Spinfoam on a Lefschetz thimble: Markov chain Monte Carlo computation of a Lorentzian spinfoam propagator

Muxin Han, Zichang Huang, Hongguang Liu, Dongxue Qu, and Yidun Wan

Spinfoam on Lefschetz Thimble: Markov Chain Monte-Carlo Computation of Lorentzian Spinfoam Propagator

Muxin Han

Department of Physics, Florida Atlantic University,
777 Glades Road, Boca Raton, FL 33431-0991, USA and
Institut für Quantengravitation, Universität Erlangen-Nürnberg, Staudtstr. 7/B2, 91058 Erlangen, Germany

Zichang Huang

State Key Laboratory of Surface Physics, Fudan University, Shanghai 200433, China and
Department of Physics, Center for Field Theory and Particle Physics,
and Institute for Nanoelectronic devices and Quantum computing, Fudan University, Shanghai 200433, China

Hongguang Liu

Institut für Quantengravitation, Universität Erlangen-Nürnberg, Staudtstr. 7/B2, 91058 Erlangen, Germany

Dongxue Qu

Department of Physics, Florida Atlantic University,
777 Glades Road, Boca Raton, FL 33431-0991, USA

Yidun Wan

State Key Laboratory of Surface Physics, Fudan University, Shanghai 200433, China
Department of Physics, Center for Field Theory and Particle Physics,
and Institute for Nanoelectronic devices and Quantum computing, Fudan University, Shanghai 200433, China and
Department of Physics and Institute for Quantum Science and Engineering,
Southern University of Science and Technology, Shenzhen 518055, China

We compute numerically the Lorentzian Engle-Pereira-Rovelli-Livine (EPRL) spinfoam propagator on a 4-simplex, by adapting the methods of Lefschetz thimble and Markov Chain Monte-Carlo to oscillatory spinfoam integrals. Our method can compute any spinfoam observables at relatively large spins. We obtain the numerical results of the propagators at different spins and demonstrate their consistency with the expected spinfoam semi-classical behavior in the large spin limit. Our results exhibit significant quantum corrections at smaller spins. Our method is reliable and thus can be employed to discover the semi-classical and quantum behaviors of the spinfoam model.

I. INTRODUCTION

Computing and studying properties of the n-point correlation functions are the core problem in quantum field theory. As emphasized by Arthur Wightman, a complete quantum field theory can be uniquely reconstructed from its n-point correlation functions. The n-point functions in Loop Quantum Gravity (LQG) [1–4], which is a promising candidate of the quantum gravity theory, have been introduced in [5, 6]. In [7–10], analysis has been carried out on the 2-point correlation function for the Penrose metric operators in the spinfoam model—the covariant formulation of LQG [2, 11–14]. This 2-point functions are usually called the spinfoam propagator. The analysis confirms that the semiclassical behavior of the spinfoam propagators, determined by the large-spin asymptotics, matches the one obtained from the Regge calculus, and indicates that LQG recovers General Relativity in 4 dimensions in an appropriate limit. Asymptotic analysis of the spinfoam amplitudes have drawn similar conclusion[15–25]. Despite the existence of the novel and crucial results brought in by these discussions of spinfoam propagators, the computational complexity has been obstructed further explorations in the spinfoam model. Nevertheless, numerical approaches to spinfoams open new windows to circumvent this obstruction. The works [26–28] have attempted numerical computing of the spinfoam amplitudes and obtained enlightening results. There are also numerical results on the spinfoam renormalization [29, 30]. In early attempts, the numerical computations of th spinfoam correlation function for 3d gravity [31] and 4d Barrett–Crane model [32, 33] are done. Yet, the numerical computation of the spinfoam correlation function for Engle-Pereira-Rovelli-Livine (EPRL) model has not been well developed.

In this paper, we propose and employ a numerical method that combines Lefschetz thimble and Markov Chain Monte-Carlo to compute the spinfoam propagator on a 4-simplex in the EPRL spinfoam model. In contrast to the algorithm in [26–28] which mostly applicable to spinfoam amplitudes with small spins, our algorithm is featured by computing spinfoams with relatively large spins. We obtain the numerical results of the propagators at different spins and compare with the spinfoam large spin asymptotics. Our numerical results are shown to be consistent with the

spinfoam asymptotics in the large spin limit while providing significant corrections at smaller spins.

In the path integral formalism, the expectation value of an arbitrary observable $\mathcal{O}[\varphi]$ can be expressed as

$$\langle \hat{\mathcal{O}} \rangle = \frac{\int D\varphi\, \mathcal{O}[\varphi]\, \mathrm{e}^{-S[\varphi]}}{\int D\varphi\, \mathrm{e}^{-S[\varphi]}}, \tag{1}$$

where S is the action. The spinfoam propagator can be expressed in terms of similar integral expressions [9]

$$G_{mn}^{abcd} = \left\langle E_n^a \cdot E_n^b E_m^c \cdot E_m^d \right\rangle - \left\langle E_n^a \cdot E_n^b \right\rangle \left\langle E_m^c \cdot E_m^d \right\rangle, \tag{2}$$

where each E_n^a is the flux operator at the face shared by tetrahedra n and a. The action S for the propagator is a complex valued function depending on 54 real variables (for one 4-simplex). Both the denominator and the nominator in the right hand side of Eq. (1) are integrals of high-dimensional oscillatory functions. Known as the sign problem [34–36], these oscillatory integrals cannot be evaluated in the conventional Monte Carlo integral method, thus have been difficult to be studied numerically.

In order to compute $\langle O \rangle$, recent progresses (see e.g. [37–42]) suggest to apply the *Picard-Lefschetz* theory to cure the sign problem, and have applied the theory to numerical computations in Lattice Field Theories. In our work, we apply the theory to the numerical computation of the spinfoam propagator and improve the numerical method for handling higher dimensional oscillatory integrals. The idea of the method is to deform the integration contour to a class of specific integral cycles, called *Lefschetz thimbles* $\mathcal{J}_\sigma$, each of which is defined as the union of all steepest descent paths ending at a given critical point of the analytic continued action. The deformation does not change the value of the integral, while it has the advantage that the imaginary part of S is constant on each Lefschetz thimble $\mathcal{J}_\sigma$, so e^{-S} becomes non-oscillatory on the thimble. The integral on each $\mathcal{J}_\sigma$ can be studied numerically with the Monte-Carlo method.

Technically, the numerical integration on the Lefschetz thimble $\mathcal{J}_\sigma$ can be achieved by the Markov Chain Monte Carlo (MCMC) method [43] [1], which treats the $\langle O \rangle$ on the thimble as the mean value of $O e^{i \arg(\det J)}$ among samples on $\mathcal{J}_\sigma$ given by the Boltzmann distribution $\mathrm{e}^{-\mathrm{Re}(S) + \log |\det J|}$ where $\det J$ is the measure factor on $\mathcal{J}_\sigma$. Specifically, in order to do the numerical integration on a high-dimensional thimble, we use a multi-chain MCMC method called the Differential Evolution Adaptive Metropolis (DREAM) algorithm [46] to compute the $\langle O \rangle$ on the thimble.

The stationary phase analysis shows that in the large spin regime, G_{mn}^{abcd} receives the dominant contribution from a single critical point corresponding to the Lorentizian geometrical 4-simplex. This infers that for large spins, the Lefschetz thimble attached to this geometrical critical point dominates the integrals for computing the propagator, while it turns out that the Lefschetz thimbles of other critical point only give exponentially small contributions.

In the case where the contribution to $\langle O \rangle$ of certain thimble is much larger than the contributions of the other thimbles, one can compute the integral on the dominant thimble and use it as a proper approximation of $\langle O \rangle$. The integral on the Lefschetz thimble can generate the same perturbative expansion as the stationary phase analysis [47]. Thus this approximation keeps all the *perturbative* quantum corrections by integrating along the dominant thimble, while neglecting *non-perturbative* corrections from other thimbles whose contributions are exponentially suppressed.

In our computation, the operators expectation values $\left\langle E_n^a \cdot E_n^b E_m^c \cdot E_m^d \right\rangle$, $\left\langle E_n^a \cdot E_n^b \right\rangle$, $\left\langle E_m^c \cdot E_m^d \right\rangle$ are computed numerically on the dominant thimble attached to the geometrical critical point. This computation capture perturbative quantum corrections to all orders in the spinfoam model, while neglecting non-perturbative corrections that are exponentially small for large spins. Given that the computational complexity is scaled exponentially large when computing perturbative expansion to high orders, our method with Lefschetz thimble is efficient and powerful.

In practice, we compute the spinfoam propagator with the coherent boundary state peaked at the boundary data of a Lorentizian geometrical 4-simplex. We scale the boundary spins by λ and numerically compute the propagator with $\lambda \in [10^2, 5 \times 10^7]$. In the regime $\lambda > 10^4$, our numerical result strongly tends to comply with the expected semiclassical limit. At smaller spins, the numerical results provide significant quantum corrections. This consistency supports the reliability of our algorithm and increases our confidence to use the algorithm as a tool to discover the semi-classical and quantum behaviors of the spinfoam model.

We emphasize that our method applies to a much boarder range of research areas involving the complex-valued actions and oscillatory integrals, e.g., finite density lattice QCD, Lee-Yang zeros, non-Hermitian systems, gauge theory with topological θ-term, etc. Similar methods have been applied in these areas [39, 48–53]. We look forward to finding novel interesting results by applying our method to these areas.

This paper is organized as follows: Section II discusses the Lorentzian spinfoam propagator in the EPRL model, the boundary state, and our parameterizations of the action the observables. Section III explains our algorithm, including

---

[1] See [44, 45] for some existing works on applying Monte Carlo method to LQG/Spinfoam.

reviewing the Picard-Lefschetz theory, MCMC, and DREAM. Section IV discusses the application of Lefschetz thimble to the spinfoam model and a few conceptual aspects. Section V discusses several optimizations to improve the efficiency of the computation. Section VI discusses the stationary phase approximation and the large spin limit of the spinfoam propagator. Section VII presents our numerical results from the Monte-Carlo on Lefschetz-thimble and compares with the large spin limit. Section VIII presents the benchmarks the platforms that we use for testing our code. In Section IX, we conclude and mention a few future perspectives.

We tested our codes in *Mathematica*™12. The codes are posted on [54].

II. SPINFOAM PROPAGATOR

In this section, we review the EPRL spinfoam model, including the definition of the vertex amplitude and the definition of the spinfoam propagator, and we setup the boundary data and critical point used in the numerical computation. In the integral representation of the spinfoam vertex amplitude, the integrand is a function depending on 54 real variables. Furthermore, we present four integrals to be computed by our numerical algorithm for computing the spinfoam propagator.

A. Boundary State

The boundary state is crucial in calculating the spinfoam amplitude. Following the basic setup in [9], our boundary state defined below is a *Lorentzian semi-classical state* peaked at both intrinsic and extrinsic geometry. We consider a 4-dimensional Lorentzian spacetime region $\mathcal{R}$ homeomorphic to a 4-ball whose 3-dimensional boundary $\partial\mathcal{R}$ is homeomorphic to a 3-sphere. Taking the coarsest triangulation, $\mathcal{R}$ is a 4-simplex, and $\partial\mathcal{R}$ is triangulated by five tetrahedra. Specifically, in this paper, we use the boundary geometry introduced in [26, 55]. Namely, we set the five vertices of the 4-simplex as

$$P_1 = (0,0,0,0), \ P_2 = (0,0,0,-2\sqrt{5}/3^{1/4}), \ P_3 = (0,0,-3^{1/4}\sqrt{5},-3^{1/4}\sqrt{5}),$$
$$P_4 = (0,-2\sqrt{10}/3^{3/4},-\sqrt{5}/3^{3/4},-\sqrt{5}/3^{1/4}), \ P_5 = (-3^{-1/4}10^{-1/2},-\sqrt{5/2}/3^{3/4},-\sqrt{5}/3^{3/4},-\sqrt{5}/3^{1/4}).$$

Then, the 4-dimensional normal vectors of the tetrahedra[2] are given by:

$$N_1 = (-1,0,0,0), \ N_2 = \left(\frac{5}{\sqrt{22}}, \sqrt{\frac{3}{22}}, 0, 0\right), \ N_3 = \left(\frac{5}{\sqrt{22}}, -\frac{1}{\sqrt{66}}, \frac{2}{\sqrt{33}}, 0\right),$$

$$N_4 = \left(\frac{5}{\sqrt{22}}, -\frac{1}{\sqrt{66}}, -\frac{1}{\sqrt{33}}, \frac{1}{\sqrt{11}}\right), \ N_5 = \left(\frac{5}{\sqrt{22}}, -\frac{1}{\sqrt{66}}, -\frac{1}{\sqrt{33}}, -\frac{1}{\sqrt{11}}\right). \tag{3}$$

The areas of the 10 faces are uniformly proportional to the data shown in Table I, and the 3-dimensional normal of each face[3] is shown in Table II.

TABLE I. Each cell shows the area of the face shared by line number tetrahedra and column number tetrahedra.

area j_{0ab} \ b a	2	3	4	5
1	5	5	5	5
2	＼	2	2	2
3	＼	＼	2	2
4	＼	＼	＼	2

The intrinsic and extrinsic geometry of the boundary tetrahedra are given by the face areas, 3d normals, and 4d normals. The corresponding semi-classical boundary quantum state is a superposition of coherent spin-network states

$$|\Psi_0\rangle = \sum_{\lambda j_{ab}} \psi_{\lambda j_0, \zeta_0} ||\lambda j_{ab}, \vec{n}_{ab}\rangle. \tag{4}$$

----

[2] Here, the we use i ($= 1, \ldots, 5$) to label the tetrahedron that does not contain the vertex P_i.
[3] We use the pair (ab) to indicate the face of the tetrahedron a pointing to tetrahedron b.

TABLE II. Each cell shows the 3-dimensional normal vector of the face shared by line number tetrahedra and column number tetrahedra.

normal $\vec{n}_{ab}$, a \ b	1	2	3	4	5
1	＼	(1,0,0)	(-0.33,0.94,0)	(-0.33,-0.47,0.82)	(-0.33,-0.47,-0.82)
2	(-1,0,0)	＼	(0.83,0.55,0)	(0.83,-0.28,0.48)	(0.83,-0.28,-0.48)
3	(0.33,-0.94,0)	(0.24,0.97,0)	＼	(-0.54,0.69,0.48)	(-0.54,0.69,-0.48)
4	(0.33,0.47,-0.82)	(0.24,-0.48,0.84)	(-0.54,0.068,0.84)	＼	(-0.54,-0.76,0.36)
5	(0.33,0.47,0.82)	(0.24,-0.48,-0.84)	(-0.54,0.068,-0.84)	(-0.54,-0.76,-0.36)	＼

We have denoted the spins by λj_{ab} and λj_{0ab} where j_{0ab} are recorded in Table I and λ is a scaling parameter. The large spin limit corresponds to large λ. The five *Livine-Speziale coherent intertwiner* $||\lambda j_{ab}, \vec{n}_{ab}\rangle$ are compatible with the $\vec{n}_{ab}$ in Table II. The wave packet ψ_{j_0, ζ_0} reads

$$\psi_{\lambda j_0, \zeta_0} = \exp\left(-i \sum_{ab} \zeta_0^{ab}(\lambda j_{ab} - \lambda j_{0ab})\right) \exp\left(-\sum_{ab,cd} \alpha^{(ab)(cd)} \frac{\lambda j_{ab} - \lambda j_{0ab}}{\sqrt{\lambda j_{0ab}}} \frac{\lambda j_{cd} - \lambda j_{0cd}}{\sqrt{\lambda j_{0cd}}}\right),$$

where ζ_0^{ab}, whose values are given in TABLE III, is related to the dihedral angles between the 4-normals (3) (see Section II D for the way to determine ζ_0^{ab}). The spin variables λj_{0ab} correspond to the areas listed in TABLE I. The matrix $\alpha^{(ab)(cd)}$ has positive definite real part, and

$$\alpha^{(ab)(cd)} = \alpha_1 P_0^{(ab)(cd)} + \alpha_2 P_1^{(ab)(cd)} + \alpha_3 P_2^{(ab)(cd)},$$

where $\alpha_1, \alpha_2, \alpha_3$ are free parameters, and $P_k^{(ab)(cd)}$ $(k = 0 \cdots 2)$ are defined as

- $P_0^{(ab)(cd)} = 1$ if $(ab) = (cd)$ and zero otherwise;

- $P_1^{(ab)(cd)} = 1$ if $a = c$, $b \neq d$ and zero otherwise;

- $P_2^{(ab)(cd)} = 1$ if $(ab) \neq (cd)$ and zero otherwise.

The spinfoam amplitude with coherent spin-networks as the boundary state depends on the free parameters α. In our numerical computation, we set $\alpha_1 = 0.2, \alpha_2 = 0.3, \alpha_3 = 0.4$ for definiteness. Any other choice of α does not affect the application of our algorithm.

