HIP Cosmology Seminars

Boltzmann Equation Solver for Thermalization

Jong-Hyun Yoon
Chungnam National University

National Research Foundation of Korea
www.kisti.re.kr
Korea Institute of Science and Technology Information

Based on arXiv:2603.28848,
submitted to Comput. Phys. Commun.
Code: github.com/best-hep/best

University of Helsinki, Helsinki
24 June 2026

UNIVERSITY OF HELSINKI
HELSINKI INSTITUTE OF PHYSICS

Outline

- Motivation: when do we need the full Boltzmann equation?
- Method: direct Monte Carlo evaluation
- The identical-particle decomposition
- Implementation
- Validation and results
- Discussion and outlook

Outline

- Motivation: when do we need the full Boltzmann equation?
- Method: direct Monte Carlo evaluation
- The identical-particle decomposition
- Implementation
- Validation and results
- Discussion and outlook

Motivation I: reheating after inflation

The standard estimate of the reheating temperature:

$$T_{\rm reh} \sim \left(\frac{90}{\pi^2 g_*}\right)^{1/4} \sqrt{\Gamma_\phi M_{\rm Pl}}$$

assumes instantaneous thermalization of the inflaton decay products.

But the actual initial state is maximally non-thermal:

- few, highly energetic particles, $E \sim m_\varphi \gg T_{reh}$
- a thermal bath at the same energy density consists of many soft particles
- the bridge between the two states is a dynamical question — *how fast, and through which momentum modes, does the spectrum relax?*

Full equilibrium requires number-changing processes

- $2 \to 2$ scattering conserves particle number: it can redistribute momenta (kinetic equilibration) but cannot adjust N at fixed E
- the end point of $2 \to 2$ alone: Bose–Einstein with some $\mu \neq 0$ fixed by the initial (N, E)
- full (chemical) equilibrium $\mu = 0$ requires $n \to m$ processes: splittings, $2 \to 3$, . . .

Consequence

A first-principles treatment of reheating needs the momentum-resolved Boltzmann equation with $n_{in} \neq n_{out}$ collision terms — precisely what existing solvers do not provide.

Motivation II: dark sectors beyond freeze-out

Where the standard machinery breaks:

- **Cannibal / SIMP dark matter:** relic set by 3 → 2 (or 2 → 3) self-interactions
- **Freeze-in → freeze-out transition:** distribution far from kinetic equilibrium;⟨σv⟩ thermal averaging invalid
- **Dark sector internal thermalization:** no contact with SM bath; T, μ not given, must emerge dynamically

Common thread

The shape of f (p, t) matters, and the relevant processes have nin ≠ nout.

Also: neutrino decoupling, baryogenesis, phase transitions with time-dependent masses.

The hierarchy of approximations

Level	Solves for	Assumes
nBE	n(t)	equilibrium shape of f ; ⟨σv⟩
cBE	n(t), T(t)	kinetic equilibrium (shape up to T, μ)
relaxation time	f (p, t)	C[f] ≈ −(f − feq)/τ: one timescale, no momentum-exchange information
fBE	f (p, t)	nothing about the shape

Related tools: micrOMEGAs, DarkSUSY, MadDM, DRAKE, ...

Outline

- Motivation: when do we need the full Boltzmann equation?
- Method: direct Monte Carlo evaluation
- The identical-particle decomposition
- Implementation
- Validation and results
- Discussion and outlook

Setup: Boltzmann equation in an expanding background

$$\frac{\partial f_a}{\partial t} - Hp\frac{\partial f_a}{\partial p} = \sum_{\text{processes}} C_a[f]$$

Working with comoving momentum q = a p, the Liouville term becomes a total derivative at fixed q:

$$\frac{df_a(q,t)}{dt} = \sum_{\text{processes}} C_a[f]$$

- this is the form implemented: grid in q, physical p = q/a(t) inside the collision integral
- user-defined a(t); built-in radiation domination; H = 0 for the flat-spacetime studies shown today
- time-dependent masses m(t) supported (phase transitions, thermal corrections)

The collision integral and its cost

$$C^{(k)}(\mathbf{p}) = \frac{1}{2E}\int d\Pi\, |\mathcal{M}|^2\, \Lambda \qquad \Lambda = \prod_{i\in\text{in}} f_i \prod_{j\in\text{out}} (1 \pm f_j) - \prod_{j\in\text{out}} f_j \prod_{i\in\text{in}} (1 \pm f_i) \qquad d\Pi = (2\pi)^4 \delta^{(4)}\left(\sum_{i\in\text{in}} p_i - \sum_{j\in\text{out}} p_j\right) \prod_{l\neq k} \frac{d^3 p_l}{(2\pi)^3 2E_l}$$

- full Bose enhancement / Pauli blocking, no approximation
- must be evaluated at every grid point p, every time step
- dimensionality after fixing the observed leg and using momentum conservation: 3(ntotal – 2)

2 → 2: 6 dims 2 → 3: 9 dims general n → m: 3(n + m – 2)

Strategy of BEST: do not reduce analytically — integrate directly with adaptive Monte Carlo.

