Machine Learning in Spinfoam Cosmology

Athanasios Kogios
in collaboration with Joseph Bunao, Pietropaolo Frisoni and Jared Wogan [1]

Perimeter Institute

May 6, 2024

[1] paper to appear

- Apply machine learning [2] to improve computations of observables in spin foams.
- Use a well defined context, spin foam cosmology.
- Compare against established computations [3] to test its effectiveness.

[2] [Bengio et al., 2021]

[3] [Frisoni, Gozzini, and Vidotto, 2023; Han et al., 2021; Dona and Frisoni, 2023; Steinhaus, 2024]

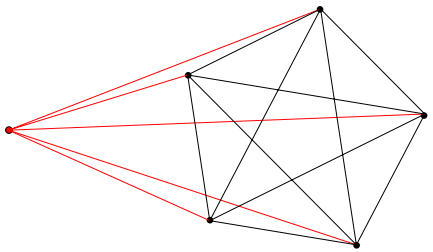
Physical Motivation

Physical Motivation

- Universe starting with a Big Bang needs initial conditions, knowledge about the past.
- Instead, consider a no-boundary initial state with no matter, no space and no time (nothing).
- Analogue of a sphere: it encompasses everything, but does not have a boundary.
- Quantum tunnelling to observable (almost) homogeneous and isotropic universe (something). *(see Francesca's talk earlier)*
- Is there a way to express the transition amplitude?

Spin Foams

Spin Foams



The nothing to something transition as a 4-simplex.

- Encode transition amplitude (change) between two boundary 3D spatial geometries (spin networks).

- 4-simplex spin foam amplitude as a transition from nothing to something.

4-simplex

- Apply it to a cosmological context: fix boundary edges to have the same spin value j (homogeneity) and impose regularity, being in any node should not change the "view" (isotropy).
- The dof are given by the boundary intertwiners (volume fluctuations of the boundary tetrahedra).
- Number of dof is large but spin quantum numbers are small (full quantum regime).
- Compute an observable!

Observables

Observables I

- Dihedral angle operator: external dihedral angle between faces a and b on the tetrahedron dual to node k (careful, not the usual definition).

- Notion of local geometry.

- In the spin basis:

$$\langle j, i_n | \cos(\theta)_k | j, i_n \rangle = \frac{i_k(i_k+1) - 2j(j+1)}{2j(j+1)} \tag{1}$$

$j \to$ boundary links (remember homogeneous sector), $i \to$ intertwiners and $|j, i_n\rangle = |j, i_1\rangle \otimes \ldots \otimes |j, i_N\rangle$.

Observables II

- The corresponding expectation value:

$$\langle \cos(\theta)_k \rangle = \frac{1}{Z} \sum_{i_n} W(j, i_n)^2 \langle j, i_n | \cos(\theta)_k | j, i_n \rangle \ , \tag{2}$$

- where the normalization factor is:

$$Z \equiv \langle \psi | \psi \rangle = \sum_{i_n} W(j, i_n)^2 \ . \tag{3}$$

- and

$$W(j, i_n)^2 \tag{4}$$

is the EPRL vertex amplitude given in terms of $15j$ symbols. *(Pietro's talk this morning)*

MCMC

MCMC

- In (1) all j assume the same (fixed for each simulation) value.
- i_k are the only dof, where $i_k \in [0, 2j+1]$.
- Due to the symmetries each vertex is in $1-1$ correspondence with a dihedral angle.
- For 4-simplex, 5 intertwiners.
- Assume 5-dimensional coordinate system, each intertwiner mapped to a coordinate.
- Probability distribution (1) defined on the 5-dim discrete space (hypergrid), that the MCMC tries to learn.

GFlowNets

Advantages

- Not possible to overtrain. The more visited trajectories the better!
- Training expenses are compensated by being faster as sampler.
- Instead of stochastically progressing like MCMC, use visited states to make educated guess for high probability areas.
- Locating far regions of high probability and alternating among them, better approximation.