TABLE III. The table of ζ_0^{ab}

ζ_0^{ab}, a \ b	2	3	4	5
1	-3.14+0.36γ	0.68+0.36γ	5.05+0.36γ	5.05+0.36γ
2	＼	5.05-0.59γ	-5.93-0.59γ	-3.20-0.59γ
3	＼	＼	-2.81-0.59γ	-5.54-0.59γ
4	＼	＼	＼	-4.37-0.59γ

B. Spinfoam Action

In the boundary formalism [2, 56, 57], the spinfoam amplitude for the boundary state $|\Psi_0\rangle$ can be written as

$$\langle W|\Psi_0\rangle = \sum_{j_{ab}} \psi_{\lambda j_0, \zeta_0} \langle W||\lambda j_{ab}, \vec{n}_{ab}\rangle, \tag{5}$$

where $\langle W|$ is a $\mathbb{C}$-valued linear functional providing the sum over the bulk geometries with the weight that defines our model for quantum gravity. The Lorentzian EPRL vertex amplitude can be expressed as

$$\langle W|\Psi_0\rangle = \sum_{j_{ab}} \psi_{j_0, \zeta_0} \int_{SL(2,\mathbb{C})^5} \prod_a dg_a \prod_{a>b} P_{ab}(g), \tag{6}$$

with

$$P_{ab}(g) = \langle \lambda j_{ab}, -\vec{n}_{ab} \left| Y^\dagger g_a^{-1} g_b Y \right| \lambda j_{ab}, \vec{n}_{ba} \rangle. \tag{7}$$

In (7), Y maps the spin-j $SU(2)$ irreducible representation $\mathcal{H}_j$ to the lowest level in $SL(2,\mathbb{C})$ $(j,\gamma j)$-irreducible representation $\mathcal{H}_{(j,\gamma j)} = \oplus_{k=j}^\infty \mathcal{H}_k$ (See [58, 59] for alternative choices of the Y-map.)[4]. Using the fact that the elements in $\mathcal{H}_{(j,\gamma j)}$ can be expressed as homogeneous functions on $\mathbb{CP}_1$, the inner product (7) is equivalent to an integral [15, 19]

$$P_{ab} = \frac{d_{\lambda j_{ab}}}{\pi} \int_{\mathbb{CP}^1} \mathrm{d}\tilde{\mathbf{z}}_{ab} \langle Z_{ba}, Z_{ba} \rangle^{-(1-\mathrm{i}\gamma)\lambda j_{ab}} \langle Z_{ab}, Z_{ab} \rangle^{-(1+\mathrm{i}\gamma)\lambda j_{ab}} \langle J\xi_{ab}, Z_{ab} \rangle^{2\lambda j_{ab}} \langle Z_{ba}, \xi_{ba} \rangle^{2\lambda j_{ab}}, \tag{8}$$

in which $d_j = 2j + 1$, $Z_{ab} = g_a^\dagger z_{ab}$, and $Z_{ba} = g_b^\dagger z_{ab}$. The integral measure $\mathrm{d}\tilde{\mathbf{z}}_{ab} = -\left(\langle Z_{ab}, Z_{ab} \rangle \langle Z_{ba}, Z_{ba} \rangle\right)^{-1} \mathrm{d}\mathbf{z}_{ab}$ (with $\mathrm{d}\mathbf{z} = \frac{i}{2}(z_0 dz_1 - z_1 dz_0) \wedge (\bar{z}_0 d\bar{z}_1 - \bar{z}_1 d\bar{z}_0)$) is homogeneous on $\mathbb{CP}^1$. The bracket $\langle \ , \ \rangle$ is the Hermitian inner product on $\mathbb{C}^2$. The spinors ξ_{ba} and $J\xi_{ab}$ [5] are related to the 3-normal $\vec{n}_{ba}$ and $-\vec{n}_{ba}$ respectively by $\vec{n}_{ba} = \langle \xi_{ba}|\vec{\sigma}|\xi_{ba} \rangle$ and $-\vec{n}_{ab} = \langle J\xi_{ab}|\vec{\sigma}|J\xi_{ab} \rangle$. TABLE IV lists the spinors that compatible with our boundary geometry.

TABLE IV. Each cell indicates a spinor ξ_{ab} corresponding to a 3-normal of a tetrahedron.

$|\xi_{ab}\rangle$ \ b a	1	2	3	4	5
1	╲	(0.71,0.71)	(0.71,-0.24+0.67 i)	(0.95,-0.17-0.25 i)	(0.30,-0.55-0.78 i)
2	(0.71,-0.71)	╲	(0.71,0.59+0.39 i)	(0.86, 0.48 - 0.16 i)	(0.51, 0.82 - 0.27 i)
3	(0.71, 0.24 - 0.67 i)	(0.71, 0.17 + 0.69 i)	╲	(0.86, -0.31 + 0.40 i)	(0.51, -0.53 + 0.68 i)
4	(0.30, 0.55 + 0.78 i)	(0.96, 0.13 - 0.25 i)	(0.96, -0.28 + 0.035 i)	╲	(0.83, -0.33 - 0.46 i)
5	(0.95, 0.17 + 0.25 i)	(0.28, 0.43 - 0.86 i)	(0.28, -0.95+ 0.12 i)	(0.57, -0.48-0.67 i)	╲

Using the integral expression of P_{ab}, we can write the amplitude as

$$\langle W|\Psi_0\rangle = \sum_{\lambda j_{ab}} \psi_{\lambda j_0, \zeta_0} \langle W||\lambda j_{ab}, \vec{n}_{ab}\rangle = \sum_{\lambda j_{ab}} \psi_{\lambda j_0, \zeta_0} \int_{SL(2,\mathbb{C})^5} \prod_a \mathrm{d}g_a \int \left(\prod_{a>b} \frac{d_{\lambda j_{ab}}}{\pi} \mathrm{d}\tilde{\mathbf{z}}_{ab} \right) \mathrm{e}^{\lambda S}, \tag{9}$$

with the spinfoam action S (without the scaling parameter λ) given by

$$S(j, g, \mathbf{z}) = \sum_{a>b} \left[2j_{ab} \log(\langle J\xi_{ab}, Z_{ab}\rangle \langle Z_{ba}, \xi_{ba}\rangle) - (1 + \mathrm{i}\gamma)j_{ab} \log \langle Z_{ab}, Z_{ab}\rangle - (1 - \mathrm{i}\gamma)j_{ab} \log \langle Z_{ba}, Z_{ba}\rangle \right]. \tag{10}$$

After gauge fixing $g_1 = 1$, the action is a function of four $SL(2,\mathbb{C})$ elements g_a, $(a = 2 \cdots 5)$, ten spinors z_{ab}, and ten area variables j_{ab}. Explicitly, we parametrize each $SL(2,\mathbb{C})$ element by six real parameters as

$$g = \begin{pmatrix} 1 + \frac{x_1 + \mathrm{i}y_1}{\sqrt{2}} & \frac{x_2 + \mathrm{i}y_2}{\sqrt{2}} \\ \frac{x_3 + \mathrm{i}y_3}{\sqrt{2}} & \frac{1 + \frac{x_2 + \mathrm{i}y_2}{\sqrt{2}} \frac{x_3 + \mathrm{i}y_3}{\sqrt{2}}}{1 + \frac{x_1 + \mathrm{i}y_1}{\sqrt{2}}} \end{pmatrix}, \tag{11}$$

and we parametrize each spinor by two real parameters in a way that

$$z = (1, x + \mathrm{i}y). \tag{12}$$

Each group variable is parametrized by six real parameters, and each spinor variable is parametrized by two real parameters. As such, the action S is a function depending on 54 real parameters.

In this parametrization, the measure of the spinor dz becomes

$$\mathrm{d}\mathbf{z} = dx dy,$$

and the Haar measure dg of $SL(2,\mathbb{C})$ is expressed as (see Appendix A in [55])

$$dg = \frac{1}{128\pi^4} \frac{dx_1 dx_2 dx_3 dy_1 dy_2 dy_3}{|1 + \frac{x + \mathrm{i}y}{\sqrt{2}}|^2}.$$

---

[4] Our numerical algorithm is applicable to these alternative choices of the Y-map.
[5] For a spinor $Z = (z_1, z_2)$, $JZ = (\bar{z}_2, -\bar{z}_1)$.

The amplitude (9) is indeed a superposition of multiple 44-dimensional integrals

$$\langle W|\Psi_0\rangle = \sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \int \mathrm{d}\phi\, U(j,\phi)\, \mathrm{e}^{\lambda S(j,\phi)}, \tag{13}$$

where ϕ stands for 44 real variables parametrizing g and z, and

$$U(j,\phi) = \frac{1}{(128\pi^4)^4} \left(\prod_{a>b} -\frac{d_{\lambda j_{ab}}}{\pi} (\langle Z_{ab}, Z_{ab}\rangle \langle Z_{ba}, Z_{ba}\rangle)^{-1} \right) \prod_a \frac{1}{|(g_a)_{1,1}|^2}.$$

C. Spinfoam Propagator

Following the boundary formalism, the expectation value of an observable $\hat{\mathcal{O}}$ is defined by

$$\langle \hat{\mathcal{O}} \rangle = \frac{\langle W|\hat{\mathcal{O}}|\Psi_0\rangle}{\langle W|\Psi_0\rangle}.$$

The spinfoam propagator G_{mn}^{abcd} is constructed as [5, 7–9]

$$\begin{aligned}
G_{mn}^{abcd} =& \frac{\langle W|E_n^a \cdot E_n^b E_m^c \cdot E_m^d|\Psi_0\rangle}{\langle W|\Psi_0\rangle} - \frac{\langle W|E_n^a \cdot E_n^b|\Psi_0\rangle}{\langle W|\Psi_0\rangle} \frac{\langle W|E_m^c \cdot E_m^d|\Psi_0\rangle}{\langle W|\Psi_0\rangle} \\
=& \frac{\sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \langle W|E_n^a \cdot E_n^b E_m^c \cdot E_m^d||\lambda j_{ab}, \vec{n}_{ab}\rangle}{\sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \langle W||\lambda j_{ab}, \vec{n}_{ab}\rangle} \\
& - \frac{\sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \langle W|E_n^a \cdot E_n^b||\lambda j_{ab}, \vec{n}_{ab}\rangle}{\sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \langle W||\lambda j_{ab}, \vec{n}_{ab}\rangle} \frac{\sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \langle W|E_m^c \cdot E_m^d||\lambda j_{ab}, \vec{n}_{ab}\rangle}{\sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \langle W||\lambda j_{ab}, \vec{n}_{ab}\rangle}.
\end{aligned} \tag{14}$$

where $E_n^a \cdot E_n^b$ is the spatial metric operator at the n-th tetrahedron. By definition, each flux operator E_n^{ai} can only act on the corresponding face state $|\lambda j_{ab}, \vec{n}_{ba}\rangle$. Thus we can explicitly express $\langle W|E_n^a \cdot E_n^b E_m^c \cdot E_m^d||\lambda j_{ab}, \vec{n}_{ab}\rangle$, $\langle W|E_n^a \cdot E_n^b||\lambda j_{ab}, \vec{n}_{ab}\rangle$, and $\langle W|E_m^c \cdot E_m^d||\lambda j_{ab}, \vec{n}_{ab}\rangle$ in integral forms by inserting E_b^{ai} into P_{ab}. Since P_{ab} appears in the amplitude under the condition $a > b$[6], the operator E_b^a and the operator E_a^b are inserted in P_{ab} in two different ways. When inserting E_b^{ai} into the right hand side of Eq. (7), the P_{ab} becomes

$$\begin{aligned}
Q_b^{ai} \equiv & \left\langle \lambda j_{ab}, -\vec{n}_{ab} \left| Y^\dagger g_a^{-1} g_b Y (E_b^a)^i \right| \lambda j_{ab}, \vec{n}_{ba} \right\rangle \\
=& \frac{d_{\lambda j_{ab}}}{\pi} \int_{\mathbb{CP}^1} \mathrm{d}\tilde{\mathbf{z}}_{ab} \langle Z_{ba}, Z_{ba}\rangle^{-(1-\mathrm{i}\gamma)\lambda j_{ab}} \langle Z_{ab}, Z_{ab}\rangle^{-(1+\mathrm{i}\gamma)\lambda j_{ab}} \times \\
& \langle J\xi_{ab}, Z_{ab}\rangle^{2\lambda j_{ab}} \langle Z_{ba}, \xi_{ba}\rangle^{2\lambda j_{ab}} \lambda j_{ab}\gamma \frac{\langle \sigma_i Z_{ba}, \xi_{ba}\rangle}{\langle Z_{ba}, \xi_{ba}\rangle}.
\end{aligned} \tag{15}$$

If inserting E_a^{bi} into the right hand side of Eq. (7), the P_{ab} becomes

$$\begin{aligned}
Q_a^{bi} \equiv & \left\langle \lambda j_{ab}, -\vec{n}_{ab} \left| (E_b^a)^{i\dagger} Y^\dagger g_a^{-1} g_b Y \right| \lambda j_{ab}, \vec{n}_{ba} \right\rangle \\
& \frac{d_{\lambda j_{ab}}}{\pi} \int_{\mathbb{CP}^1} \mathrm{d}\tilde{\mathbf{z}}_{ab} \langle Z_{ba}, Z_{ba}\rangle^{-(1-\mathrm{i}\gamma)\lambda j_{ab}} \langle Z_{ab}, Z_{ab}\rangle^{-(1+\mathrm{i}\gamma)\lambda j_{ab}} \times \\
& \langle J\xi_{ab}, Z_{ab}\rangle^{2\lambda j_{ab}} \langle Z_{ba}, \xi_{ba}\rangle^{2\lambda j_{ab}} (-\lambda j_{ab}\gamma) \frac{\langle J\xi_{ab}, \sigma_i Z_{ab}\rangle}{\langle J\xi_{ab}, Z_{ab}\rangle}
\end{aligned} \tag{16}$$

Defining

$$A_{ab}^i = \gamma\lambda j_{ab} \frac{\langle \sigma^i Z_{ba}, \xi_{ba}\rangle}{\langle Z_{ba}, \xi_{ba}\rangle}, \quad A_{ba}^i = -\gamma\lambda j_{ab} \frac{\langle J\xi_{ba}, \sigma_i Z_{ba}\rangle}{\langle J\xi_{ba}, Z_{ba}\rangle}, \tag{17}$$

---

[6] We assume a is always greater than b here.

we have

$$Q_a^{b\,i} = P_{ab}A_{ab}^i, \ Q_b^{a\,i} = P_{ab}A_{ba}^i. \tag{18}$$

Then we obtain the following integral expressions of the ingredients in (14)

$$\langle W|E_n^a \cdot E_n^b E_m^c \cdot E_m^d|\Psi_0\rangle = \sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \int \mathrm{d}\phi\, U(j,\phi)\, [A_{an}(j,\phi) \cdot A_{bn}(j,\phi)]\, [A_{cm}(j,\phi) \cdot A_{dm}(j,\phi)]\, \mathrm{e}^{\lambda S(j,\phi)}, \tag{19}$$

$$\langle W|E_n^a \cdot E_n^b|\Psi_0\rangle = \sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \int \mathrm{d}\phi\, U(j,\phi)\, A_{an}(j,\phi) \cdot A_{bn}(j,\phi)\, \mathrm{e}^{\lambda S(j,\phi)}, \tag{20}$$

$$\langle W|E_m^c \cdot E_m^d|\Psi_0\rangle = \sum_{j_{ab}} \psi_{\lambda j_0,\zeta_0} \int \mathrm{d}\phi\, U(j,\phi)\, A_{cm}(j,\phi) \cdot A_{dm}(j,\phi)\, \mathrm{e}^{\lambda S(j,\phi)}. \tag{21}$$

D. Large Spin Approximation

Equations (19), (20), and (21) are all expressed as summations of the $J_{ab} \equiv \lambda j_{ab}$ in the domain of non-negative half integers. This type of the summation can be rewritten as

$$\sum_{J \in \frac{\mathbb{Z}^+}{2} \cup 0} f(J) = \frac{1}{2} \sum_{J \in \mathbb{Z}} f(|J/2|) + \frac{1}{2} f(0) = 2 \sum_{k \in \mathbb{Z}} \int_0^\infty dJ f(J) \mathrm{e}^{4\pi \mathrm{i} k J} + \frac{1}{2} f(0), \tag{22}$$

where use of the Poisson summation formula is made in the second step. Applying this formula to (19) - (21), the term $\frac{1}{2}f(0)$ is exponentially small when all the λj_{0ab} are large in $\psi_{\lambda j_0,\phi_0}$ because the term contains a Gaussian peaked at $j_{ab} = j_{0ab}$. By neglecting $\frac{1}{2}f(0)$, (19) - (21) are expressed as integrals

$$\langle W|\Psi_0\rangle = (2\lambda)^{10} \sum_{\{k_{ab}\} \in \mathbb{Z}^{10}} \int_0^\infty \mathrm{d}^{10}j \int \mathrm{d}\phi\, U\, \mathrm{e}^{-\lambda S_{tot}^{(k)}}, \tag{23}$$

$$\langle W|E_n^a \cdot E_n^b E_m^c \cdot E_m^d|\Psi_0\rangle = (2\lambda)^{10} \sum_{\{k_{ab}\} \in \mathbb{Z}^{10}} \int_0^\infty \mathrm{d}^{10}j \int \mathrm{d}\phi\, U\, \mathrm{e}^{-\lambda S_{tot}^{(k)}} (A_{an} \cdot A_{bn})(A_{cm} \cdot A_{dm}), \tag{24}$$

$$\langle W|E_n^a \cdot E_n^b|\Psi_0\rangle = (2\lambda)^{10} \sum_{\{k_{ab}\} \in \mathbb{Z}^{10}} \int_0^\infty \mathrm{d}^{10}j \int \mathrm{d}\phi\, U\, \mathrm{e}^{-\lambda S_{tot}^{(k)}} (A_{an} \cdot A_{bn}), \tag{25}$$

$$\langle W|E_m^c \cdot E_m^d|\Psi_0\rangle = (2\lambda)^{10} \sum_{\{k_{ab}\} \in \mathbb{Z}^{10}} \int_0^\infty \mathrm{d}^{10}j \int \mathrm{d}\phi\, U\, \mathrm{e}^{-\lambda S_{tot}^{(k)}} (A_{cm} \cdot A_{dm}), \tag{26}$$

with the total action $S_{tot}^{(k)}$ written as

$$S_{tot}^{(k)} = S_{tot} + 4\pi \mathrm{i} \sum_{a>b} j_{ab} k_{ab}, \tag{27}$$

$$S_{tot} = \mathrm{i} \sum_{ab} \zeta_0^{ab}(j_{ab} - j_{0ab}) + \sum_{ab,cd} \alpha^{(ab)(cd)} \frac{j_{ab} - j_{0ab}}{\sqrt{j_{0ab}}} \frac{j_{cd} - j_{0cd}}{\sqrt{j_{0cd}}} - S(j,\phi), \tag{28}$$

where S is given in (10). Recall that when $\lambda j_{ab} \in \mathbb{Z}/2$, λS is defined up to $2\pi i\mathbb{Z}$ because it contains logarithms which are multi-valued, so $e^{\lambda S}$ is single-valued. Nonetheless, when replacing the sums over j_{ab} by the integral $\int \mathrm{d}^{10}j$, $e^{\lambda S}$ becomes multi-valued since j_{ab} becomes continuous. The integrands in (23) - (26) are understood as being defined on the covering space of the logarithms in S, while the integration domain of $\int \mathrm{d}\phi$ is in the principle branch $\mathrm{Im}(\log(x)) \in (-\pi, \pi]$ of the covering space.