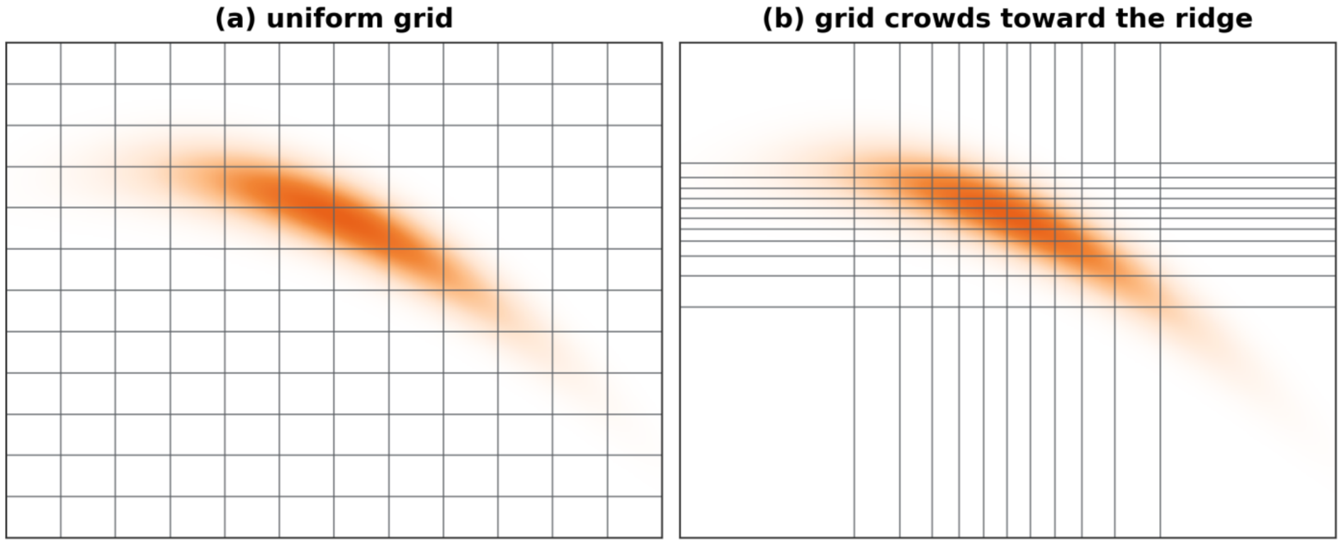
Why Vegas?

- adaptive importance sampling (Lepage ’78, ’21): the grid map concentrates evaluations where the integrand is large
- error $\propto 1/\sqrt{N_{\mathrm{eval}}}$, independent of dimension
- vectorized batch mode: integrand evaluated on O(10^4) points per call (NumPy arrays)
- integrator reuse: the adapted map is kept across grid points and time steps — the integrand changes slowly, so only minor re-adaptation is needed

Two integrators per process

Gain and loss terms integrated separately: avoids cancellation between two large numbers;
each map adapts to its own integrand.

How the adaptive map works



each axis is rescaled so every cell carries equal integrand weight

- Each axis is rescaled independently so every cell carries equal integrand weight — grid lines crowd where the integrand is large (a → b)
- Sampling uniformly in the warped coordinates then concentrates evaluations on the ridge — same number of points, far more useful ones

Phase-space construction: three particle roles

For isotropic f (|p|), fix the observed particle along z.

- Observed (iobs): pinned at grid momentum p — the argument of C[f](p)
- Conserved (icons): 3-momentum fixed exactly by

$$\mathbf{p}_{\text{cons}} = \sum_{\substack{i\in\text{in}\\ i\neq\text{cons}}} \mathbf{p}_i - \sum_{\substack{j\in\text{out}\\ j\neq\text{cons}}} \mathbf{p}_j \qquad E_{\text{cons}} = \sqrt{|\mathbf{p}_{\text{cons}}|^2 + m_{\text{cons}}^2}$$

- Integrated: remaining ntotal − 2 particles, spherical coordinates (rk , θk , φk), Jacobian $r_k^2 \sin\theta_k$

Placement of the conserved particle

For nin ≠ nout: put it on the side with fewer particles (e.g. initial state for 2 → 3). Then pcons, reconstructed as a difference of totals, is kinematically allowed (real, on-shell energy) over a broad region of the sampled phase space — crucial for MC efficiency.

