+k \pmod N}, \qquad k = 0, \ldots, N-1$$

This average over Euclidean-time origins is not simply cosmetic: it uses time-translation invariance to reduce variance substantially. Computing g_k naively costs $O(N^2)$ per configuration; the repository instead uses the circular autocorrelation identity:

$$g(\boldsymbol{x}) = \frac{1}{N}\,\text{IFFT}\Big(\text{FFT}(x)\,\overline{\text{FFT}(x)}\Big)_{\mathbb{R}}$$

implemented in `correlator_single_configuration` and vectorized across samples in `correlator_per_configuration` (`src/observables.py`). An optional `subtract_mean` flag computes the correlator of $x - \langle x \rangle_{\boldsymbol{x}}$ per configuration; for the symmetric potential the exact $\langle x \rangle$ vanishes, but mean subtraction can still reduce finite-sample drift and stabilize the FFT estimator in some regimes.

Because configurations along the Markov chain are correlated, the repository estimates correlator uncertainties by binning plus bootstrap (`bootstrap_correlator`):

- compute g_k for each stored configuration;
- bin consecutive configurations into bins of size B and replace each bin by its mean (this damps short-range autocorrelation);
- bootstrap-resample the bins with replacement to produce n_{boot} correlator replicas and extract standard deviations and (optionally) 16–84% bands.

The analysis stage can optionally symmetrize correlators under $k \leftrightarrow N-k$ before extracting effective masses and fitting gaps; when enabled, symmetrization is applied consistently to the mean correlator, its error bars, and all bootstrap replicas.

Gap extraction and diagnostics

Given a correlator estimate G_k and errors σ_k, the pipeline proceeds exactly as the theory suggests:

- compute $\Delta_{\text{eff}}(\tau)$ using Eq. (5) and check for a plateau;
- fit the correlator in a chosen window to the two-parameter model Eq. (4) (`fit_gap_cosh`);
- propagate uncertainty by repeating the same fit over bootstrap correlator replicas (`bootstrap_gap_from_correlators`).

The default fit window is chosen as a fraction of β (by default 0.2β to 0.5β), mapped to lattice indices with a minimum of three interior points. In practice the plateau behaviour of Δ_{eff} is the correct guide for whether the default window is sensible at fixed (β, N) and coupling.

The repository also computes a simple integrated autocorrelation-time estimate for the scalar series:

$$y_s = \frac{1}{N} \sum_{i=0}^{N-1} \big(x_i^{(s)}\big)^2$$

via `integrated_autocorrelation_time` (`src/analysis.py`). This is used to report two operational diagnostics: an effective sample size $N_{\text{eff}} \simeq N_{\text{stored}} / \max(1, 2\tau_{\text{int}})$ and a suggested bin size $B \simeq \lceil 2\tau_{\text{int}} \rceil$. The notebook pipeline optionally runs a short pilot at $\lambda = 0$ to auto-select a bin size before launching the full λ scan.

6 Conclusions

The overall logic of the repository is exactly the logical chain developed in this note: derive the Euclidean lattice action Eq. (2), validate the correlator against the harmonic closed form Eq. (6), and then use the same correlator-to-spectrum bridge Eq. (3) to extract gaps nonperturbatively at $\lambda > 0$.