The critical point of $S_{tot}^{(k)}$ satisfies the following equations

$$\mathrm{Re}\,(S_{tot}) = 0, \quad \partial_{j_{ab}} S_{tot} = -4\pi \mathrm{i} k_{ab}, \quad \partial_\phi S = 0. \tag{29}$$

The real part of the second equation above implies $j_{ab} = j_{0ab}$ at the critical point (α has a positive definite real part). Then, $\mathrm{Re}\,(S_{tot}) = \mathrm{Re}\,(S) = 0$ and $\partial_\phi S = 0$ are the standard critical equations extensively studied in the asymptotics

of 4-simplex amplitude (see e.g. [15, 19]). Given the boundary data in Section II A, there are 2 solutions (up to gauge freedom) of $\mathrm{Re}\,(S) = 0$ and $\partial_\phi S = 0$ in the integration domain, corresponding to the Lorentzian 4-simplex geometry with opposite orientations. The 4-simplex geometry is consistent with the boundary data in Section II A. We denote these two solutions by (j_0, ϕ_0) and (j_0, ϕ_0').

In order that Eqs.(29) have solution. We demand the ζ_0^{ab} in Table III to satisfy the following relation

$$i\zeta_0^{ab} = \frac{\partial S\,(j, \phi)}{\partial j_{ab}}\bigg|_{j_0, \phi_0}. \tag{30}$$

Here, ζ_0^{ab} is defined modulo $4\pi\mathbb{Z}$ in $\psi_{\lambda j_0, \zeta_0}$ when $\lambda j_{ab}, \lambda j_{0ab} \in \mathbb{Z}/2$. When we replace the sums over j_{ab} by integrals over continuous j_{ab}, this $4\pi\mathbb{Z}$ gauge symmetry is broken in each integral, so we fix the values of ζ_0^{ab} as in Table III.

The ζ_0^{ab} satisfying (30) gives $\partial_{j_{ab}} S_{tot} = 0$ at one solution (j_0, ϕ_0) and $\partial_{j_{ab}} S_{tot} \neq 0$ but $\partial_{j_{ab}} S_{tot} \notin 4\pi\mathbb{Z}$ at the other solution (j_0, ϕ_0') [7]. Therefore, any $k_{ab} \neq 0$ leads to Eqs.(29) have no solution. When λ is large, the integrals in (23) - (26) are suppressed exponentially, unless all $k_{ab} = 0$. When $k_{ab} = 0$, Eqs.(29) have a unique solution (j_0, ϕ_0). which is the critical point of the integral.

Furthermore, because of the Gaussian in $\psi_{\lambda j_0, \zeta_0}$, the integrals (23) - (26) are dominant in the neighborhood where j_{ab} is close to the j_{0ab}. When λ is large, we can approximate the $\int_0^\infty$ in these integrals to $\int_{-\infty}^\infty$, while their differences are exponentially suppressed.

We rewrite (23) - (26) as below

$$\langle W|\Psi_0\rangle \simeq \int_{-\infty}^\infty \mathrm{d}^{10}j \int \mathrm{d}\phi\, \tilde{U}\, e^{-\lambda S_{tot}}, \tag{31}$$

$$\langle W|E_n^a \cdot E_n^b E_m^c \cdot E_m^d|\Psi_0\rangle \simeq \int_{-\infty}^\infty \mathrm{d}^{10}j \int \mathrm{d}\phi\, \tilde{U}\, e^{-\lambda S_{tot}}(A_{an} \cdot A_{bn})(A_{cm} \cdot A_{dm}), \tag{32}$$

$$\langle W|E_n^a \cdot E_n^b|\Psi_0\rangle \simeq \int_{-\infty}^\infty \mathrm{d}^{10}j \int \mathrm{d}\phi\, \tilde{U}\, e^{-\lambda S_{tot}}(A_{an} \cdot A_{bn}), \tag{33}$$

$$\langle W|E_m^c \cdot E_m^d|\Psi_0\rangle \simeq \int_{-\infty}^\infty \mathrm{d}^{10}j \int \mathrm{d}\phi\, \tilde{U}\, e^{-\lambda S_{tot}}(A_{cm} \cdot A_{dm}), \tag{34}$$

where $\tilde{U} \equiv (2\lambda)^{10}U$. Our numerical program computes expectation values

$$\frac{\langle W|E_n^a \cdot E_n^b E_m^c \cdot E_m^d|\Psi_0\rangle}{\langle W|\Psi_0\rangle}, \quad \frac{\langle W|E_n^a \cdot E_n^b|\Psi_0\rangle}{\langle W|\Psi_0\rangle}, \quad \frac{\langle W|E_m^c \cdot E_m^d|\Psi_0\rangle}{\langle W|\Psi_0\rangle} \tag{35}$$

by their approximate integral expressions in (31) - (34). Comparing to the original definition in (14), we have neglected 3 types contributions, $\frac{1}{2}f(0)$ in (22), $k_{ab} \neq 0$, and $\int_{-\infty}^0 \mathrm{d}j_{ab}$, that are exponentially suppressed when λ is large.

In order to apply the method of Lefschetz thimble, in (31) - (34), we extend the integration domain of $\int \mathrm{d}\phi$ on the cover space beyond the principle branch to allow $\mathrm{Im}(\log(x)) \in (-\infty, \infty)$, so that $\int \mathrm{d}\phi$ is along the integration cycle connecting to the infinity of the cover space. (31) - (34) have no critical point beyond the principle branch (by fixing the values of ζ_0^{ab}). Therefore, extending the integrals only add contributions that are exponentially suppressed at large λ.

E. Critical Point

Finding the critical point of the total action S_{tot} is important for both the asymptotic expansion method and our Lefschetz thimble method. The action S_{tot} has only one critical point corresponding to the aforementioned 4-simplex geometry. As the solution to the critical equations $\mathrm{Re}(S_{tot}) = 0$, $\partial_\phi S_{tot} = 0$, and $\partial_j S_{tot} = 0$, the critical values of the group elements g_{0a}, $(a = 2 \cdots 5)$ are the spinor representations of the Lorentz transformations converting $N_2 \cdots N_5$ in (3) to $(1, 0, 0, 0)$ [8]. TABLE V lists the explicit values of g_{0a}, $(a = 2 \cdots 5)$.

---

[7] The spinfoam action $S = i\sum_{a>b} j_{ab}(\pm\gamma\Theta_{ab} + 2\theta_{ab})$ at 2 solution [15]. Θ_{ab} are the dihedral angles of the 4-simplex, and $\theta_{ab} = \psi_{ab} - \psi_{ba}$ give the overall phase of the asymptotics. When we set $\zeta_0^{ab} = \gamma\Theta_{ab} + 2\theta_{ab}$, we have $\partial_{j_{ab}} S_{tot} = 0$ at one solution and $\partial_{j_{ab}} S_{tot} = 2\gamma\Theta_{ab} \notin 4\pi\mathbb{Z}$ at the other solution.

[8] More details are found [26, 55]

TABLE V. Each cell of the table is the critical point of a-th group element.

a	1	2	3	4	5
g_{0a}	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0.18i & 1.01i \\ 1.01i & 0.18i \end{pmatrix}$	$\begin{pmatrix} 0.18i & 0.96-0.34i \\ -0.96-0.34i & 0.18i \end{pmatrix}$	$\begin{pmatrix} 1.01i & -0.48-0.34i \\ 0.48-0.34i & -0.65i \end{pmatrix}$	$\begin{pmatrix} -0.65i & -0.48-0.34i \\ 0.48-0.34i & 1.01i \end{pmatrix}$

Having had the critical values of the group elements, we can plug them into the critical equations $\mathrm{Re}(S_{tot}) = 0$, which is equivalent to

$$|\xi_{ab}\rangle = \frac{e^{i\psi_{ab}}}{\|Z_{ab}\|} g_a^\dagger |z_{ab}\rangle, \quad \text{and} \quad |J\xi_{ba}\rangle = \frac{e^{i\psi_{ba}}}{\|Z_{ba}\|} g_b^\dagger |z_{ab}\rangle, \tag{36}$$

with $\|Z_{ab}\| = |\langle Z_{ab}, Z_{ab}\rangle|^{1/2}$, in order to determine the spinors z_{ab} and the phase factors ψ_{ba}, ψ_{ab}. The normalized components of the spinors z_{ab} are recorded in TABLE VI.

TABLE VI. Each cell indicates a spinor z_{0ab}.

$|z_{0ab}\rangle$ \ b a	1	2	3	4	5
1	＼	(1,1)	(1,-0.333+0.942i)	(1,-0.184-0.259i)	(1,-1.817-2.569 i)
2	(1,1)	＼	(1,0.685-0.729i)	(1, 1.857 + 0.989 i)	(1, 0.420 + 0.223 i)
3	(1, 0.333 - 0.943 i)	(1, 0.685 - 0.729 i)	＼	(1, 0.313 + 2.080 i)	(1, 0.071 + 0.470 i)
4	(1, -0.184-0.259i)	(1, 1.857 + 0.989 i)	(1,0.313 + 2.080 i)	＼	(1, 0.058+0.082i)
5	(1, -1.817-2.569 i)	(1, 0.420 + 0.223 i)	(1,0.071 + 0.470 i)	(1, 0.058+0.082i)	＼

Based on the critical g_{0ab} and z_{0ab}, we apply our parameterizations (11) and (12) but let them centered at the critical point:

$$g = g_0 \begin{pmatrix} 1 + \frac{x_1+iy_1}{\sqrt{2}} & \frac{x_2+iy_2}{\sqrt{2}} \\ \frac{x_3+iy_3}{\sqrt{2}} & \frac{1+\frac{x_2+iy_2}{\sqrt{2}}\frac{x_3+iy_3}{\sqrt{2}}}{1+\frac{x_1+iy_1}{\sqrt{2}}} \end{pmatrix}, \quad \text{and} \quad z = (1, z_0 + x + iy), \tag{37}$$

where g_0 stands for the critical value of the group variable in TABLE V and z_0 is the second component of the spinor variable in TABLE VI. When all the parameters are zero, the group variables and spinors take their critical values.

To conclude, here comes 2 main points of this section: Firstly, the total action S_{tot} in (27) depends on 54 real parameters, and specifically we parametrize the four group variables g_a and the ten spinor variables z_{ab} as (37). Secondly, the spinfoam propagator G_{mn}^{abcd} can be computed by the integrals (31) - (34). In the next section, we will show the algorithm to evaluate these integrals.

III. THEORY AND ALGORITHM

The total action S_{tot} is a complex valued, such that the integrands in (31) - (34) are highly oscillatory, especially when λ is large. This fact plagues the attempts of using the conventional Monte-Carlo method to compute the spinfoam propagator. In this section, we review how to use *Picard-Lefschetz theory* to transform these types of integrals to be non-oscillatory (see e.g. [37, 39]) for reviews), and we present the algorithm that combines the thimble and Markov-Chain Monte Carlo (MCMC) methods and can compute the expectation value of an observable when the action is complex valued.

A. Lefschetz Thimble

The thimble method is a high-dimensional generalization the elementary saddle point integration along the steepest decent (SD). The thimble method can be further generalized as follows.

The starting point of computing the integral

$$A = \int d^n x f(\vec{x}) e^{-S(\vec{x})}, \tag{38}$$

is to analytically continue both $f(\vec{x})$ and $S(\vec{x})$ to be holomorphic functions $\hat{f}(\vec{z})$ and $\hat{S}(\vec{z})$, such that (38) becomes an integral of analytic functions $\hat{f}(\vec{z})$ and $\hat{S}(\vec{z})$ of complex variables on the real domain

$$A = \int_{\mathbb{R}^n} \mathrm{d}^n z \hat{f}(\vec{z}) \mathrm{e}^{-\hat{S}(\vec{z})}, \tag{39}$$

where $\mathrm{d}^n z \hat{f}(\vec{z}) \mathrm{e}^{-\hat{S}(\vec{z})}$ is a holomorphic n-form restricted on $\mathbb{R}^n$.

The Picard-Lefschetz theory shows that the integral A can be decomposed into a linear combination of integrals over real n-dimensional integral cycles $\mathcal{J}_\sigma$, $\sigma = 1, 2, 3, \cdots$

$$\int_{\mathbb{R}^n} \mathrm{d}^n z \hat{f}(\vec{z}) \mathrm{e}^{-\hat{S}(\vec{z})} = \sum_\sigma n_\sigma \int_{\mathcal{J}_\sigma} \mathrm{d}^n z \hat{f}(\vec{z}) \mathrm{e}^{-\hat{S}(\vec{z})}, \tag{40}$$

when $\mathbb{R}^n$ is homologically equivalent to $\sum_\sigma n_\sigma \mathcal{J}_\sigma$. The holomorphic n-form $\mathrm{d}^n z \hat{f}(\vec{z}) \mathrm{e}^{-\hat{S}(\vec{z})}$ is restricted in a class of $\mathcal{J}_\sigma$ on the right hand side. This decomposition is given by the n-dimensional real sub-manifolds $\mathcal{J}_\sigma$, each attached to a critical point p_σ satisfying $\partial_z \hat{S}(p_\sigma) = 0$ [9]. Each $\mathcal{J}_\sigma$, called a Lefschetz thimble, is a union of SD paths that are solutions to the SD equations

$$\frac{\mathrm{d}z^a}{\mathrm{d}t} = -\frac{\overline{\partial \hat{S}(\vec{z})}}{\partial \overline{z^a}}, \tag{41}$$

and falls to the critical point p_σ when $t \to \infty$. Here we call t the flow time.

Since

$$\frac{\mathrm{d}\hat{S}}{\mathrm{d}t} = \frac{\partial \hat{S}}{\partial z^a} \frac{\mathrm{d}z^a}{\mathrm{d}t} = -\left|\frac{\partial \hat{S}}{\partial z^a}\right|^2, \tag{42}$$

$\mathrm{Re}(\hat{S})$ monotonically decreases along each SD path and approaches its minimum at the critical point, while $\mathrm{Im}(\hat{S})$ is conserved along each path. Thus, on each thimble $\mathcal{J}_\sigma$,

$$\int_{\mathcal{J}_\sigma} \mathrm{d}^n z \hat{f}(\vec{z}) \mathrm{e}^{-\hat{S}(\vec{z})} = \mathrm{e}^{-\mathrm{i}\,\mathrm{Im}(\hat{S}(p_\sigma))} \int_{\mathcal{J}_\sigma} \mathrm{d}^n z \hat{f}(\vec{z}) \mathrm{e}^{-\mathrm{Re}(\hat{S}(\vec{z}))} \tag{43}$$

becomes a non-oscillatory integral times a constant phase $\mathrm{e}^{-\mathrm{i}\,\mathrm{Im}(\hat{S}(p_\sigma))}$. On $\mathcal{J}_\sigma$, $\mathrm{Re}(\hat{S})$ grows when moving far away from the critical point, so the integrand is exponentially suppressed at the infinity, and the integral on $\mathcal{J}_\sigma$ is convergent.

The Lefschetz-Thimbles $\{\mathcal{J}_\sigma\}$ presents a good basis of relative homology group for the integral (43) [60]. Using this basis, the integral (40) is valid for a specific set $\{n_\sigma\}$ of the weights of the thimbles. Consider $\hat{f}$ as an observable. The expectation value $\langle f \rangle$ is given by

$$\langle f \rangle = \frac{\int_{\mathbb{R}^n} \mathrm{d}^n z \hat{f}(\vec{z}) \mathrm{e}^{-\hat{S}(\vec{z})}}{\int_{\mathbb{R}^n} \mathrm{d}^n z \mathrm{e}^{-\hat{S}(\vec{z})}} = \frac{\sum_\sigma n_\sigma \int_{\mathcal{J}_\sigma} \mathrm{d}^n z \hat{f}(\vec{z}) \mathrm{e}^{-\hat{S}(\vec{z})}}{\sum_\sigma n_\sigma \int_{\mathcal{J}_\sigma} \mathrm{d}^n z \mathrm{e}^{-\hat{S}(\vec{z})}}. \tag{44}$$

A weight n_σ is the intersection number between the original integration cycle $\mathbb{R}^n$ and the manifold of the steepest ascent (SA) paths approaching the critical point p_σ as $t \to \infty$. The SA paths are solutions to the SA equations

$$\frac{\mathrm{d}z^a}{\mathrm{d}t} = \frac{\overline{\partial \hat{S}(\vec{z})}}{\partial \overline{z^a}}. \tag{45}$$

Along each SA path, $\mathrm{Re}(\hat{S})$ monotonically increases and approach the maximum at the critical point, while $\mathrm{Re}(\hat{S})$ is conserved along the path. Computing these weights are challenging in general (see e.g. [61, 62] for some recent progresses). Nevertheless, in the cases where one of the thimble, denoted by $\mathcal{J}_{\sigma'}$, dominates the integral, we may neglect the contribution of other thimbles and re-express (44) as

$$\langle f \rangle \simeq \frac{n_{\sigma'} \mathrm{e}^{-\mathrm{i}\,\mathrm{Im}(S(p_{\sigma'}))} \int_{\mathcal{J}_{\sigma'}} \mathrm{d}^n z \, \hat{f}(\vec{z}) \mathrm{e}^{-\mathrm{Re}(\hat{S}(\vec{z}))}}{n_{\sigma'} \mathrm{e}^{-\mathrm{i}\,\mathrm{Im}(S(p_{\sigma'}))} \int_{\mathcal{J}_{\sigma'}} \mathrm{d}^n z \, \mathrm{e}^{-\mathrm{Re}(\hat{S}(\vec{z}))}} = \frac{\int_{\mathcal{J}_{\sigma'}} \mathrm{d}^n z \, \hat{f}(\vec{z}) \mathrm{e}^{-\mathrm{Re}(\hat{S}(\vec{z}))}}{\int_{\mathcal{J}_{\sigma'}} \mathrm{d}^n z \, \mathrm{e}^{-\mathrm{Re}(\hat{S}(\vec{z}))}}, \tag{46}$$

which can be considered as a mean value provided by a sampling on the thimble $\mathcal{J}_{\sigma'}$ with a Boltzmann factor $\mathrm{e}^{-\mathrm{Re}(\hat{S}(\vec{z}))}$. Then, it is possible to use the MCMC method to numerically compute $\langle f \rangle$. Each integral involved in the spinfoam propagator has a single critical point in its integration domain. The Lefschetz thimble of the critical point is dominant. Thus the computation of the spinfoam propagator is the case where Eq. (46) applies.

---

[9] We do not impose $\mathrm{Re}(\hat{S}) = 0$ for critical points in the complex space.