Energy conservation: a deliberate approximation

$$\delta(E_{\rm in} - E_{\rm out}) \approx \frac{1}{\sigma_E\sqrt{2\pi}}\,\mathrm{exp}\left[-\frac{(E_{\rm in} - E_{\rm out})^2}{2\sigma_E^2}\right], \qquad \sigma_E = w\,\frac{E_{\rm in} + E_{\rm out}}{2}$$

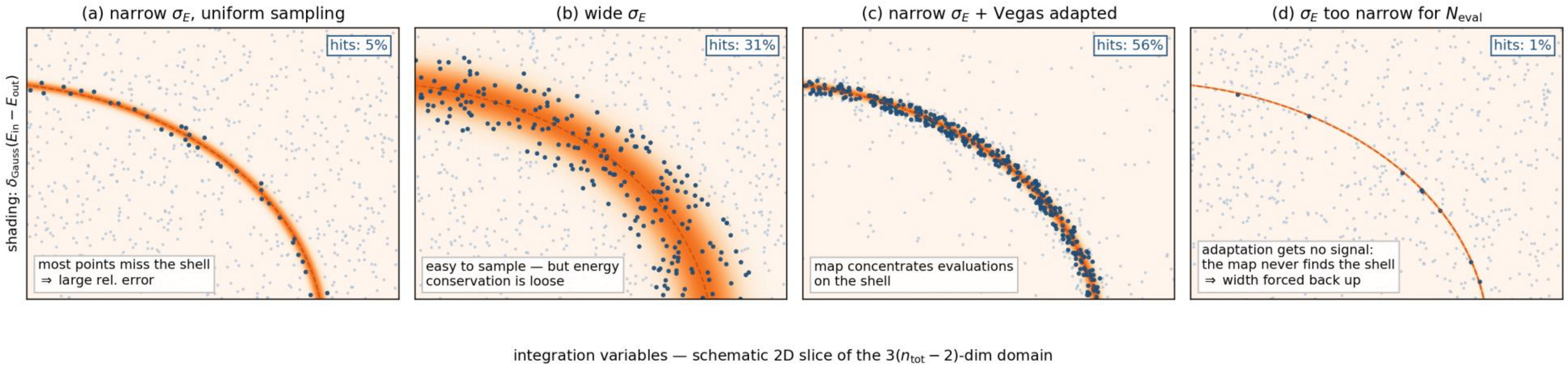
“Why not use the δ to remove one dimension exactly?”

- exact treatment requires solving for one momentum magnitude ⇒ case-dependent roots that do not generalize to arbitrary n → m
- the Gaussian keeps one uniform construction for any multiplicity

The price — and how it is controlled:

- smaller w: stricter energy conservation, but the integrand lives on a thinner shell ⇒ harder to sample, larger MC error
- the code adapts w automatically: widened when the relative MC error exceeds an upper threshold (max rel err, default 0.1), narrowed below a lower threshold (min rel err, default 0.01)
- empirically (2 → 2): Neval ∼ 10^5 with initial w ∼ 10^−2 is stable; an order of magnitude fewer evaluations ⇒ w driven up, energy conservation lost

The width–sampling trade-off, in pictures



- smaller w is not monotonically better: below what the map can resolve at a given Neval, the relative error grows and the solver widens w automatically
- the usable window of w therefore grows with Neval: more evaluations buy tighter energy conservation

Outline

- Motivation: when do we need the full Boltzmann equation?
- Method: direct Monte Carlo evaluation
- The identical-particle decomposition
- Implementation
- Validation and results
- Discussion and outlook

Where does the observed momentum sit?

For φφ ↔ φφφ: a scattering event changes fφ(p) whenever p coincides with any of p1, . . . , p5:

$$C_a(\mathbf{p}) = \sum_{k=1}^{n_{\text{total}}} \delta_{a,s_k}\, C^{(k)}(\mathbf{p})$$