Results

Results I

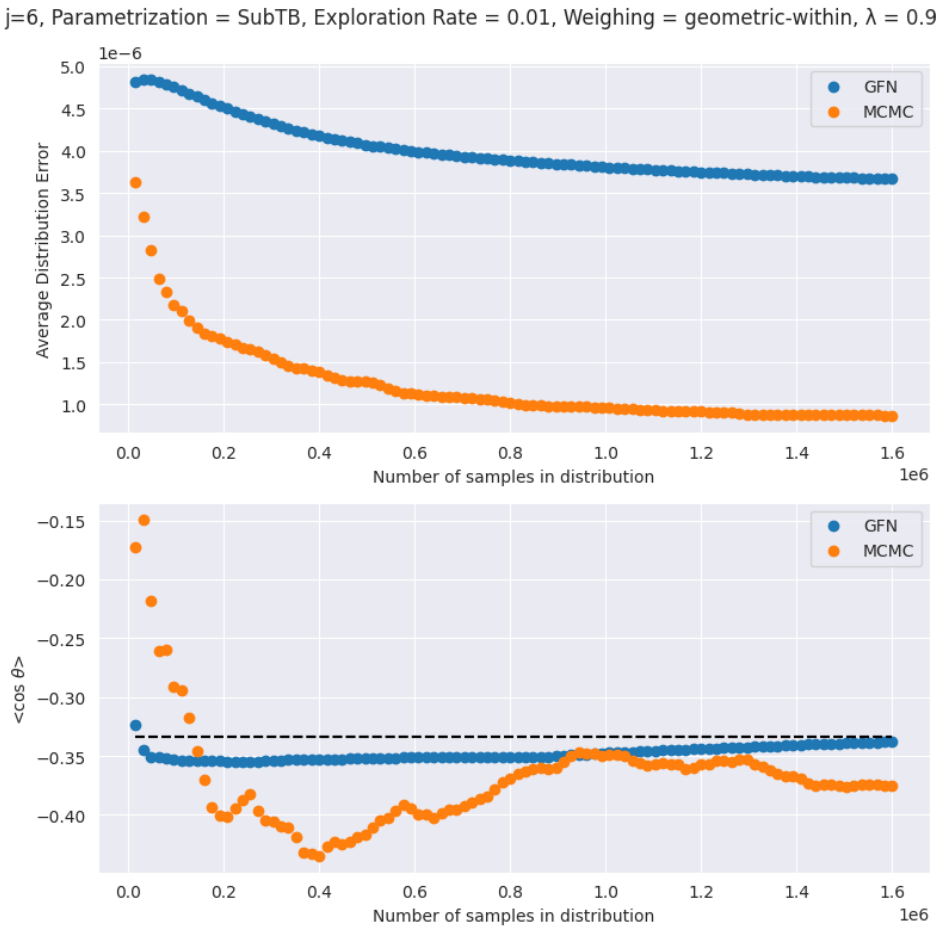


Figure 1: Upper: L1 error, Lower: Observable estimation for SubTB, exploration: rate 0.01, weighing: geometric-within, λ: 0.9.