B. Thimbles Generated by Flows

As the first step to apply the the Lefschetz thimble framework to numerics, we need to find the Lefschetz thimble $\mathcal{J}_\sigma$ for a given critical point p_σ (FIG. 1 (a)). By definition, one might try to decide if a point is on the thimble by checking if it falls to p_σ after flowing infinitely long time described by the SD equation; however, it is hard in practice due to the infinite flow time. Naively, we might also use p_σ as the initial point and using SA equation to generate the thimble as the union of the paths going away from the p_σ, but this way is also problematic because p_σ is a fixed point of the SA equation.

We follow the method reviewed in [39] to bypass the difficulty of generating $\mathcal{J}_\sigma$ numerically. We consider a small real n-dimensional neighborhood V_σ of the critical point p_σ and a slightly different integral cycle denoted by $\widehat{\mathcal{J}}_\sigma$. $\widehat{\mathcal{J}}_\sigma$ is the union of solutions to the SD equations (41) flowing to V_σ after infinite time evolution. $\widehat{\mathcal{J}}_\sigma$ is also real n-dimensional. $\widehat{\mathcal{J}}_\sigma$ is a good approximation of the true thimble $\mathcal{J}_\sigma$ when the size of V_σ is small, as all the SD paths on $\widehat{\mathcal{J}}_\sigma$ connect with V_σ. $\widehat{\mathcal{J}}_\sigma$ (FIG. 1 (b)) approaches $\mathcal{J}_\sigma$ when V_σ shrinks to the critical point p_σ. Since the integrand is analytic, and $\widehat{\mathcal{J}}_\sigma$ is a deformation of $\mathcal{J}_\sigma$, the integral

$$\int_{\widehat{\mathcal{J}}_\sigma} d^n z \hat{f}(\vec{z}) e^{-\hat{S}(\vec{z})} \tag{47}$$

is the same as (43); however, since $\mathrm{Im}(\hat{S})$ is no longer constant in $\widehat{\mathcal{J}}_\sigma$, the above integral becomes oscillatory in contrast to the integral on $\mathcal{J}_\sigma$. If V_σ is small enough, however, the fluctuation of the $\mathrm{Im}(\hat{S})$ on $\widehat{\mathcal{J}}_\sigma$ is so small that the oscillation of the integral is weak enough to keep the Monte Carlo method accurate.

Since infinite time evolution is involved, finding the entire $\widehat{\mathcal{J}}_\sigma$ is not numerically practical. A practical integral cycle $\tilde{\mathcal{J}}_\sigma$ is the union of the solutions to the SD equations (41) falling to V_σ after finite but sufficiently long flow time. The thimble $\tilde{\mathcal{J}}_\sigma$ approaches $\widehat{\mathcal{J}}_\sigma$ when the flow time is infinite. Similar to the method in [61], we can find the approximate $\tilde{\mathcal{J}}_\sigma$ in an inverse process. Namely, we choose a small real n-dimensional neighborhood V_σ of the critical point, and flow upward from points in V_σ according to the SA equation with a finite time T. The end points of these flows form a real n-dimensional manifold $\tilde{\mathcal{J}}_\sigma$ (FIG. 1). The thimble $\tilde{\mathcal{J}}_\sigma$ does not reach the infinity of the Lefschetz thimble $\mathcal{J}_\sigma$. The size of $\tilde{\mathcal{J}}_\sigma$ depends on the choice of T.

Our two-step approximation of $\mathcal{J}_\sigma$ is illustrated in the following diagram:

$$\mathcal{J}_\sigma \xrightarrow{\text{Fix } V_\sigma} \widehat{\mathcal{J}}_\sigma \xrightarrow{\text{Fix } T} \tilde{\mathcal{J}}_\sigma.$$

In the first step, we use the $\widehat{\mathcal{J}}_\sigma$ as the union of all the steepest decent paths falling to V_σ to approximate $\mathcal{J}_\sigma$. In our computation, we set the size of V_σ by setting a tolerance of the fluctuation of the $\mathrm{Im}(\hat{S})$ on $\widehat{\mathcal{J}}_\sigma$. In the second step, we use $\tilde{\mathcal{J}}_\sigma$ as the union of the finitely evolved steepest ascent paths starting from the points in V_σ to approximate $\widehat{\mathcal{J}}_\sigma$. Thus, the longer T and smaller V_σ are, the better approximation of $\mathcal{J}_\sigma$ is achieved as $\widehat{\mathcal{J}}_\sigma$.

Another remark is that in the second step of the approximation, making $\tilde{\mathcal{J}}_\sigma$ very large is actually unnecessary. When computing (47), we sample the points on the thimble with the probability distribution $e^{-\mathrm{Re}(S)}$, and the contributions to the integral from points far away from the critical point are exponentially suppressed. Thus, we can choose the T parameter, which provides sufficiently large $\tilde{\mathcal{J}}_\sigma$ containing points that contribute dominantly to (47), while ensuring increasing T only add negligible contribution to the integral. The result should converge when further increasing T. In fact, experiences from our computation and other existing results [39, 42, 63] suggest even $T < 1$ suffice to result in good accuracy.

The choice of real n-dimensional V_σ depends on the local behavior of the SA equations (45) around the critical point p_σ. Consider a small holomorphic variation $\omega^k = \delta z^k$, we linearize (45):

$$\frac{d\omega^k}{dt} = \overline{\frac{\partial^2 \hat{S}}{\partial z_k \partial z_l}} \cdot \overline{\omega^l}, \tag{48}$$

In the neighborhood of p_σ, $\overline{\frac{\partial^2 \hat{S}}{\partial z_k \partial z_l}}$ can be approximated by the Hessian $\bar{\mathbf{H}}$ of $\hat{S}(\vec{z})$ at p_σ, and solution of (48) is given by

$$\omega = \sum_{a=1}^{2n} e^{\lambda^a t} \omega_a,$$

where λ^a and ω_a are the eigenvalues and the corresponding eigenvectors of the generalized eigenvalue equation:
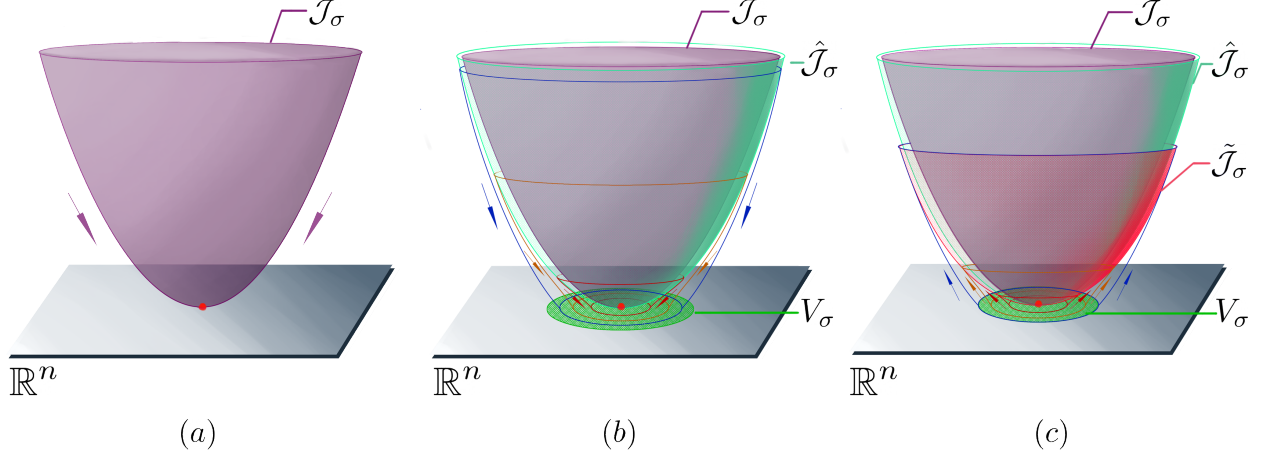
$$\mathbf{H}\omega = \lambda \bar{\omega}. \tag{49}$$

FIG. 1. (a) A Lefschetz thimble $\mathcal{J}_\sigma$ (purple surface) is the union of all the SD paths falling to the critical point p_σ (red dot) when $t \to \infty$.
(b) $\hat{\mathcal{J}}_\sigma$ (green transparent surface) is defined as the union of points that can flow to V_σ (green disk at the bottom) after an infinite time evolution by the SD equation. For example, the cross-sections of $\hat{\mathcal{J}}_\sigma$ (illustrated by the blue, yellow and red circles in $\hat{\mathcal{J}}_\sigma$) flow to the cross-sections in V_σ (blue, yellow and red circles in the green disk).
(c) $\tilde{\mathcal{J}}_\sigma$ (red transparent surface) is generated by upward flowing every points in V_σ with a finite time evolution by the SA equation. The cross-sections in V_σ (blue, yellow and red circles in the green disk) flow upward to the cross-sections in $\tilde{\mathcal{J}}_\sigma$ (blue, yellow and red circles in $\tilde{\mathcal{J}}_\sigma$).

Using the Takagi factorization [64], we covert (49) to a real $2n$-dimensional eigenvalue equation:

$$
\begin{bmatrix} \mathbf{H}_\mathbb{R} & \mathbf{H}_\mathbb{I} \\ \mathbf{H}_\mathbb{I} & -\mathbf{H}_\mathbb{R} \end{bmatrix} \begin{bmatrix} \omega_\mathbb{R} \\ \omega_\mathbb{I} \end{bmatrix} = \lambda \begin{bmatrix} \omega_\mathbb{R} \\ \omega_\mathbb{I} \end{bmatrix},
\tag{50}
$$

where $\mathbf{H}_\mathbb{R}$ and $\mathbf{H}_\mathbb{I}$ are the real and imaginary parts of the Hessian $\mathbf{H}$. The eigenvalues λ in (50), which are equivalent to λ in (49), come in pairs $\{\pm\lambda_i\}$. The eigenvectors $(\omega_{i\mathbb{R}}, \omega_{i\mathbb{I}})$, called the Takagi vectors, can reconstruct the eigenvectors $\{\omega_i\}$ by

$$
\omega_i = \omega_{i\mathbb{R}} + \mathrm{i}\omega_{i\mathbb{I}}.
$$

The flow given by (45) is repulsive along the eigenvectors ω_a with positive eigenvalues, and is attractive along the eigenvectors ω_a with negative eigenvalues. The paths along the attractive directions converge to the critical point p_σ, so they are not the paths that can form $\tilde{\mathcal{J}}_\sigma$. Only the paths flowing along the repulsive directions can form the $\tilde{\mathcal{J}}_\sigma$. We denote $\hat{\omega}_a$ as the normalized eigenvectors with positive eigenvalues. The space $\hat{V}_\sigma$ understood as a local neighborhood in the $\vec{z}$-coordinate chart at p_σ is expressed as

$$
\hat{V}_\sigma = \{\vec{z}|\vec{z} = \sum_{a=1}^{n} \hat{\omega}_i x^i + \vec{z}_\sigma, \text{ each } x^i \in \mathbb{R} \text{ is small}\}.
\tag{51}
$$

where $\vec{z}_\sigma$ are coordinates of p_σ. $\hat{V}_\sigma$ turns out to be the best choice of V_σ.

By $\delta z^k = (\partial z^k / \partial x^i)\delta x^i$ for the coordinates $\{x^i\}$ on $\hat{V}_\sigma$, and assuming $\mathrm{d}\delta x^i / \mathrm{d}t = 0$, $J_i^k \equiv \partial z^k / \partial x^i$ satisfies the same equation as (48), i.e.

$$
\frac{\mathrm{d}(J_i^k)_t}{\mathrm{d}t} = \sum_{l=1}^{n} \overline{\frac{\partial^2 \hat{S}}{\partial z_k \partial z_l}} \overline{(J_i^l)_t}.
\tag{52}
$$

The solution J_t is the Jacobian matrix of a flow of coordinate changes from $\{x^i\}$ to $\{z^i\}$. The initial condition J_0 is the constant $n \times n$-matrix, whose columns are the vectors $\hat{\omega}_a$. In what follows, $J := J_T$ is the Jacobian for changing from $\{x^i\}$ to $\{z^i\}$ on $\tilde{\mathcal{J}}_\sigma$.

By the coordinate change, for any holomorphic function $\psi(z)$, its integral on $\tilde{\mathcal{J}}_\sigma$ can be expressed by the integral of $\{x^i\}$ in $\hat{V}_\sigma$

$$\int_{\tilde{\mathcal{J}}_\sigma} \mathrm{d}^n z \, \psi(z) = \int_{\hat{V}_\sigma} \mathrm{d}^n x \, \det(J(x)) \, \psi(z(x)). \tag{53}$$

In the active point of view, for a fixed flow time T, every point in $\hat{V}_\sigma$ flows upward to $\tilde{\mathcal{J}}_\sigma$ according to the SA equation (45). We define the local diffeomorphism

$$\mathcal{C}_T : \hat{V}_\sigma \to \tilde{\mathcal{J}}_\sigma,$$

that can map the initial point $p \in \hat{V}_\sigma$ to the end point $\mathcal{C}_T(p) \in \tilde{\mathcal{J}}_\sigma$ of the SA path with the finite evolution time T. The coordinate change from $\{x^i\}$ to $\{z^i\}$ is induced by $\mathcal{C}_T$.

As a result, for any given observable f, its expectation value can be computed by

$$\langle f \rangle \simeq \frac{\int_{\tilde{\mathcal{J}}_\sigma} \mathrm{d}^n z \, \hat{f}(z) \, \mathrm{e}^{-\hat{S}(z)}}{\int_{\tilde{\mathcal{J}}_\sigma} \mathrm{d}^n z \, \mathrm{e}^{-\hat{S}(z)}} = \frac{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \det(J(x)) \, \hat{f}\left(\mathcal{C}_T(x)\right) \, \mathrm{e}^{-\hat{S}(\mathcal{C}_T(x))}}{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \det(J(x)) \, \mathrm{e}^{-\hat{S}(\mathcal{C}_T(x))}}$$
$$= \frac{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \mathrm{e}^{\mathrm{i}(\arg(\det(J)) - \mathrm{Im}(\hat{S}))} \, \hat{f} \, \mathrm{e}^{-\mathrm{Re}(\hat{S}) + \log(|\det(J)|)}}{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \mathrm{e}^{\mathrm{i}(\arg(\det(J)) - \mathrm{Im}(\hat{S}))} \, \mathrm{e}^{-\mathrm{Re}(\hat{S}) + \log(|\det(J)|)}}, \tag{54}$$

where in the second step we apply (53). Note that $\det(J)$ is a complex number, and that $\log(\det(J))$ is given by $\log(|\det(J)|) + \mathrm{i} \arg(\det(J))$.

We define $\mathrm{Re}(\hat{S}) - \log(\det(J)) \equiv S_{eff}$ as the purely real effective action. For any observable $\mathcal{O}$, we define its expectation with respect to the effective action as

$$\langle \mathcal{O} \rangle_{eff} = \frac{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \mathcal{O} \, \mathrm{e}^{-S_{eff}}}{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \mathrm{e}^{-S_{eff}}}. \tag{55}$$

We define $\arg(\det(J)) - \mathrm{Im}(\hat{S})$ as the residual phase θ_{res}, and rewrite (54) as

$$\langle f \rangle \simeq \frac{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \hat{f} \, \mathrm{e}^{\mathrm{i}\theta_{res}} \, \mathrm{e}^{-S_{eff}}}{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \mathrm{e}^{-S_{eff}}} \times \frac{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \mathrm{e}^{-S_{eff}}}{\int_{\hat{V}_\sigma} \mathrm{d}^n x \, \mathrm{e}^{\mathrm{i}\theta_{res}} \, \mathrm{e}^{-S_{eff}}} = \frac{\langle \mathrm{e}^{\mathrm{i}\theta_{res}} \hat{f} \rangle_{eff}}{\langle \mathrm{e}^{\mathrm{i}\theta_{res}} \rangle_{eff}}. \tag{56}$$

Note that $\mathrm{Im}(\hat{S}(\vec{z}))$ is not a constant in $\hat{V}_\sigma$. The fluctuation of $\mathrm{Im}(\hat{S}(\vec{z}))$ tends to vanish when shrinking $\hat{V}_\sigma$. In our computation, we set a maximal tolerance $\mathcal{E}$ of the fluctuation of $\mathrm{Im}(\hat{S}(\vec{z}))$. The tolerance determines the size of $\hat{V}_\sigma$, such that at any point $p \in \hat{V}_\sigma$, $|\mathrm{Im}(\hat{S}(p)) - \mathrm{Im}(\hat{S}(p_\sigma))| \leq \mathcal{E}$. Similarly, $\arg(\det(J))$ does not have strong fluctuation in our case of the spinfoam expectation values. When the integrands are weakly oscillatory functions, both $\langle \mathrm{e}^{\mathrm{i}\theta_{res}} \hat{f} \rangle_{eff}$ and $\langle \mathrm{e}^{\mathrm{i}\theta_{res}} \rangle_{eff}$ can be accurately computed by the MCMC method.

C. Differential Evolution Adaptive Metropolis (DREAM) Algorithm

By now, we have converted the problem of computing $\langle f \rangle$ to the problem of how to sample on the tangent space $\hat{V}_{\sigma'}$ with a Boltzmann factor $\mathrm{e}^{-S_{eff}}$ to compute $\langle \mathrm{e}^{\mathrm{i}\theta_{res}} \rangle_{eff}$ and $\langle \hat{f} \mathrm{e}^{\mathrm{i}\theta_{res}} \rangle_{eff}$. The latter problem can be numerically solved by Markov Chain Monte Carlo (MCMC) method [43].

The MCMC is a class of algorithms designed for sampling from a posterior probability distribution $\mathrm{e}^{-S_{eff}}$. Each Markov chain can be regarded as generated by a random 'walker' moving in the integration domain. In general, in each step of the 'walker', MCMC methods use the accept/reject scheme to adjust the transition distribution that dictates the orientation and the length of the step, such that the points sampled by the 'walker' converge to a desired posterior distribution after a 'long march'.

Since in our computation of the integrals (31) - (34), $\mathrm{e}^{-S_{eff}}$ is high-dimensional and complicated, we choose a MCMC method—the Differential Evolution Adaptive Metropolis (DREAM)[46, 65]. The DREAM algorithm has not been combined with the Lefschetz thimble method in the literature.

The DREAM algorithm runs multiple Markov chains in parallel. This algorithm is able to sample in different regions of the integration domain simultaneously. In case that S_{eff} has some local minima other than the critical point (global minimum), unlike the one-chain scheme, this multi-chain scheme prevents a sampling procedure from

being trapped in the neighborhood of any local minimum [46]. Besides, this multi-chain scheme is more adaptable to the architecture of high performance computers designed for multitasking.