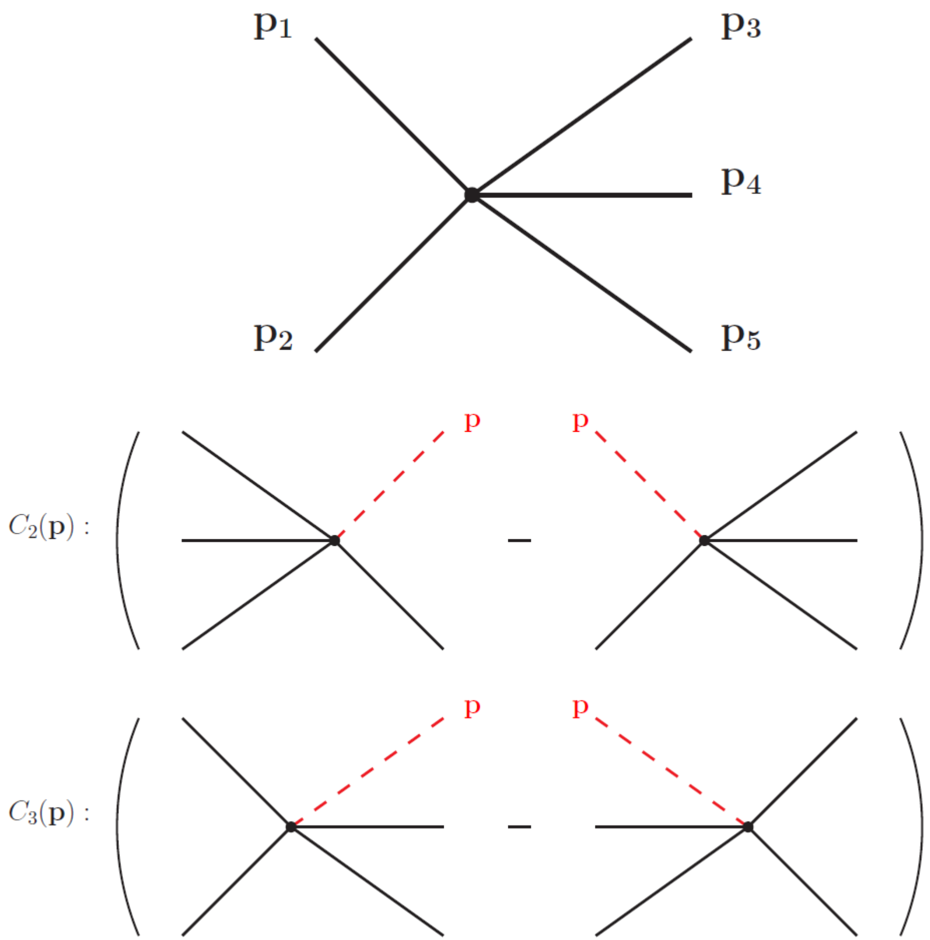
k enters through the measure: the observed leg is excluded from the integration, so which side it sits on changes which momenta are integrated.

Permutation symmetry of dummy variables ⇒ only the side matters:

$$\boxed{C_a(\mathbf{p}) = n^a_\alpha\, C_{n_\alpha}(\mathbf{p}) + n^a_\beta\, C_{n_\beta}(\mathbf{p})}$$

- $n^a_{\alpha,\beta}$: multiplicity of species a on each side; |M|^2 includes symmetry factors
- φφ ↔ φφφ: C_φ = 2 C_2 + 3 C_3
- symmetric case (nα = nβ, e.g. φφ → φφ): relabeling gives Cnα = Cnβ ⇒ collapses to the familiar single term (multiplicity 2)

The decomposition, diagrammatically



Why C2 ≠ C3: the role of f (p) and the integration measure both differ.

C2(p) — p on the 2-side:

gain: f3f4f5 (1+f (p))(1+f2)

loss: f (p)f2 (1+f3)(1+f4)(1+f5)

C3(p) — p on the 3-side:

gain: f1f2 (1+f (p))(1+f4)(1+f5)

loss: f (p)f4f5 (1+f1)(1+f2)

These are different integrals — not related by relabeling when nα ≠ nβ.

Energy conservation requires the complete sum

Integrate E Ca(p) over all momenta using the full decomposition:

$$\int \frac{d^3p}{(2\pi)^3}\, E\, C_a(p) \propto \int d\Pi_{\rm all} \left(\sum_{i\in \rm in} E_i - \sum_{j\in \rm out} E_j \right) |\mathcal{M}|^2\, \Lambda$$

vanishes by the energy–momentum δ-function — but only if the sum over all positions is complete.

- omit C2 for 2 ↔ 3: $\int E\, C_3\, p^2\, dp \neq 0 \Rightarrow$ systematic energy non-conservation (not an MC artifact)
- invisible in the nBE: momentum integration absorbs the sum over positions
- invisible to all previous fBE solvers: for 2 → 2, Cnα = Cnβ automatically

Outline

- Motivation: when do we need the full Boltzmann equation?
- Method: direct Monte Carlo evaluation
- The identical-particle decomposition
- Implementation
- Validation and results
- Discussion and outlook

The code in one slide

```
from besthep import BEST

solver = BEST(q_min=0.1, q_max=20.0,
              n_grid=64)
solver.initialize_species(
    'phi', init_f,
    stat='boson', mass=1.0)
solver.add_process(
    'cannibal',
    ['phi','phi'],
    ['phi','phi','phi'],
    matrix_element,
    coupling=1.0, neval=int(1e7))
for step in range(n_steps):
    solver.evolve_step(dt=dt)
```

- Python: numpy (vectorized batches), mpi4py, scipy
- dimensionality, phase-space construction, and the 2C2+3C3 decomposition: all automatic from the process declaration
- quantum statistics per species (boson/fermion/MB)
- time-dependent masses: set mass func('phi', ...)
- checkpoint/restart: single pickle file = full state (grids, f , Vegas maps, history)