Results II

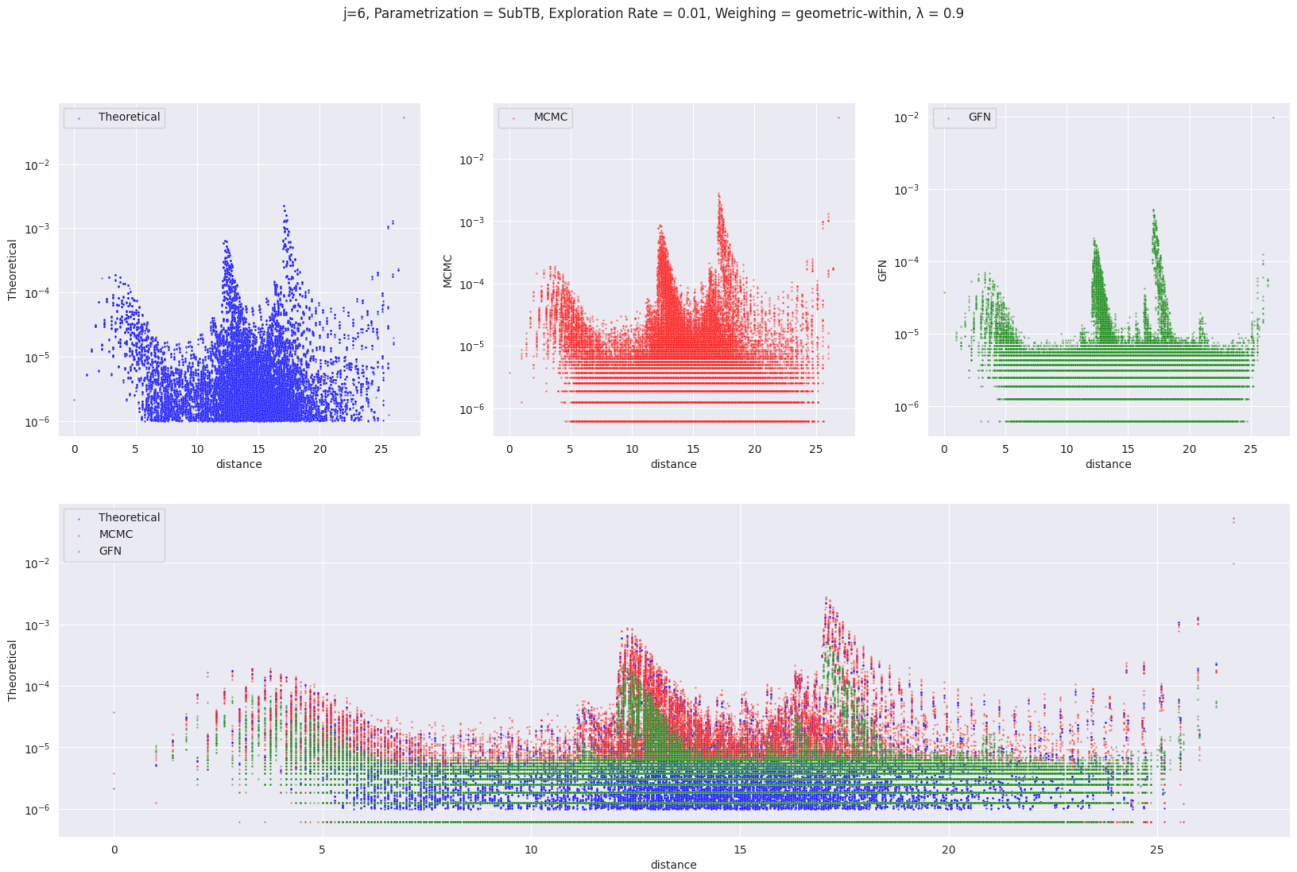


Figure 2: Euclidean distance plot for SubTB, exploration: rate 0.01, weighing: geometric-within, λ: 0.9.

Discussion

Discussion I

- Different philosophies.
- GFlowNet searches for the most important, therefore, mostly contributing peaks. MCMC tries to learn the whole simulation (thus, better L1 error for MCMC, but further from the value).
- GFlowNet performs better in cases where the peaks are far apart (more complicated).

Discussion II

- Hybrid algorithm for more complicated cases: GFlowNet to locate the peaks, MCMC to learn the corresponding, underlying distribution.
- GFlowNet was run with limited parameters. Run more tests.
- Maybe not the correct problem: initially for building drug molecules. Apply GFlowNets in problems involving constructive actions eg building new states from old ones.

- Apply machine learning to improve computations of observables in spin foams.
- Use a well defined context, spin foam cosmology.
- Compare against established computations to test its effectiveness.

Bibliography I


Bengio, Emmanuel et al. (2021). "Flow Network based Generative Models for Non-Iterative Diverse Candidate Generation". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., pp. 27381–27394. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/e614f646836aaed9f89ce58e837e2310-Paper.pdf.


Frisoni, Pietropaolo, Francesco Gozzini, and Francesca Vidotto (2023). "Markov chain Monte Carlo methods for graph refinement in spinfoam cosmology". In: *Class. Quant. Grav.* 40.10, p. 105001. DOI: 10.1088/1361-6382/acc5d6. arXiv: 2207.02881 [gr-qc].


Han, Muxin et al. (2021). "Spinfoam on a Lefschetz thimble: Markov chain Monte Carlo computation of a Lorentzian spinfoam propagator". In: *Physical Review D* 103.8. ISSN: 2470-0029. DOI: 10.1103/physrevd.103.084026. URL: http://dx.doi.org/10.1103/PhysRevD.103.084026.


Dona, Pietro and Pietropaolo Frisoni (2023). *Summing bulk quantum numbers with Monte Carlo in spin foam theories*. arXiv: 2302.00072 [gr-qc].

Bibliography II

 Steinhaus, Sebastian (Mar. 2024). “A Monte Carlo algorithm for spin foam intertwiners”. In: arXiv: 2403.04836 [gr-qc].