The performance of a MCMC method also depends on the quality of the candidates of each update provided by the method. On the one hand, at each update, if a candidate is too far from the current location of the chain, the candidate is rejected and wastes the computation resource. On the other hand, if a candidate is too close to the current location of the chain, although it is highly probable to be accepted, the Markov chain may take numerous updates to explore the whole integration domain. In order to balance the progress of the chain in each step and a reasonable acceptance rate, the DREAM algorithm possesses the following features

- At each update, the DREAM uses a genetic algorithm to provide a candidate for each chain, based on the current location of other chains.

- Similar to the Gibbs sampling, the DREAM does not update all the components in one update. Instead, it implements a randomized subspace sampling strategy. For each update of each chain, the algorithm does not update all the components of the sample. There is a probability, called the crossover ratio CR, to decide whether a component needs to be updated or not.

- The algorithm suggests to do test runs, called burn-in runs, before the formal sampling. In the burn-in runs, the crossover ratio CR and the parameters used in the genetic algorithm are adapted. Furthermore, in the burn-in runs, the unwanted chains can be removed by using inter-quartile range method.

Explicitly, after a proper burn-in run, we can compute the $\langle f \rangle$ in the following steps.

1. Compute the Hessian of the action $\hat{S}(\vec{z})$ at the critical point p_σ. Use (50) to compute the basis vectors $\{\hat{\omega}_i\}$ of the thimble's tangent space $\hat{V}_\sigma$. As claimed before, the action $\hat{S}(\vec{z})$ is defined in $\mathbb{C}^n$, and $\hat{V}_\sigma$ is real n-dimensional.

2. Choose M points in $\mathbb{R}^n$ close to the critical point [10], and denote them as $\mathbf{x}_s^{(0)}$, $(s = 1, \cdots, M)$, where the index (0) indicates the initial step. Then, $\mathbf{x}_s^{(0)} \cdot \hat{\omega}$, $(s = 1, \cdots, M)$ are the initial points of the M Markov chains on $\hat{V}_\sigma$ (some more details of our choice of the initial points are shown in Section V).

3. Generate CR from a given multi-nomial distribution constructed in a burn-in run. Construct a n-element series $\{u^i\}$, where each u^i is drawn from a uniform distribution $U(0,1)$. Then, construct a n-dimensional vector $\mathbf{v}$ whose components are given by

$$v^i = \begin{cases} 0 & \text{If } u^i > CR; \\ 1 & \text{otherwise.} \end{cases}$$

4. Create x_s^{cand} of each chain by

$$\mathbf{x}_s^{cand} = \mathbf{x}_s^{(t-1)} + \mathbf{v} \cdot (\mathbf{1}_n + \mathbf{e})\beta(\delta, d')(\sum_{j=1}^{\delta} \mathbf{x}_{R_1(j)}^{(t-1)} - \sum_{d=1}^{\delta} \mathbf{x}_{R_2(d)}^{(t-1)}) + \epsilon, \ s = 1, \cdots M, \tag{57}$$

where d' is the total number of non-zero components in $\mathbf{v}$, δ is the number of the pairs used to generate the candidates, and $R_1(j)$, $R_2(d) \in \{1, \cdots, M\}$, with $R_1(j) \neq R_2(d) \neq s$ for $j = 1, \cdots \delta$ and $d = 1, \cdots, \delta$. The values of ϵ and $\mathbf{e}$ are drawn from the Gaussian distribution $N(0, b^*)$ and uniform distribution $\text{Uniform}(-b, b)$ with $|b| < 1$ respectively. The parameter b^* is chosen to be very small compared with the width of the target posterior distribution. The scaling factor is $\beta(\delta, d')$. At every fifth generation, we set $\beta = 1$. Denote the candidate update for the s-th chain, i.e., $\mathbf{x}_s^{cand} \cdot \hat{\omega}$, as $\hat{\mathbf{x}}_s^{cand}$.

5. Compute $\tilde{\mathbf{x}}_s^{cand} = \mathcal{C}_T(\hat{\mathbf{x}}_s^{cand})$ as the candidate sample on the thimble, and compute the corresponding Jacobian by solving (48). Then, use the Jacobian and $\tilde{\mathbf{x}}_s^{cand}$ to compute $S_{eff}(\hat{\mathbf{x}}_s^{cand})$.

6. For each chain, construct an acceptance rate as

$$r_s^{(t)} = \min\left(1, \frac{S_{eff}(\hat{\mathbf{x}}_s^{cand})}{S_{eff}(\hat{\mathbf{x}}_s^{(t-1)})}\right).$$

Nevertheless, The following two exceptions where we let the rate becomes 0 exist:

---

[10] As suggested in [46], M should be greater than or equal to n

- when $|\text{Im}(\hat{S}(\hat{\mathbf{x}}_s^{cand})) - \text{Im}(\hat{S}(\hat{\mathbf{x}}_s^0))| > \mathcal{E}$, where $\mathcal{E}$ is the preset tolerance of the fluctuation of $\text{Im}(\hat{S})$;
- when numerically $\mathcal{C}_T(\hat{\mathbf{x}}_s^{cand})$ cannot be accurately computed.

The second exception is due to that the SA equations become unstable when it flows to the region far from the critical point; however, such far points contribute very little to the final result due to the exponential suppression $e^{\text{Re}(\hat{S})}$. Hence, we ignore the contributions of such far points.

7. Create a number $a \in [0,1]$ from uniform distribution, and then use it to decide whether a candidate can be accepted or not:

$$x_s^{(t)} = \begin{cases} x_s^{cand} & \text{If } a < r_s^{(t)}, \\ x_s^{(t-1)} & otherwise. \end{cases}$$

8. Return to step 3 until a sufficiently large number of samples are collected.

9. Use the collected samples to compute

$$\langle f \rangle = \frac{\langle e^{i\theta_{res}} \hat{f} \rangle_{eff}}{\langle e^{i\theta_{res}} \rangle_{eff}}.$$

When we have a large number of samples, $\langle \mathcal{O} \rangle_{eff}$ is equivalent to arithmetic mean among the samples

$$\langle \mathcal{O} \rangle_{eff} = \sum_{\text{samples}} \mathcal{O}(\text{sample}). \tag{58}$$

since the above procedure has produced the desired probability distribution $e^{-S_{eff}}$ for the samples.

We provide a more detailed discussion of the MCMC methods in **Appendix** A. In order to perform better, several optimizations exist, such as choosing a better the flow-time T, effectively solving the SA equations to get $\mathcal{C}_T$, and properly tuning the multi-nomial distribution of CR and the scale factor β during the burn-in runs. We provide the details of these optimizations in Section V.

IV. SPINFOAM ON LEFSCHETZ THIMBLE

In the above section, we have described the general algorithm of integrals on Lefschetz thimbles. In this section, we apply the Lefschetz thimble to the spinfoam model.

We need to first complexify the spinfoam variables j_{ab}, g_a, z_{ab} and analytically continue the integrands in (31) - (34). The analytic continuation makes $g_a^\dagger$ and $\langle z_{ab}|$ independent of g_a and $|z_{ab}\rangle$. Equivalently, using the parametrizations (11) and (12), we complexify $\vec{x}$ and $\vec{y}$ and analytically continue the integrands to holomorphic functions of $\vec{x}$ and $\vec{y}$. The spin variables j_{ab} are also complexified, and the integrands are holomorphic in j_{ab}. The real scaling parameter λ is kept real. The analytical continuation render the integrands and in particular, the action denoted by $\tilde{S}_{tot}$, holomorphic functions of 54 complex variables. The thimble is a real 54-dimensional sub-manifold in the space of complexified spinfoam variables. Note that the analytic continuation of the spinfoam integrand has been discussed in [18] (see also [66]).

After analytical continuation, $\tilde{S}_{tot}$ may have more critical points than S_{tot} does. Complex critical points may exist in addition to the real critical points discussed above. The complex critical points are away from the real integration domain, where the spinfoam integrals (31) - (34) are defined. These spinfoam integrals admit decompositions as in (40), where σ's may contain both real and complex critical points. A complex critical point $p_{\tilde{\sigma}}$ contributes to the integrals if $n_{\tilde{\sigma}} \neq 0$, i.e. there exist SA paths approaching $p_{\tilde{\sigma}}$ from the space of real variables. Thus, $n_{\tilde{\sigma}} \neq 0$ implies that $\text{Re}(\tilde{S}_{tot}(p_{\tilde{\sigma}})) > 0$ because on the space of real variables, $\text{Re}(\tilde{S}_{tot}) = \text{Re}(S_{tot}) \geq 0$. Strictly positive $\text{Re}(\tilde{S}_{tot}(p_{\tilde{\sigma}}))$ implies that when the spinfoam integrals are decomposed as in Eq. (40), $\mathcal{J}_{\tilde{\sigma}}$ contributes exponentially small at large λ.

At large λ, the single geometrical critical point p_{geo} dominates the spinfoam integrals (31) - (34), as discussed in Section II E. Hence, the single Lefschetz thimble $\mathcal{J}_{geo}$ associated to p_{geo} dominates the decomposition (40) of the spinfoam integrals. Therefore, (46) is applicable to the expectation values in (35) at large λ. As a result, we pass from (31) - (34) to integrals on $\mathcal{J}_{geo}$

$$\langle W | \Psi_0 \rangle \simeq \int_{\mathcal{J}_{geo}} \mathrm{d}j \mathrm{d}\phi \, \hat{U} \, e^{-\lambda \hat{\tilde{S}}_{tot}}, \tag{59}$$

$$\langle W|E_n^a \cdot E_n^b E_m^c \cdot E_m^d|\Psi_0\rangle \simeq \int_{\mathcal{J}_{geo}} \mathrm{d}j\mathrm{d}\phi\, \hat{U}\, \mathrm{e}^{-\lambda \hat{S}_{tot}}(\hat{A}_{an} \cdot \hat{A}_{bn})(\hat{A}_{cm} \cdot \hat{A}_{dm}), \tag{60}$$

$$\langle W|E_n^a \cdot E_n^b|\Psi_0\rangle \simeq \int_{\mathcal{J}_{geo}} \mathrm{d}j\mathrm{d}\phi\, \hat{U}\, \mathrm{e}^{-\lambda \hat{S}_{tot}}(\hat{A}_{an} \cdot \hat{A}_{bn}), \tag{61}$$

$$\langle W|E_m^c \cdot E_m^d|\Psi_0\rangle \simeq \int_{\mathcal{J}_{geo}} \mathrm{d}j\mathrm{d}\phi\, \hat{U}\, \mathrm{e}^{-\lambda \hat{S}_{tot}}(\hat{A}_{cm} \cdot \hat{A}_{dm}), \tag{62}$$

where all $\hat{U}, \hat{S}_{tot}, \hat{A}_{an}$ are holomorphic functions of complexified spinfoam variables. The expectation values in (35) reduce to the desired forms as $\langle f\rangle$ in (46) and can be computed by MCMC methods.

Equations (59) - (62) capture the contributions of the dominant critical point p_{geo} to (31) - (34) and include all orders of perturbative $1/\lambda$ corrections. Namely, when we expand (31) - (34) as $1/\lambda$ power series at the critical point p_{geo}, under the stationary phase approximation, the power series are the same as expanding (59) - (62) in $1/\lambda$ (see e.g. [47] for a general argument). Nevertheless, the approximation leading to (59) - (62) from (23) - (26) neglect the contributions that are exponentially suppressed at large λ. These contributions are (1) integrals with $k_{ab} \neq 0$ in (23) - (26), (2) extending some integrals to infinite such as $\int \mathrm{d}j_{ab}$ and $\int \mathrm{d}\phi$ on the cover space, and (3) the complex critical points and corresponding Lefschetz thimbles.

What we have shown so far is that each quantity in (13), (19) - (20) and the spinfoam propagator can be expressed as the power series $\sum_s a_s \left(\frac{1}{\lambda}\right)^s$ plus contributions exponentially suppressed (or namely suppressed faster than $O(1/\lambda^N)$ for any integer N) at large λ. Eqs. (59) - (62) capture the power series while neglecting the exponentially suppressed contributions. The power series contain all the perturbative quantum corrections. The exponentially suppressed contributions may be called non-perturbative corrections, as they contain the sub-dominant thimbles associated with the complex critical points generated by the analytical continuation. In this language, Eqs. (59) - (62) capture all perturbative quantum corrections in (13), (19) - (20) while neglecting non-perturbative corrections.

It is known that in the traditional stationary phase expansion, the computational complexity grows exponentially when computing a_s—the coefficient of $O(1/\lambda^s)$ correction—with larger s, so it is very difficult to sum the power series $\sum_s a_s \left(\frac{1}{\lambda}\right)^s$ in the traditional approach. In this sense, our method with the Lefschetz thimble is a powerful way to compute the spinfoam propagator containing perturbative quantum corrections to all orders.

Besides, similar to the idea in [38, 47], we can consider the integral (59) with the Lefschetz thimble as a new definition of the spinfoam model. When generalizing to arbitrary simplicial complex $\mathcal{K}$, we define the spinfoam model on Lefschetz thimble by

$$Z_{\mathcal{J}} = \int_{\mathcal{J}} \mathrm{d}j\mathrm{d}\phi\, \hat{U}_{\mathcal{K}}\, e^{-\lambda \hat{S}_{\mathcal{K}}}, \tag{63}$$

where $\hat{S}_{\mathcal{K}}$ is the analytic continuation of the spinfoam action on $\mathcal{K}$ [18]. Here, $\int \mathrm{d}j$ integrates all internal spins, and $\mathcal{J}$ is the Lefschetz thimble associated with a single critical point. Applying the Lefschetz thimble to spinfoam model has been proposed earlier in the context of coupling to cosmological constant [67–69]. Eq. (63) has the advantage to focus on the contributions from a single critical point and excludes other critical points. In particular, when $\mathcal{J}$ is the thimble of the critical point corresponding to the Lorentzian Regge geometry, it excludes contributions from vector geometries and the geometries with flipping orientations (see e.g. [19, 70] for the classification of critical points). In addition, Eq. (63) is a better formulation from the computational point of view, as the main point of this paper.

Given the critical point, the spinfoam model on Lefschetz thimble has the same perturbative $1/\lambda$ expansion as the usual definition of the spinfoam amplitude and in particular has the same semi-classical limit as the usual spinfoam amplitude, as shown in the numerical results in Section VII. The small λ behavior of (63) is different the usual spinfoam amplitude because non-perturbative corrections are not negligible at small λ.

V. OPTIMIZATIONS

In this section, we provide some technical details of the optimizations used in our computation.

A. Optimizations of Solving Steepest Ascent (SA) Equations

One crucial step of our algorithm is to solve the SA equation (45) for a given initial condition. For spinfoam model, the action is written as $\lambda \tilde{S}_{tot}$ after analytic continuion ($\lambda \tilde{S}_{tot}$ plays the role of $\hat{S}(\vec{z})$). The right hand side of the SA

equation is proportional to the scaling parameter λ. The idea of the ODE numerical solvers is to use a difference equation to approach the given differential equation. In our case, the difference equation is given by

$$\frac{\mathrm{d}z^a}{\mathrm{d}t} = \lambda \frac{\partial \overline{\tilde{S}_{tot}}}{\partial \overline{z^a}} \sim \Delta z^a = \lambda \frac{\partial \overline{\tilde{S}_{tot}}}{\partial \overline{z^a}} \Delta t,$$

where the Δz^a in the left hand side is supposed to be small to bound the numerical error at each update. For a fixed error tolerance in each time step, the time step Δt has to be small at large λ, or at large $|\partial \tilde{S}_{tot}/\partial z^a|$. This fact has two indications. First, at large λ, in order to keep the accuracy, the total evolution can not be too long. Second, at large $|\partial \tilde{S}_{tot}/\partial z^a|$, the numerical solver may be inaccurate.

In view of the first implication, we would let the total evolution time T be an element of $\{\tau/\lambda | \tau \in [0.1, 1]\}$. In our computation, we have $\lambda \in [10^2, 10^7]$ and set the tolerance $\mathcal{E} = 0.1/\lambda$ of the fluctuation of $\mathrm{Im}(\hat{S}_{tot})$. The tolerance $\mathcal{E}$ determines the shape and size of $\hat{V}_\sigma$. The value of T determines the size the $\tilde{\mathcal{J}}_\sigma$ in which the MCMC is actually carried out. Thanks to the fast decaying $\mathrm{e}^{-\lambda \mathrm{Re}(\hat{S}_{tot})}$ when the Markov chains are moving far from the critical point p_σ, we find that a relatively small T is sufficient to generate a $\tilde{\mathcal{J}}_\sigma$ large enough, where the Markov chain can sample all dominantly contributing points. The points outside $\tilde{\mathcal{J}}_\sigma$ contribute exponentially small and are neglected.

The second implication usually happens when the initial point x is not close to the critical point. In this case, $\mathcal{C}_T(x)$ is very far away from the critical point and $|\partial \tilde{S}_{tot}/\partial z^a|$ at $\mathcal{C}_T(x)$ will be large. The real part $\mathrm{Re}(\tilde{S}_{tot})$ is large at $\mathcal{C}_T(x)$. These points contribute exponentially small to the final integral and are negligible. In our work, we use the embedded Runge-Kutta-Fehlberg Method (RKF45) to be the numerical solver of the steepest ascent equation $(45)^{11}$. The solver can automatically determine the step size based on the given tolerance of the total error. In the accept/reject step of the DREAM algorithm, we can directly reject a candidate x^{cand} if one of the following two events occurs in solving $\mathcal{C}_T(x^{cand})$

- The adaptive ODE step size Δt is smaller than a threshold.

- $|\mathrm{Im}(\tilde{S}_{tot}(\mathcal{C}_T(x^{cand}))) - \mathrm{Im}(\tilde{S}_{tot}(x^{cand}))|$ exceeds the tolerance $\mathcal{E}$.

To be concrete, we set the tolerance of the total error for the ODE as 10^{-20}, set the threshold for the smallest ODE step size as 10^{-6}, and the maximum error tolerance on the imaginary part of the action as 0.1.

B. Optimizations of the DREAM

To make our sampling procedure in the DREAM algorithm more efficient, we can optimize the choice initial points, the evolution time T, and burn-in runs.