Momentum-dependent matrix elements

```
def matrix_element(momenta, coupling):
    # momenta: (n_total, 3, N)
    # particles in declared order
    # (incoming first); Cartesian
    # components; Vegas batch index
    p1, p2 = momenta[0], momenta[1]
    E1 = np.sqrt((p1**2).sum(0) + m**2)
    E2 = np.sqrt((p2**2).sum(0) + m**2)
    s  = (E1 + E2)**2 \
       - ((p1 + p2)**2).sum(0)
    # e.g. s-channel Breit-Wigner:
    return coupling**2 * mR**4 / \
      ((s - mR**2)**2 + mR**2*GR**2)
```

- only 3-momenta are passed; energies reconstructed on-shell from the declared masses
- Mandelstam variables built directly from the array — fully vectorized over the batch
- worked s- and t-channel Breit–Wigner example shipped with the code
- the conserved particle's momentum vector enters here — which is why its sign convention matters

Interpolation and the tails of f (p)

On the grid: cubic spline (linear scale).

Beyond the grid — and this matters: the collision integral is sensitive to tail behavior through the statistical factors.

- fit $\log(1/f + \eta) = a + bE$ near each edge of the grid ($\eta = +1/-1/0$ for BE/FD/MB)
- this is the exact functional form of an equilibrium tail — the extrapolation degrades gracefully as the system thermalizes
- separate fits at the low- and high-momentum edges

Time stepping and parallelization

Time integration

- explicit Euler and Heun (predictor–corrector); adaptive Δt via $\max_p |\Delta t\, C[f]/f| < \epsilon$ ($\epsilon = 0.3$ here)

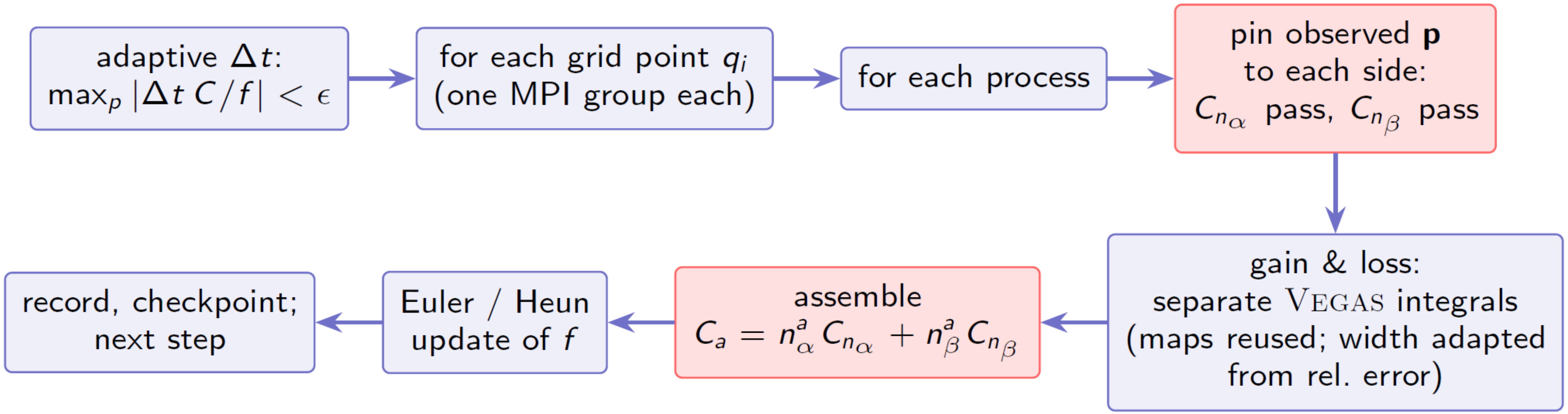
Parallelization

- momentum grid points are independent ⇒ distribute across MPI ranks
- near-linear scaling up to $N_{proc} \approx N_{grid}$; additional internal Vegas parallelism beyond that (n r parallel groups)

Output

- single pickle file = checkpoint and primary output: f , grids, process definitions, Vegas maps, full history; plotting script included

One time step, end to end



- the two highlighted boxes are where n → m generality lives: the two-pass structure and the multiplicity-weighted assembly — both automatic from the process declaration
- cost anatomy of the ~ 300 s 2 → 3 step: dominated by the two Vegas passes at Neval = 10^7 each

Outline

- Motivation: when do we need the full Boltzmann equation?
- Method: direct Monte Carlo evaluation
- The identical-particle decomposition
- Implementation
- Validation and results
- Discussion and outlook