1. Optimizing the choice of the initial points

In our work, we use 108 parallel chains to run the DREAM algorithm, so we need 108 initial points at the beginning. Although in principle, the initial points can be randomly chosen, a good choice may possibly reduce the number of burn-in runs to save time and computational resources. Our choice of the initial points is given by the following scheme:

- Compute the basis vectors $\{\hat{\omega}_i\}$ of the thimble's tangent space,

- For each $\hat{\omega}_i$, use enumeration method to find a number η_i so that $S_{eff}(\eta_i \hat{\omega}_i)$ is valued in between 1 and 0.1,

- Use $\{\pm \eta_i \hat{\omega}_i; \; i = 1, \cdots, 54\}$ as the initial points.

Since we treat $\mathrm{e}^{-S_{eff}}$ as a Boltzmann factor, we can consider S_{eff} as the energy. By this scheme, our choice covers more directions and synchronizes the energy of the initial points.

---

[11] In principle, other embedded Runge-Kutta should also work.

2. *Optimizing the flow time T*

The flow time T is an undetermined parameter of the algorithm. The setting of T cannot be too large or too small. If T is too small, the fluctuation of $\text{Im}(\tilde{S}_{tot})$ may be too large to keep the result accurate. If T is too large, only a very small portion of $\hat{V}_\sigma$ contributes to the integrals because $\mathrm{e}^{-S_{eff}}$ decays faster at larger T. Then the shape of the target distribution $\mathrm{e}^{-S_{eff}}$ is too sharp when T is too large. In principle, in contrast to the 'moderate' distributions, this sharp distribution can only be approximated by a longer Markov chain. In other words, too large T make the algorithm less efficient. In practice, we suggest the following scheme to deal with the choice of T:

- Run the algorithm with several trials with different time T.

- Update the Markov chains in each T-trial similar number of times.

- Sort the computational results of the propagator (or at least one component of the propagator) from the T-trials in decreasing order of T.

- The results from many T-trials are close to one another. As such, increasing T only adds negligible contribution to the integral. We take the mean value of these results as the final result.

For example, we show our results of the component G_{14}^{2315} of the spinfoam propagator, in the case of $\lambda = 100$ and $\lambda = 10^5$ in FIG. 2 and FIG. 3 respectively.

(a)Absolute values ($\lambda = 100$) (b)Arguments ($\lambda = 100$)
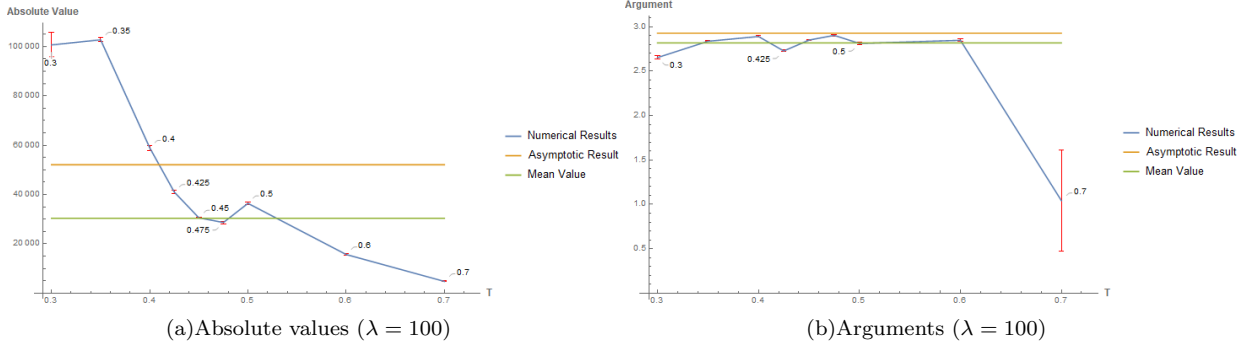
FIG. 2. The absolute values and arguments of G_{14}^{2315} ($\lambda = 100$) computed by the thimble algorithm with different T.

In the case where $\lambda = 100$, the set of flow times $T = \{0.3, 0.35, 0.4, 0.425, 0.35, 0.475, 0.5, 0.6, 0.7\}$ is used for the computations. In each T-trial, the value of G_{14}^{2315} is computed based on roughly 10^7 samples. The results on the blue line in FIG. 2 indicate that, given the number of samples, the results from the T-trials with $T = \{0.425, 0.35, 0.475, 0.5, 0.6\}$ are close to one another, so their mean value $(28373 \pm 610)\mathrm{e}^{\mathrm{i}(2.83 \pm 0.006)}$ (shown as the green line in FIG. 2) is taken as the numerical result of G_{14}^{2315}. This result has $(37.90 \pm 0.6)\%$ percentage difference comparing with the result from the asymptotic expansion up to $O(1/\lambda)$ (shown as the yellow line in FIG. 2).
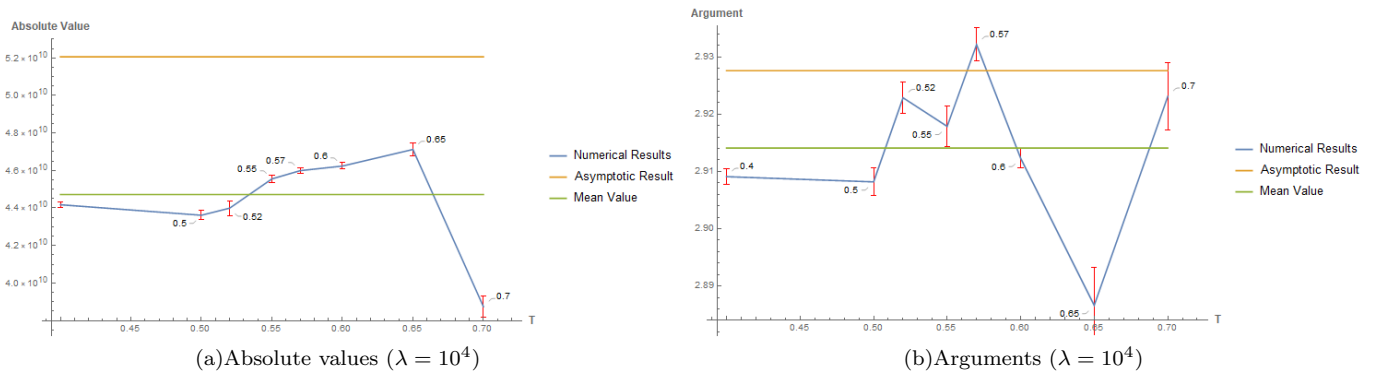

(a)Absolute values ($\lambda = 10^4$) (b)Arguments ($\lambda = 10^4$)

FIG. 3. The absolute values and arguments of G_{14}^{2315} ($\lambda = 10^4$) computed by the thimble algorithm with different T.

Similarly, in the case where $\lambda = 10^4$, we choose the set of flow times $T = \{0.4, 0.5, 0.52, 0.55, 0.57, 0.6, 0.65, 0.7\}$. FIG. 3 depicts the results based on over 10^6 samples[12]. Except the results from the T-trials with $T = \{0.4, 0.7\}$, the results are close to one another and their mean value is $((4.548 \pm 0.015) \times 10^{10})e^{i(2.913 \pm 0.0017)}$ (the green line in FIG. 3). This value differs from that due to the asymptotic expansion (the yellow line in FIG. 3) by $(13.22 \pm 0.28)\%$ of the latter.

3. Optimizing the burn-in stage

As mentioned before, the multi-nomial distribution determining the crossover ratio CR and the scale factor β of the DREAM can be tuned during the burn-in runs. Following the Ref.[46], one can adopt **Algorithm** 1 in the burn-in runs to tune the multi-nomial distribution$(.\ ; p_1, \cdots, p_m)$.

Algorithm 1 DREAM burn-in

1: initial $t \leftarrow 1$, $L_m \leftarrow 0$, $p_m = 1/n_{cr}, m = 1, \cdots, n_{cr}$
2: **while** burn-in steps $t < K$ **do**
3: **for** chains $i = 1, \cdots, M$ **do**
4: $m \sim$ multinomial$(.\ ; p_1, \cdots, p_m)$
5: $CR \leftarrow m/n_{CR}$ and $L_m = L_m + 1$
6: Create a candidate
7: Accept/Reject the candidate
8: $\Delta_m \leftarrow \Delta_m + \sum_{j=1}^{d}((x_i^{(t)})^j - (x_i^{(t-1)})^j)^2/r_j^2$, where r denotes the standard deviation current locations of the chains.
9: **end for**
10: $p_m \leftarrow tN \cdot (\Delta_m/L_m)/\sum_{j=1}^{n_{CR}} \Delta_j$
11: $t \leftarrow t + 1$
12: **end while**

For the scale factor, as mentioned in Ref.[46], if the target distribution is Gaussian, the optimal $\beta = 2.4/\sqrt{2d'\delta}$ yields the acceptance rate equal to 0.44 for $d' = 1$, around 0.2 for larger d'. But based on our test, this choice is not suitable for our case. In order to optimize the performance of the algorithm, we tune β based on the acceptance rate during the burn-in runs. At the beginning of the burn-in stage, we set $\beta = 2.4C/\sqrt{2d'\delta}$ with $C = 1$. In each update during the burn-in, we count the number of accepted candidates as α, then we compute the acceptance rate in this update as $\alpha/108$. If the acceptance rate is greater than 0.4, we multiply C by 1.2, whereas if the acceptance rate is greater than 0.1, we multiply C by 0.5[13]. After a long burn-in stage, we expect that the β is tuned such that the acceptance rate is round 0.3, which is a optimal value for high-dimensional problem.

VI.   THE LARGE SPIN LIMIT OF SPINFOAM PROPAGATOR

In this section, we depart from Lefschetz thimbles and discuss the standard stationary phase analysis in the large-λ limit of the spinfoam propagator. We are going to compare between the large-λ limit the results from MCMC on Lefschetz thimble in Section VII.

As in Refs.[5, 7–9], the spinfoam propagator can be computed by the asymptotic expansion following the stationary phase analysis in Ref.[71]:

$$\left| \int_K u(x)e^{i\lambda f(x)}dx - e^{i\lambda f(x_0)}\left[\det\left(\frac{\lambda f''(x_0)}{2\pi i}\right)\right]^{-\frac{1}{2}} \sum_{s=0}^{k-1}\left(\frac{1}{\lambda}\right)^s L_s u(x_0) \right| \leq C\left(\frac{1}{\lambda}\right)^k \sum_{|\alpha| \leq 2k} \sup |D^\alpha u|, \qquad (64)$$

with

$$g_{x_0}(x) = f(x) - f(x_0) - \frac{1}{2}H^{ab}(x_0)(x - x_0)_a(x - x_0)_b,$$

---

[12] Compared with the case of $\lambda = 100$, 10^6 is already a sufficiently large number of samples to make the results at $\lambda = 10^4$ converge because the contributed region in $\hat{V}$ at $\lambda = 10^4$ is smaller and thus easier to be simpled than that at $\lambda = 100$.

[13] The threshold and the multiple rate here is chosen by hand. One may need to adjust them for other computing tasks.

and

$$L_s u(x_0) = i^{-s} \sum_{l-m=s} \sum_{2l \geq 3m} \frac{(-1)^l 2^{-l}}{l! m!} \left[\sum_{a,b=1}^{n} H_{ab}^{-1}(x_0) \frac{\partial^2}{\partial x_a \partial x_b} \right]^l (g_{x_0}^m u)(x_0),$$

where $H(x) = f''(x)$ is the Hessian matrix. Using the parametrization mentioned in Section II, we use $Mathematica^{\text{TM}}$ to compute the Hessian of the spinfoam action and derivatives of A_{ab}^i and U defined in Section II C. Then following (64), we compute the $1/\lambda$ expansion of $\langle E_n^a \cdot E_n^b E_m^c \cdot E_m^d \rangle$, $\langle E_n^a \cdot E_n^b \rangle$, and the spinfoam propagator G_{mn}^{abcd}. The code of this computation is shared in [72]. The results are used as the reference data in comparison with the Lefschetz thimble Monte-Carlo computations.

For example, if we keep the expansion (64) to the first order of $1/\lambda$ (keeping the terms of $s = 0, 1$), the components $\langle E_1^2 \cdot E_1^3 E_4^1 \cdot E_4^5 \rangle$, $\langle E_4^1 \cdot E_4^5 \rangle$, $\langle E_1^2 \cdot E_1^3 \rangle$ and G_{14}^{2315} are given by

$$\frac{\langle E_1^2 \cdot E_1^3 E_4^1 \cdot E_4^5 \rangle}{\lambda^4} = 0.006944 + \frac{0.03659 - 0.009716\mathrm{i}}{\lambda} + O\left(\frac{1}{\lambda^2}\right), \tag{65}$$

$$\frac{\langle E_1^2 \cdot E_1^3 \rangle}{\lambda^2} = -0.08333 + \frac{2.292 - 0.5092\mathrm{i}}{\lambda} + O\left(\frac{1}{\lambda^2}\right) \tag{66}$$

$$\frac{\langle E_4^1 \cdot E_4^5 \rangle}{\lambda^2} = -0.08333 + \frac{1.242 - 0.2599\mathrm{i}}{\lambda} + O\left(\frac{1}{\lambda^2}\right), \tag{67}$$

$$G_{14}^{2315} \sim -(0.05087 - 0.01106\mathrm{i})\lambda^3 + O(\lambda^2) \tag{68}$$

We choose the Barbero-Immirzi parameter as $\gamma = -0.1$ in the above numerics.

The λ^3 behavior of the spinfoam propagator can be seen analytically, by the known leading order formula [5, 7–10]:

$$G_{mn}^{abcd} \sim \lambda^{-1} H_{\alpha\beta}^{-1} \partial_\alpha (A_{an} \cdot A_{bn}) \partial_\beta (A_{bm} \cdot A_{dm}), \tag{69}$$

where H is the Hessian of S_{tot} at the critical point p_{geo}, and the indices α, β correspond to the variables j and ϕ defined in Section II. By definition (17), A_{an} is proportional to λ, so (69) is at the order of λ^3. This result turns out to be consistent with our numerical result from the Lefschetz thimble Monte-Carlo in the large-λ limit.

VII. NUMERICAL RESULTS

Recall (14) that the spinfoam propagator is obtained by computing the expectation values $\langle E_n^a \cdot E_n^b E_m^c \cdot E_m^d \rangle$ and $\langle E_n^a \cdot E_n^b \rangle$. We choose the Barbero-Immirzi parameter $\gamma = -0.1$ in this computation and compute these expectation values and the propagator in the situations where $\lambda = 50, 100, 1000, 10000, 50000, 100000, 500000, 1000000, 5000000, 10000000$ and 50000000. In the first and second parts of this section, we show the numerical results respectively of the expectation values and of the propagator.

A. Expectation Values

The expectation value of $\langle E_n^a \cdot E_n^b E_m^c \cdot E_m^d \rangle$ is a tensor with 1275 non-zero components, and the expectation value of $\langle E_n^a \cdot E_n^b \rangle$ consists 50 non-zero components. We numerically compute these components depending on a sufficiently large number (over 10^7) of samples obtained by the MCMC method. As an example of our computation, Figs. 4, 5, and 6 plot the absolute values and the arguments of the components $\langle E_1^2 \cdot E_1^3 E_4^1 \cdot E_4^5 \rangle$, $\langle E_4^1 \cdot E_4^5 \rangle$ and $\langle E_1^2 \cdot E_1^3 \rangle$ respectively. In all these plots, the numerical results are shown on the blue lines, and the results given by the asymptotic expansion (65), (66), (67), (68) are shown on the yellow lines. TABLE VII, TABLE VIII and TABLE IX record the percentage difference between the numerical results and the asymptotic results.

On the one hand, our results match the results from the asymptotics very well in the large spin limit. For the components we show here, the percentage differences between the asymptotic results and the numerical results are smaller than 0.03% when λ is greater than 10^5, and the percentage differences tend to become smaller at larger λ. On the other hand, when λ is small, the percentage differences become large because the higher order $1/\lambda$ corrections become important in this realm.
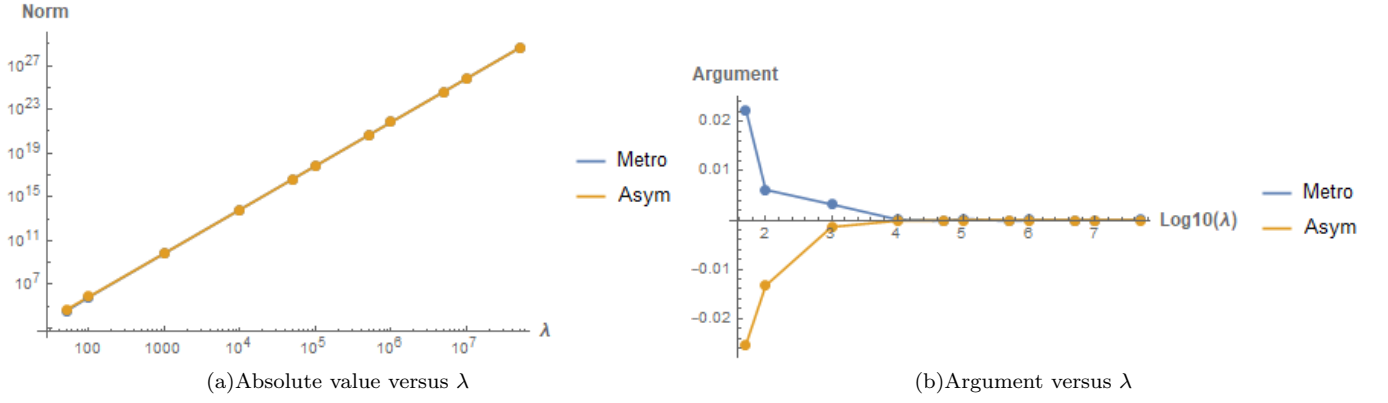
FIG. 4. The absolute values and the arguments of $\langle E_1^2 \cdot E_1^3 E_4^1 \cdot E_4^5 \rangle$. The results of the asymptotics are shown in the yellow lines, and the numerical results are indicated by blue line.
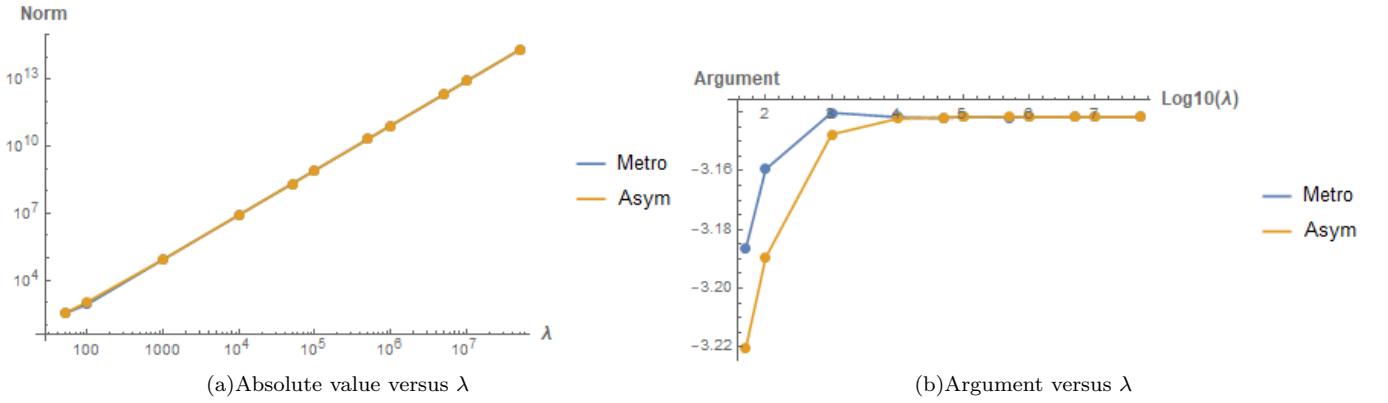

FIG. 5. Absolute values and arguments of $\langle E_1^2 \cdot E_1^3 \rangle$. The results of the asymptotics are drawn in yellow line, and the numerical results are indicated by blue line.
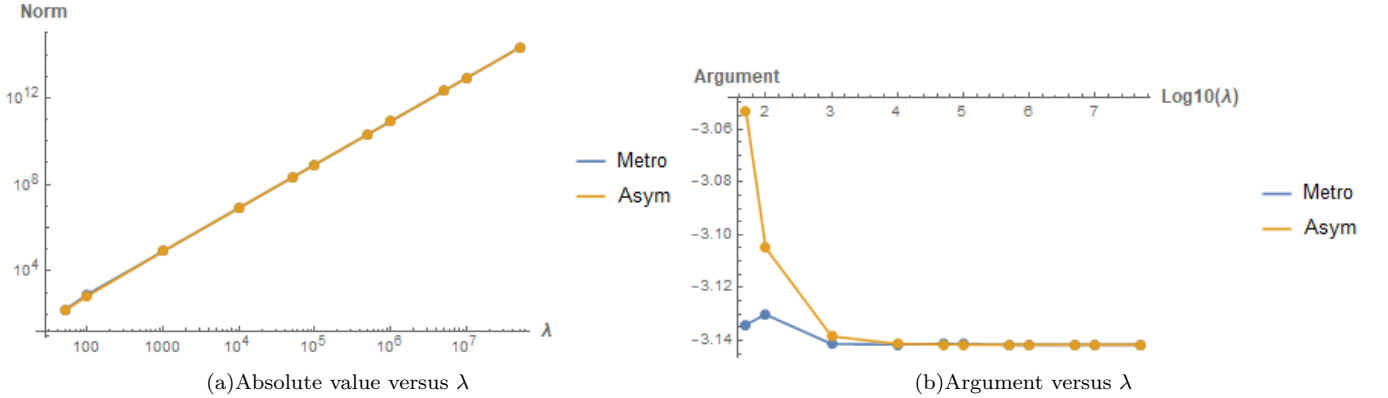

FIG. 6. Absolute values and arguments of $\langle E_4^1 \cdot E_4^5 \rangle$. The results of the asymptotics are drawn in yellow lines, and the numerical results are indicated by the blue lines.

TABLE VII. The difference between the numerically computed and asymptotically expanded $\langle E_1^2 \cdot E_1^3 E_4^1 \cdot E_4^5 \rangle$

λ	10^2	10^3	10^4	5×10^4	10^5	5×10^5	10^6	5×10^6	10^7	5×10^7
Difference (%)	8.71	0.79	0.12	0.052	0.036	0.017	0.0062	0.0018	0.00037	0.00069

TABLE VIII. The difference between the numerically computed and asymptotically expanded $\langle E_1^2 \cdot E_1^3 \rangle$

λ	10^2	10^3	10^4	5×10^4	10^5	5×10^5	10^6	5×10^6	10^7	5×10^7
Difference (%)	22.32	2.00	0.31	0.078	0.022	0.016	0.012	0.0022	0.000047	0.0016

TABLE IX. The difference between the numerically computed and asymptotically expanded $\langle E_4^1 \cdot E_4^5 \rangle$

λ	10^2	10^3	10^4	5×10^4	10^5	5×10^5	10^6	5×10^6	10^7	5×10^7
Difference (%)	18.66	1.18	0.18	0.026	0.017	0.00054	0.0037	0.00035	0.00036	0.00083

B. Spinfoam Propagator

The propagator G_{mn}^{abcd} also has 1275 non-zero components. Figure 7 plots the absolute values and the arguments of the component G_{14}^{2315}. The percentage differences between the asymptotic limit (68) and numerical results from the Lefschetz thimble Monte-Carlo are shown in TABLE X. For the results with $\lambda > 10^6$, the percentage difference is smaller than 4%. This comparison shows that the asymptotic expansion and numerical Lefschetz thimble Monte-Carlo are consistent in the large spin limit. Similar to the computation of the expectation values, their differences become large in small spin realm because of the non-negligible contributions of the higher order $1/\lambda$ corrections.
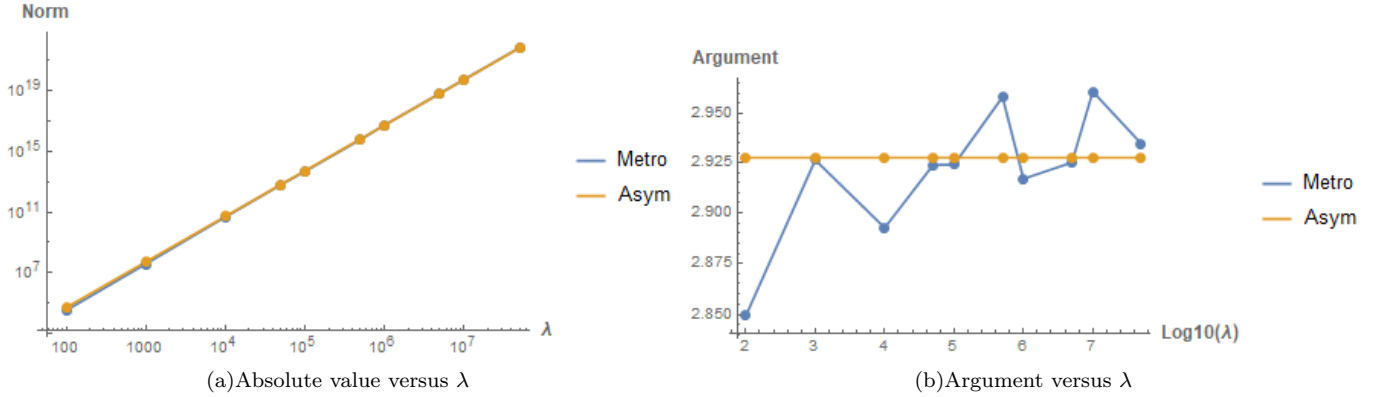

(a)Absolute value versus λ (b)Argument versus λ

FIG. 7. Absolute values and arguments of G_{14}^{2315} corresponding to different λ. The results of the asymptotics are drawn in yellow line, and the numerical results are indicated by blue line.

We compute all the 1275 components of the propagator G_{mn}^{abcd} and compare them with the results from the leading order asymptotics (69). Figure 8(a) shows the histograms of the percentage differences of the components of G_{mn}^{abcd} between the asymptotic limit (69) and the results from the Lefschetz thimble Monte-Carlo, at $\lambda = 10^6$, based on 8985600 samples, 9504000 samples and 12787200 samples. As we can see, the percentage differences for most of the components (1067 components for the result of 8985600 samples, 1095 components for the result of 9504000 samples and 1144 components for the result of 12787200 samples) are smaller than 10%. There are however several components with percentage differences greater than 100%, one of which is nearly as much as 120% in the 12787200 samples case. But a strong tendency that the percentage differences become smaller with respect to the increasing number of the samples (the maximum difference decreasing from 147% in the 8985600 samples cases to 120% in the 16761600 samples case) implicates that when $\lambda = 10^6$, 12787200 samples may not be enough to make the Markov chains perfectly converging to the desired distribution and cause such big differences. The percentage differences of these components will further decrease when the number of the samples increases.

Figure 8 (b) draws a comparison between the histogram of the percentage differences of the components when $\lambda = 10^7$ and when $\lambda = 10^6$. The results are all achieved with 12787200 samples. In the case of $\lambda = 10^7$, the percentage differences of most of the components are less than 10% and the maximum difference is around 45%. The comparison shows that the Markov chains converge to the desired distribution faster than they do in the case of $\lambda = 10^6$, and the Lefschetz thimble Monte-Carlo results for $\lambda = 10^7$ are more consistent to the asymptotic limit (69). This fact might suggest that when $\lambda = 10^7$, the less important $1/\lambda$ correction and the easier converging Markov chains are correlated.

In summary, the expectation values of the metric operators and the propagator obtained from Lefschetz thimble Monte-Carlo show their compatibility to the asymptotics from the stationary phase analysis in the large-λ limit. As

TABLE X. The difference between the numerically computed and asymptotically expanded G_{14}^{2315}

λ	10^2	10^3	10^4	5×10^4	10^5	5×10^5	10^6	5×10^6	10^7	5×10^7
Difference (%)	37.90	27.00	13.22	2.76	10.09	8.86	1.89	1.13	3.90	2.06

(a)$\lambda = 10^6$ with different number of samples

(b)$\lambda = 10^6$ v.s. $\lambda = 10^7$
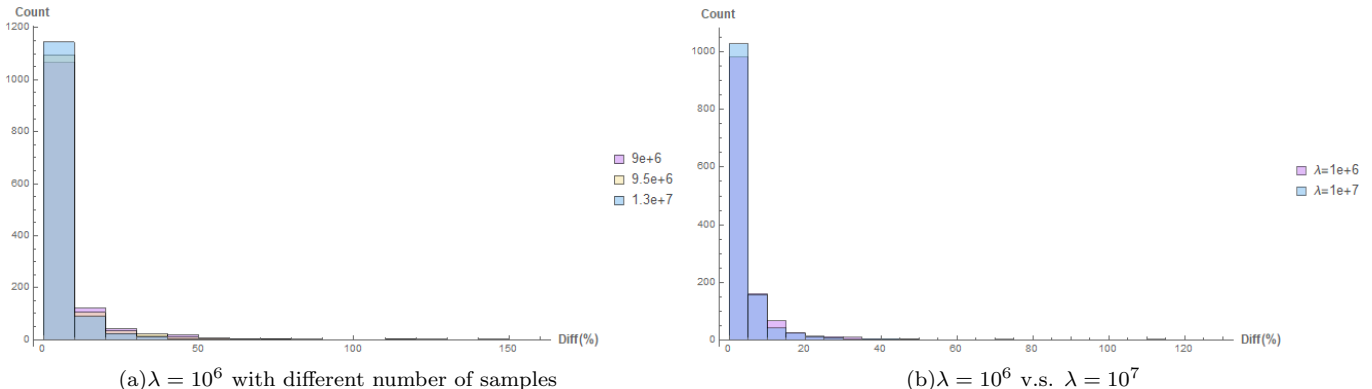
FIG. 8. Histogram of the percentage errors of the components of G_{mn}^{abccd} for (a) $\lambda = 10^6$ and (b) $\lambda = 10^7$.

λ increases, the compatibility to the asymptotics tends to be improved. These results fulfill our expectation about the semi-classical behavior of the spinfoam propagator and validates our algorithm and coding.

VIII. BENCHMARKS

TABLE XI. The computer platforms

Platform	CPU	RAM	OS	Mathematica™Version
(1)	AMD EPYC™7742x2	512G DDR4-3200	Ubuntu™20.04.1 LTS	12
(2)	AMD EPYC™7642x2	512G DDR4-3200	Ubuntu™20.04.1 LTS	12
(3)	AMD Ryzen™3800XT	32G DDR4-3200	Windows™10 version 2004	12

We have tested our code [54] in three different platforms shown in TABLE XI. On platforms (1) and (2), the code runs with 54 parallel Mathematica™kernels, and for platform (3), the code runs with 16 parallel Mathematica™kernels. Both platforms (1) and (2) can update 60000 samples per hour, and platform (3) can update 20000 samples per hour.

IX. CONCLUSION

In this paper, we propose a numerical method (combining the methods of Lefschetz thimble and DREAM) that can compute the expectation value of any observable in a quantum system described by a complex-valued action. We apply our method to compute the EPRL spinfoam propagator on a 4-simplex. Our computation focuses on the spinfoam propagator with relatively large spins $\lambda \geq 100$. Our results not only comply with the expected spinfoam semiclassical behavior but also gives rise to a quantum correction. The theory of Lefschetz thimble indicates that the quantum correction due to our computation sums $1/\lambda$ corrections to all orders.

In principle, the method is applicable to all types of the spin foam model with Lorentzian or Euclidean signature, with different choices of the Y-map and with the different values of the Barbero-Immirzi parameter γ. Although, the explicit propagator components are different with respect to these choices, the conclusion that the numerical results comply with the semiclassical behavior holds for all these choices.

Our method is efficient and scalable for the numerical computation of the spinfoam model. We are capable of numerically computing spinfoams with relatively large spins. Our method can also compute the spinfoam model with multiple 4-simplices. The Lefschetz thimble and Monte-Carlo methods are able to compute oscillatory integrals of a few hundreds variables (see e.g. [39, 73] and the references therein). For instance, the EPRL spinfoam model (with the coherent intertwiner boundary) on the complex Δ_3 of three 4-simplices and an internal face is a 133-(real)dimensional

integral (see [70] for the integral formula on multiple 4-simplices), and thus can be handled by our method. Therefore, a future task is to compute the correlation functions of the spinfoam model on Δ_3. An interesting aspect of the spinfoam model on Δ_3 is its relation to the flatness problem: The spinfoam integral is dominated by the flat geometry (with vanishing deficit angle) when spins are large [18, 27, 74–76]. Hence, it is interesting to compute the spinfoam expectation value of the deficit angle on Δ_3 and demonstrate its dependence on spins. Our method is suitable for computing expectation values and especially for studying their behaviors with large spins.

As another interesting future work, we can apply the Pachner 1-5 move and subdivide a 4-simplex into five 4-simplices, the spinfoam model on the resulting complex Δ_5 is a 230-(real)dimensional integral, and is expected to be handled by our method. Our method can compute spinfoam expectation values and correlation functions on Δ_5, compare them with the results on a 4-simplex, and understand their behaviors under the 1-5 move. Similar studies should be applied to the other elementary Pachner moves. The resulting behaviors should be useful for studying the renormalization of spinfoam model under changing triangulation.

As an efficient way to compute oscillatory integrals, our method has wide applications and is not restricted to LQG/Spinfoam. In addition to Lattice Field Theory, where similar methods have been extensively applied, our method also applies to Topological Field Theory and Knot Theory. The analytically continued Chern-Simons theory and colored Jones polynomial are related to the finite dimensional integrals on Lefschetz thimbles. These finite dimensional integrals are known as holomorphic blocks [37, 77] (see also [69] for the relation with spinfoams). It is natural to apply our method to compute holomorphic blocks because they are defined on Lefschetz thimbles.

Appendix A: Markov Chain Monte Carlo Methods

MCMC methods comprise a class of algorithms designed for sampling from a posterior probability distribution $\pi(x)$ on a given space V. In this appendix we review the main idea of the MCMC methods and several specific algorithms

1. Markov Chain Monte Carlo methods in general

A Markov chain is defined as a stochastic model describing a sequence of random variates in which the probability distribution of each variate only depends on the the value of the previous one variate attained. For a Markov chain sequentially comprises N sampled data, the Markov chain central limit theorem [78][43] guarantees that, when $N \to \infty$, the Markov chain will reach its equilibrium state so that sampled points by Markov chain will converge to a posterior probability distribution. An equilibrium chain must follow the Bayes local balance condition:

$$\pi(y)K(x|y) = \pi(x)K(y|x), \tag{A1}$$

where $\pi(x)$ is the posterior probability distribution and the transition kernel $K(x|y)$ is the conditional probability distribution of a random variate on the chain if its previous random variate is sampled as a value y.

Imagine that a Markov chain as a 'walker' moving on a phase space. Once a point being reached by the 'walker', it is considered being sampled once. Then the Bayes local balance condition means that the marginal probability distribution of the 'walker' first appears on x then moves to y is the same as the one that the 'walker' appears on y then moves to x. Thus the future movement of the 'walker' has no bias. The distribution of the sampled point will converge to the static probability distribution $\pi(x)$.

The idea of the MCMC integral method is to simulate such a 'walker' randomly walking on phase space. If each step of the 'walker's' random movement is designed to satisfy the transition kernel $K(x|y)$, after a long march, the history of 'walker's' random movement will follow a posterior distribution, e.g., $\exp(S_{eff})$. Then, one can compute the mean value of the function $f(x)$ among the sampled history points to approximate $\langle f \rangle_{eff}$.

2. Metropolis algorithm

Following the idea of MCMC, one important question is how to construct the transition kernel $K(x|y)$ so that the Markov chain can sample from a desired posterior distribution like $\exp(S_{eff})$. In Metropolis algorithm[79] [43, Chapter1.12], which is a type MCMC method, the transition kernel $K(x|y)$ is constructed as

$$K(x|y) = \alpha(x, y)p(x|y), \tag{A2}$$

where $p(x|y)$ can be any proposal transition distribution and the acceptance rate $\alpha(x, y)$ is defined as

$$\alpha(x, y) = \min \left\{ 1, \frac{\pi(x)p(y|x)}{\pi(y)p(x|y)} \right\}. \tag{A3}$$

One can easily check the validation of the algorithm by plugging (A2) back to the Bayes local balance condition.