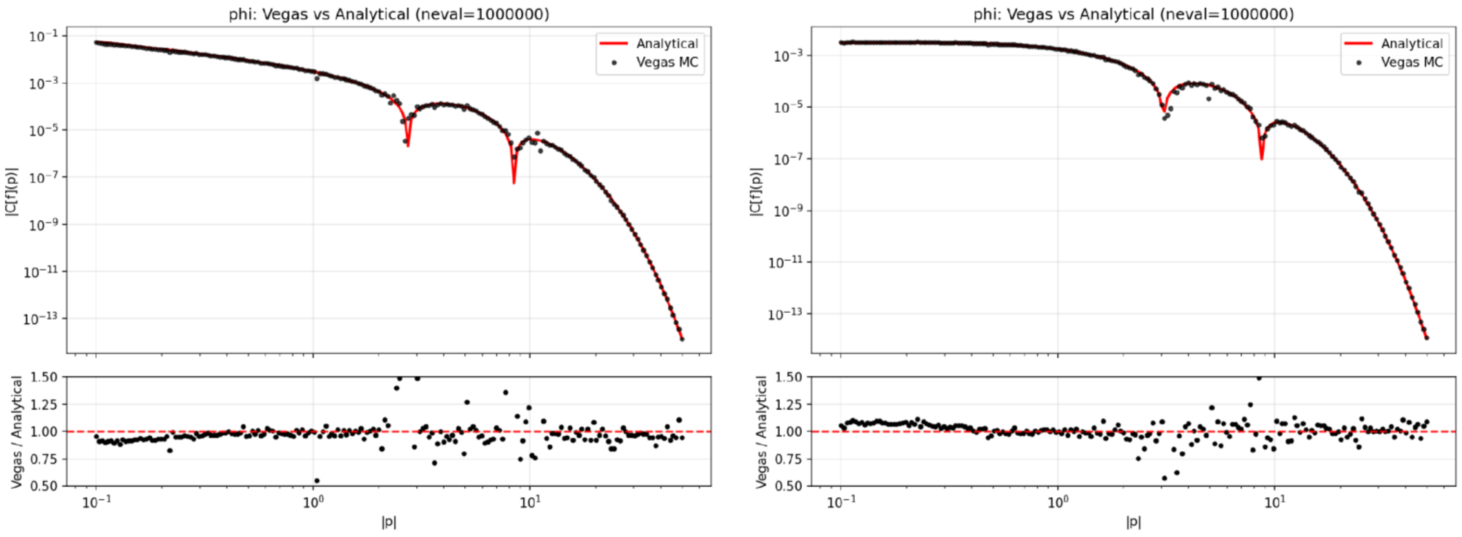
Validation: semi-analytical 2 → 2 benchmark

Independent cross-check, following Ala-Mattinen et al. '22:

$$C_{\rm BW}(p_1) = \frac{2}{(2\pi)^4}\frac{1}{2E_1}\int dp_3\,\frac{p_3^2}{2E_3}\int dp_4\,\frac{p_4^2}{2E_4}\,F(p_1,p_3,p_4)\,f_3 f_4(1\pm f_1)(1\pm f_2)$$

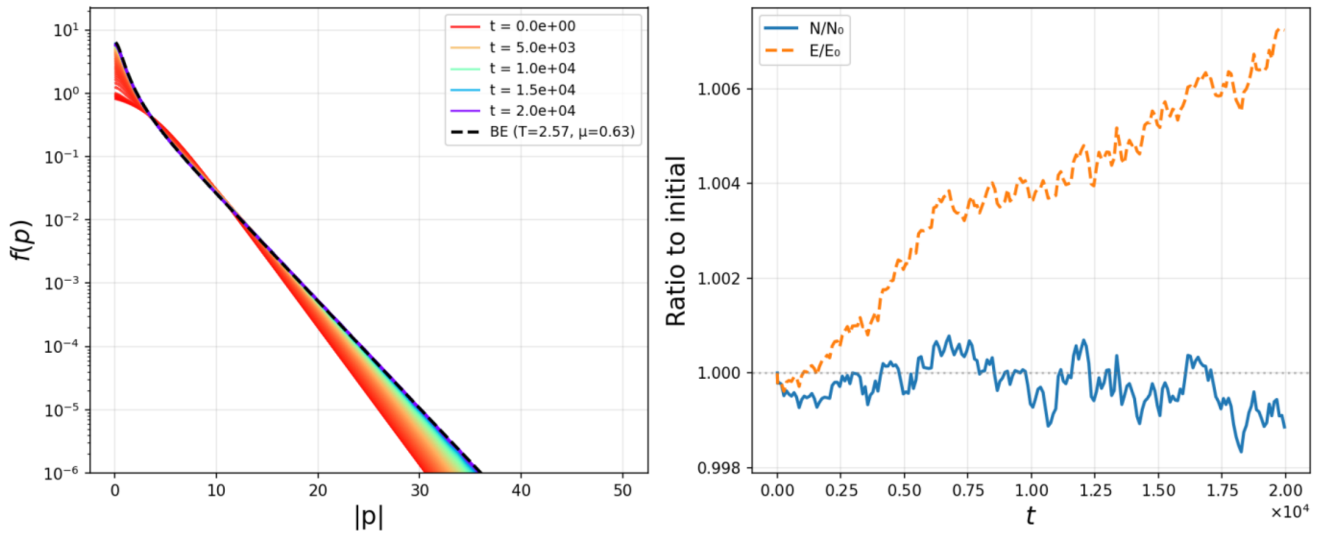
- angular integrations done analytically; 9 → 2 dimensions
- energy conservation exact (E2 = E3 + E4 − E1, no Gaussian)
- ⇒ tests both the MC accuracy and the systematic error from the finite δ-width