3. Metropolis-Hastings algorithm

As an improvement of the Metropolis algorithm, the Metropolis-Hastings (MH) algorithm uses symmetric proposal transition distributions to construct transition kernels $K(x|y)$. When a transition distribution is symmetric, $p(x|y) = p(y|x)$ and the acceptance rate $\alpha(x, y)$ reduces to

$$\alpha(x, y) = \min \left\{ 1, \frac{\pi(x)}{\pi(y)} \right\}. \tag{A4}$$

In details, the MH algorithm generate the samples in the steps shown in **Algorithm** 2

Algorithm 2 Metropolis-Hastings (MH) algorithm

1: initial $x^{(0)}$
2: **for** iteration $i = 1, 2, \cdots N$ **do**
3: Propose candidate x^{cand} from $p(x|x^{(i-1)})$
4: Acceptance rate $\alpha \leftarrow \min \left\{ 1, \frac{\pi(x^{cand})}{\pi(x^{(i-1)})} \right\}$
5: $u \sim \text{Uniform}(u; 0, 1)$
6: **if** $u < \alpha$ **then**
7: $x^{(i)} \leftarrow x^{cand}$
8: **else**
9: $x^{(i)} \leftarrow x^{(i-1)}$
10: **end if**
11: **end for**

When N goes to be large, the sampled data $\{x^{(i)}\}$ follows the posterior distribution $\pi(x)$.

4. Adaptive Metropolis-Hastings Algorithm

The simplicity of the MH algorithm make it easy to be applied; however, the performance of the algorithm depends on the tuning of some internal variables, such as, the scale and orientation of the proposal distribution. On one hand, if the proposal distribution is too wide, most of the candidates will be rejected and the chain's convergence to the target distribution will be delayed. On the other hand, if the proposal distribution is too narrow, although most of the candidate will be accept, it may take a very large number of updates to make the Markov chain move to the most probable region and converge to the target distribution. One improved MH is the Adaptive Metropolis Hastings (AM) algorithm [80, 81]. The AM algorithm uses single Markov chain, but it is a powerful algorithm that can automatically select the appropriate proposal distribution. For a d-dimensional phase space, by using the sampled data, AM continuously adapt the covariance, denoted by C_t, of the Gaussian proposal distribution $p(x|x^{(i-1)}) = N(x^{(i_1)}, C_t)$. Explicitly,

$$C_t = S_d Cov(x^{(0)}, \cdots, x^{(i-1)}) + \epsilon I_d, \tag{A5}$$

where the S_d is the scaling factor depending on d, I_d is the d-dimensional identity matrix, and ϵ is a very small data comparing to the scale of $\{x^{(i)}\}$. If the target distribution is a Gaussian distribution, the optimal S_d is $2.4^2/d$ so that the acceptance rate α can stay around 0.3. The detailed steps of AM (**Algorithm** 3) is similar to MH.

Algorithm 3 Adaptive Metropolis-Hastings (AM) algorithm

1: initial $x^{(0)}$, $C_t \leftarrow I_d$
2: **for** iteration $i = 1, 2, \cdots N$ **do**
3: Propose candidate x^{cand} from $N(x^{(i_1)}, C_t)$
4: Acceptance rate $\alpha \leftarrow \min\left\{1, \frac{\pi(x^{cand})}{\pi(x^{(i-1)})}\right\}$
5: $u \sim$Uniform$(u; 0, 1)$
6: **if** $u < \alpha$ **then**
7: $x^{(i)} \leftarrow x^{cand}$
8: **else**
9: $x^{(i)} \leftarrow x^{(i-1)}$
10: **end if**
11: $C_t \leftarrow S_d Cov(x^{(0)}, \cdots, x^{(i)}) + \epsilon I_d$
12: **end for**

---

5. Differential evolution Markov Chain algorithm

The AM algorithm works fine for many simple inference problems, but its efficiency becomes low when dealing with complicated posterior distribution, especially for high-dimensional problems. In order to deal with these limitations, Differential evolution Markov chain (DE-MC) [82] is developed. In DE-MC, M different Markov chains $\{x_s^{(t)}, s = 1, \cdots, M\}$ are run in parallel. In stead of using the covariance of the previoused samples, i.e.,$Cov(x^{(0)}, \cdots, x^{(i-1)})$, DE-MC uses the current location of the chains to generate the candidates. At each updating step, instead of using proposal distribution to generate candidates, DE-MC uses a genetic algorithm, called Differential evolution algorithm, to generate the the candidates as

$$x_s^{cand} = x_s^{(t-1)} + \gamma(x_{s_1}^{(t-1)} - x_{s_2}^{(t-1)}) + \epsilon, \ s = 1, \cdots M, \tag{A6}$$

where γ is a scaling factor, s_1 and s_2 are labels of another two chains different from the chain s, and ϵ is a number draw from the uniform distribution Uniform$(-b, b)$ with $|b| < 1$. Similar to the AM algorithm, when the posterior distribution is d-dimensional Gaussian, the optimal choice of γ is $2.4/\sqrt{2d}$. For every 10 update steps, $\gamma = 1$ to allow a direct jumps between the chains. When dealing with multimodal posterior distribution, this brings in a huge of advantage comparing to single chain AM algorithm. In AM algorithm, it is hard to make the Markov chain tunnel between two distinct possible regions, while in DE-MC, different chains can 'explore' different regions simultaneously and the jumps between the regions are allowed so that the samples from different regions are well mixed to satisfy the target posterior distribution. The steps of the DE-MC algorithm is shown in **Algorithm** 4.

The DE-MC is efficiently accommodate to the situation when the target posterior distribution is complicated, and high-dimensional. The efficiency can be further enhanced by making several modifications on the algorithm. These modifications brings in the Differential Evolution Adaptive Metropolis (DREAM) algorithm used in our paper to compute the propagator.

Algorithm 4 DE-MC algorithm

1: initial $x_s^{(0)}$, $s = 1, \cdots, M$
2: **for** iteration $i = 1, 2, \cdots N$ **do**
3: **for** chains $j = 1, \cdots, M$ **do**
4: **while** $s_1 = j$ **do**
5: $s_1 \leftarrow$ RandomInteger$(1, M)$
6: **end while**
7: **while** $s_2 = j$ or $s_2 = s_1$ **do**
8: $s_2 \leftarrow$ RandomInteger$(1, M)$
9: **end while**
10: Propose candidate $x^{cand} \leftarrow x_j^{(i-1)} + \gamma(x_{s_1}^{(i-1)} - x_{s_2}^{(i-1)}) + \epsilon$
11: Acceptance rate $\alpha \leftarrow \min\left\{1, \frac{\pi(x^{cand})}{\pi(x_j^{(i-1)})}\right\}$
12: $u \sim$ Uniform$(u; 0, 1)$
13: **if** $u < \alpha$ **then**
14: $x_j^{(i)} \leftarrow x_j^{cand}$
15: **else**
16: $x_j^{(i)} \leftarrow x_j^{(i-1)}$
17: **end if**
18: **end for**
19: **end for**

---

[1] T. Thiemann, *Modern Canonical Quantum General Relativity* (Cambridge University Press, 2007).
[2] C. Rovelli and F. Vidotto, *COVARIANT LOOP QUANTUM GRAVITY an elementary introduction to quantum gravity and spin foam theory*, ISBN 978-1-107-06962-6 (Cambridge University Press, 2015).
[3] M. Han, W. Huang, and Y. Ma, Int.J.Mod.Phys. **D16**, 1397 (2007), gr-qc/0509064.
[4] A. Ashtekar and J. Lewandowski, Class.Quant.Grav. **21**, R53 (2004), gr-qc/0404018.
[5] C. Rovelli, Phys.Rev.Lett. **97**, 151301 (2006), gr-qc/0508124.
[6] C. Rovelli and M. Zhang, Class.Quant.Grav. **28**, 175010 (2011), 1105.0566.
[7] E. Bianchi, L. Modesto, C. Rovelli, and S. Speziale, Class.Quant.Grav. **23**, 6989 (2006), gr-qc/0604044.
[8] E. Bianchi, E. Magliaro, and C. Perini, Nucl.Phys. **B822**, 245 (2009), 0905.4082.
[9] E. Bianchi and Y. Ding, Phys.Rev. **D86**, 104040 (2012), 1109.6538.
[10] M. Han, Z. Huang, and A. Zipfel, Phys. Rev. **D97**, 084055 (2018), 1711.11162.
[11] J. Engle, E. Livine, R. Pereira, and C. Rovelli, Nucl.Phys. **B799**, 136 (2008), 0711.0146.
[12] L. Freidel and K. Krasnov, Class. Quant. Grav. **25**, 125018 (2008), 0708.1595.
[13] M. Reisenberger and C. Rovelli, in *2001: A relativistic spacetime odyssey. Proceedings, 25th Johns Hopkins Workshop on Problems in particle theory, Florence, Italy, September 3-5, 2001* (2000), pp. 431–448, gr-qc/0002083.
[14] A. Perez, Class.Quant.Grav. **20**, R43 (2003), gr-qc/0301113.
[15] J. W. Barrett, R. Dowdall, W. J. Fairbairn, F. Hellmann, and R. Pereira, Class.Quant.Grav. **27**, 165009 (2010), 0907.2440.
[16] J. W. Barrett, W. J. Fairbairn, and F. Hellmann, Int. J. Mod. Phys. A **25**, 2897 (2010), 0912.4907.
[17] F. Conrady and L. Freidel, Phys.Rev. **D78**, 104023 (2008), 0809.2280.
[18] M. Han, Class.Quant.Grav. **31**, 015004 (2014), 1304.5627.
[19] M. Han and M. Zhang, Class.Quant.Grav. **30**, 165012 (2013), 1109.0499.
[20] M. Han and M. Zhang, Class.Quant.Grav. **29**, 165004 (2012), 1109.0500.
[21] W. Kaminski, M. Kisielowski, and H. Sahlmann, Class. Quant. Grav. **35**, 135012 (2018), 1705.02862.
[22] H. Liu and M. Han, Physical Review D **99** (2019), ISSN 2470-0029, URL http://dx.doi.org/10.1103/PhysRevD.99.084040.
[23] M. Han, Z. Huang, and A. Zipfel, Phys. Rev. D **100**, 024060 (2019), 1812.02110.
[24] P. Dona and S. Speziale, Phys. Rev. D **102**, 086016 (2020), 2007.09089.
[25] H. M. Haggard, M. Han, W. Kaminski, and A. Riello, Nucl. Phys. **B900**, 1 (2015), 1412.7546.
[26] P. Donà, M. Fanizza, G. Sarno, and S. Speziale (2019), 1903.12624.
[27] P. Dona, F. Gozzini, and G. Sarno, *Numerical analysis of spin foam dynamics and the flatness problem* (2020), 2004.12911.
[28] P. Donà and G. Sarno, General Relativity and Gravitation **50** (2018), ISSN 1572-9532, URL http://dx.doi.org/10.1007/s10714-018-2452-7.
[29] B. Bahr and S. Steinhaus, Phys. Rev. Lett. **117**, 141302 (2016), 1605.07649.
[30] B. Bahr, G. Rabuffo, and S. Steinhaus, Phys. Rev. D **98**, 106026 (2018), 1804.00023.

[31] S. Speziale, JHEP **05**, 039 (2006), gr-qc/0512102.
[32] J. D. Christensen, E. R. Livine, and S. Speziale, Physics Letters B **670**, 403 (2009), ISSN 0370-2693, URL http://www.sciencedirect.com/science/article/pii/S0370269308013415.
[33] J. D. Christensen, I. Khavkine, E. R. Livine, and S. Speziale, Classical and Quantum Gravity **27**, 035012 (2010), URL https://doi.org/10.1088/0264-9381/27/3/035012.
[34] K. N. Anagnostopoulos and J. Nishimura, Phys. Rev. D **66**, 106008 (2002), URL https://link.aps.org/doi/10.1103/PhysRevD.66.106008.
[35] P. de Forcrand, *Simulating qcd at finite density* (2010), 1005.0539.
[36] O. Philipsen, PoS **ConfinementVIII**, 011 (2012).
[37] E. Witten, AMS/IP Stud. Adv. Math. **50**, 347 (2011), 1001.2933.
[38] E. Witten (2010), 1009.6032.
[39] A. Alexandru, G. Basar, P. F. Bedaque, and N. C. Warrington (2020), 2007.05436.
[40] M. Cristoforetti, F. Di Renzo, A. Mukherjee, and L. Scorzato, PoS **LATTICE2013**, 197 (2014), 1312.1052.
[41] A. Alexandru, P. F. Bedaque, H. Lamm, and S. Lawrence, Phys. Rev. D **96**, 094505 (2017), 1709.01971.
[42] A. Alexandru, G. Başar, and P. Bedaque, Phys. Rev. D **93**, 014504 (2016), 1510.03258.
[43] S. Brooks, X.-L. Meng, G. Jones, and A. Galman, *Handbook for Markov chain Monte Carlo* (Taylor and Francis, 2011).
[44] A. Barrau, T. Cailleteau, X. Cao, J. Diaz-Polo, and J. Grain, Phys. Rev. Lett. **107**, 251301 (2011), 1109.4239.
[45] S. Steinhaus and J. Thürigen, Phys. Rev. D **98**, 026013 (2018), 1803.10289.
[46] J. Vrugt, C. ter Braak, C. Diks, B. Robinson, J. Hyman, and D. Higdon, International Journal of Nonlinear Sciences and Numerical Simulation **10**, 273 (2009), ISSN 1565-1339.
[47] M. Cristoforetti, F. Di Renzo, and L. Scorzato (AuroraScience), Phys. Rev. **D86**, 074506 (2012), 1205.3996.
[48] T. Kanazawa and Y. Tanizaki, Journal of High Energy Physics **2015**, 44 (2015), ISSN 1029-8479, URL https://doi.org/10.1007/JHEP03(2015)044.
[49] Y. Tanizaki, Y. Hidaka, and T. Hayata, New Journal of Physics **18**, 033002 (2016), URL https://doi.org/10.1088/1367-2630/18/3/033002.
[50] M. Fukuma and N. Matsumoto, *Worldvolume approach to the tempered lefschetz thimble method* (2020), 2012.08468.
[51] Y. Tanizaki, H. Nishimura, and K. Kashiwa, Phys. Rev. D **91**, 101701 (2015), URL https://link.aps.org/doi/10.1103/PhysRevD.91.101701.
[52] M. Ulybyshev, C. Winterowd, and S. Zafeiropoulos, Phys. Rev. D **101**, 014508 (2020), URL https://link.aps.org/doi/10.1103/PhysRevD.101.014508.
[53] M. Ulybyshev, C. Winterowd, and S. Zafeiropoulos (2019), 1906.02726.
[54] H. Zichang, *Spinfoam propagator code*, https://gitee.com/ZCHuang1126/spinfoam-propagator.git (2020).
[55] M. Han, Z. Huang, H. Liu, and D. Qu, Phys. Rev. D **102**, 124010 (2020), 2007.01998.
[56] R. Oeckl, *The general boundary approach to quantum gravity* (2003), gr-qc/0312081.
[57] R. Oeckl, Advances in Theoretical and Mathematical Physics **12**, 319–352 (2008), ISSN 1095-0753, URL http://dx.doi.org/10.4310/ATMP.2008.v12.n2.a3.
[58] Y. Ding, M. Han, and C. Rovelli, Phys. Rev. D **83**, 124020 (2011), 1011.2149.
[59] P. Dona, M. Fanizza, P. Martin-Dussaud, and S. Speziale (2020), 2011.13909.
[60] L. Scorzato, PoS **LATTICE2015**, 016 (2016), 1512.08039.
[61] P. F. Bedaque, EPJ Web Conf. **175**, 01020 (2018), 1711.05868.
[62] S. Bluecher, J. M. Pawlowski, M. Scherzer, M. Schlosser, I.-O. Stamatescu, S. Syrkowski, and F. P. Ziegler, SciPost Phys. **5**, 044 (2018), 1803.08418.
[63] A. Alexandru, G. Basar, P. F. Bedaque, G. W. Ridgway, and N. C. Warrington, JHEP **05**, 053 (2016), 1512.08764.
[64] T. TAKAGI, Japanese journal of mathematics :transactions and abstracts **1**, 83 (1924).
[65] J. A. Vrugt, Environmental Modelling and Software **75**, 273 (2016), ISSN 1364-8152, URL http://www.sciencedirect.com/science/article/pii/S1364815215300396.
[66] M. Han and H. Liu, *Analytic continuation of spin foam models* (2020), to appear.
[67] H. M. Haggard, M. Han, W. Kamiński, and A. Riello, Phys. Lett. **B752**, 258 (2016), 1509.00458.
[68] H. M. Haggard, M. Han, W. Kaminski, and A. Riello (2015), 1512.07690.
[69] M. Han, JHEP **01**, 065 (2016), 1509.00466.
[70] M. Han and T. Krajewski, Class.Quant.Grav. **31**, 015009 (2014), 1304.5626.
[71] L. Hörmander, *The Analysis of Linear Partial Differential Operators I*, vol. 256 of *Grundlehren der mathematischen Wissenschaften* (Springer Berlin Heidelberg, Berlin, Heidelberg, 1998), ISBN 978-3-642-96752-8.
[72] H. Zichang, *Spinfoam propagator asymptotic code*, https://gitee.com/ZCHuang1126/spin-foam-propagator-asymtotic.git (2020).
[73] A. Alexandru, G. Başar, P. F. Bedaque, H. Lamm, and S. Lawrence, Phys. Rev. D **98**, 034506 (2018), 1807.02027.
[74] V. Bonzom, Phys. Rev. **D80**, 064028 (2009), 0905.1501.
[75] F. Hellmann and W. Kaminski, JHEP **10**, 165 (2013), 1307.1679.
[76] C. Perini (2012), 1211.4807.
[77] C. Beem, T. Dimofte, and S. Pasquetti, JHEP **12**, 177 (2014), 1211.1986.
[78] H. Holzmann, Stochastics **77**, 371 (2005), https://doi.org/10.1080/17442500500190060, URL https://doi.org/10.1080/17442500500190060.
[79] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, The Journal of Chemical Physics **21**, 1087 (1953), https://doi.org/10.1063/1.1699114, URL https://doi.org/10.1063/1.1699114.

[80] H. Haario, E. Saksman, and J. Tamminen, Computational Statistics **14**, 375 (1999), ISSN 1613-9658, URL `https://doi.org/10.1007/s001800050022`.

[81] H. Haario, E. Saksman, and J. Tamminen, Bernoulli **7**, 223 (2001), URL `https://projecteuclid.org:443/euclid.bj/1080222083`.

[82] C. J. F. T. Braak, Statistics and Computing **16**, 239 (2006), ISSN 1573-1375, URL `https://doi.org/10.1007/s11222-006-8769-1`.