Benchmark result



- massless and massive φ, non-thermal f , constant |M|^2, H = 0; both methods include the identical-particle multiplicity of 2
- agreement at the few-% level where the collision rate is large
- lower panels: ratio of the net rate (gain − loss), whose denominator passes through zero — by-eye deviations near the zero crossings reflect the vanishing denominator plus low MC signal-to-noise there

Thermalization: 2 → 2 elastic scattering



- no thermal input anywhere: the Bose–Einstein form, the Bose-stimulated low-p enhancement, and (Teq, μ) all emerge from |M|^2 + quantum statistics + conservation
- the endpoint is a prediction, not a fit: (Teq, μ) fixed by the conserved N and E
- gain = loss at fBE is not built in — it must emerge between two independently adapted integrators
- N, E conserved to ~ 1% over the full evolution (MC noise level)

Which way does the system relax?

For φφ ↔ φφφ, chemical equilibrium forces 3μ = 2μ ⇒ μ = 0: number-changing processes erase the chemical potential. Only energy is conserved.

So the end point is fully predicted before running anything:

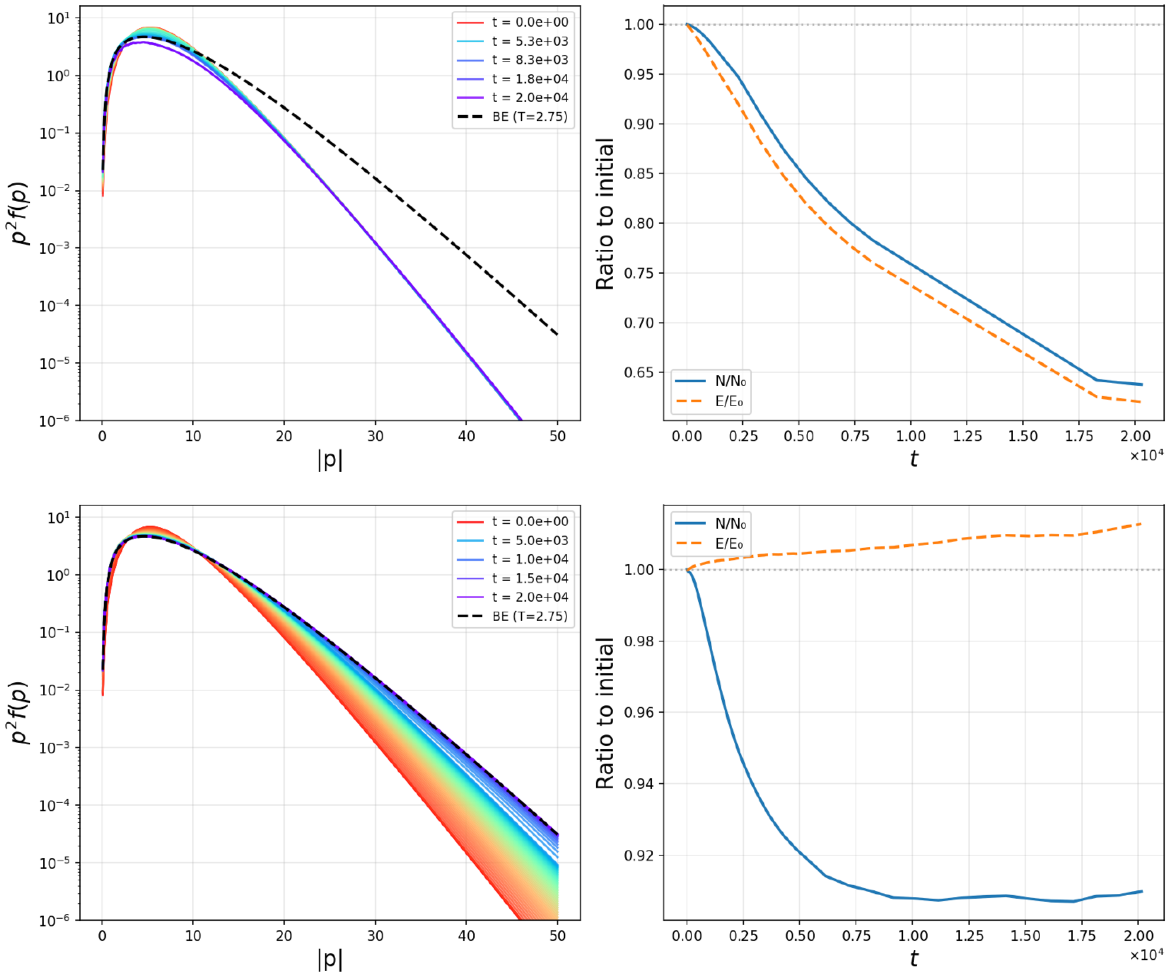
$$E_{\mathrm{BE}}(T_{\mathrm{eq}}, \mu{=}0) \stackrel{!}{=} E_0 \;\Rightarrow\; T_{\mathrm{eq}}, \qquad N_{\mathrm{eq}} = N_{\mathrm{BE}}(T_{\mathrm{eq}}, \mu{=}0)$$

For the initial condition f0(p) = [1 + e(p−3)/2]^−1, mφ = 1:

	initial	μ = 0 equilibrium
⟨E⟩ = E/N	7.00	7.77
T	—	2.75
N/N0	1	0.90

- initial ⟨E⟩ below the equilibrium value ⇒ too many particles for the available energy ⇒ net 3 → 2: N must decrease (∼10%) — genuine cannibalization
- the direction is set by the initial condition, not by the process label "2 → 3"

The key test: φφ ↔ φφφ



Top: naive implementation (C3 only)

- ~ 40% energy loss — systematic, not statistical
- wrong equilibrium: missing C2 distorts the production/absorption balance

Bottom: full 2 C2 + 3 C3

- energy conserved to ~ 1%
- f → Bose–Einstein with μ = 0 (chemical eq.: 3μ = 2μ)
- N/N0 → 0.9: the predicted ~10% decrease from the previous slide — net 3 → 2, μ → 0

Computational performance

	2 → 2	2 → 3
dimensions	6	9
Ngrid	272	136
Neval	10^6	10^7
MPI ranks	272	544 (4/grid point)
time per step	~ 50 s (Heun)	~ 300 s (Euler)

- 2 → 3 cost dominated by the two passes (C2 and C3)
- production runs: KISTI Nurion; full thermalization histories in a few hours wall time
- laptop: debugging / reduced-parameter validation (Ngrid ~ 64, single process, O(hour) example included with the code)

Outline

- Motivation: when do we need the full Boltzmann equation?
- Method: direct Monte Carlo evaluation
- The identical-particle decomposition
- Implementation
- Validation and results
- Discussion and outlook

Limitations

- **Cost**: direct MC in 9+ dims is expensive — this is the price of generality. Production n → m runs need a cluster; not (yet) a laptop tool for full projects.
- **MC noise floor from gain–loss cancellation**: the absolute error on $C_{net} = C_{gain} - C_{loss}$ scales with the individual terms, not the net. Where both are small (tails) this is harmless; where they are large and cancelling (balance points, late times everywhere) the noise floor persists as the true $C \to 0$ — ⇒ %-level random-walk drift of conserved quantities, and the slowest stages of relaxation sink below the noise.
- **Explicit time stepping**: Euler/Heun with adaptive Δt. Adequate here because the thermalization problems studied are not stiff; a genuinely stiff regime would favor implicit methods (cf. Drake), which Best does not currently provide.
- **Energy conservation is approximate by construction**: controlled, monitored, adaptive — but a Gaussian width, not a δ.

Outlook

- **Reheating**: evolve the actual decay-product spectrum through number-changing processes ⇒ thermalization timescale and a refined Treh beyond the instantaneous approximation
- **Cannibal / SIMP relic computations** with the full f (p, t), including the regime where kinetic equilibrium within the dark sector is not maintained
- **Multi-species networks**: conversions, decays, inelastic channels — same process interface, differing only in input/output assignment; cost scales linearly with the number of processes (each integrated independently), dominated by per-process phase-space dimensionality
- **Momentum-dependent** |M|^2: array interface momenta[particle, xyz, batch]; worked Breit–Wigner example shipped with the code

Summary

- BEST: momentum-resolved Boltzmann solver for arbitrary n → m processes — direct Vegas MC in 3(ntotal − 2) dimensions, quantum statistics, massive species, expansion, MPI-parallel
- exact momentum conservation + adaptive Gaussian energy conservation: one uniform construction for any multiplicity
- identical particles with nin ≠ nout require the full decomposition $C_a = n^a_\alpha C_{n_\alpha} + n^a_\beta C_{n_\beta}$
- validated against an exact semi-analytical 2 → 2 benchmark; 2 ↔ 3 thermalization demonstrated with %-level conservation

github.com/best-hep/best

All the BEST.