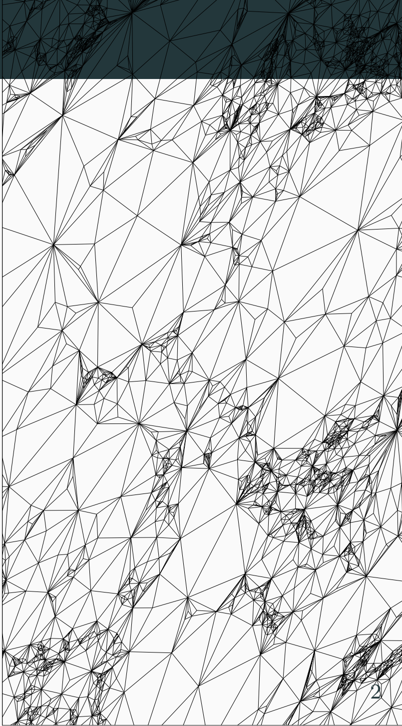
Matrix Models with Quantum Computers

Burak Varol

May 16, 2024

CQTA-DESY Zeuthen

Outline



Introduction and Theory

Discretization on Quantum Computer

Results

Conclusion and Outlook

Introduction and Theory

Introduction and Underlying Theory

- End Goal: Simulate Matrix Models of the kind:

$$e^{Z} = \int \prod_{\alpha=1}^{\nu-1} dM^{(\alpha)} e^{-\operatorname{Tr}\left[V(M^{(\alpha)})\right]}$$

with $V(M^{(\alpha)}) = \sum_{\alpha=1}^{\nu-1} V_\alpha M^{(\alpha)} - \sum_{\alpha=1}^{\nu-2} c_\alpha M^{(\alpha)} M^{(\alpha+1)}$

Introduction and Underlying Theory

- End Goal: Simulate Matrix Models of the kind:

$$e^Z = \int \prod_{\alpha=1}^{\nu-1} dM^{(\alpha)} e^{-\operatorname{Tr}\left[V(M^{(\alpha)})\right]}$$

  with $V(M^{(\alpha)}) = \sum_{\alpha=1}^{\nu-1} V_\alpha M^{(\alpha)} - \sum_{\alpha=1}^{\nu-2} c_\alpha M^{(\alpha)} M^{(\alpha+1)}$

- Tessellations of the Space

- End Goal: Simulate Matrix Models of the kind:

$$e^Z = \int \prod_{\alpha=1}^{\nu-1} dM^{(\alpha)} e^{-\operatorname{Tr}\left[V(M^{(\alpha)})\right]}$$

with $V(M^{(\alpha)}) = \sum_{\alpha=1}^{\nu-1} V_\alpha M^{(\alpha)} - \sum_{\alpha=1}^{\nu-2} c_\alpha M^{(\alpha)} M^{(\alpha+1)}$

- Tessellations of the Space
- Duality to 2D Quantum Gravity!

Connection to Pure Gravity - One Matrix Model

- Consider for $N \times N$ Hermitian Matrices:

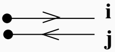
$$e^Z = \int e^{-N\,\mathrm{Tr}\left[M^2 - gM^3\right]} dM$$

Connection to Pure Gravity - One Matrix Model

- Consider for $N \times N$ Hermitian Matrices:

$$e^{Z} = \int e^{-N\,\mathrm{Tr}\left[M^2 - gM^3\right]} dM$$

- Fatgraphs: M_{ij}

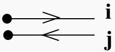


Connection to Pure Gravity - One Matrix Model

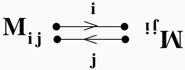
- Consider for $N \times N$ Hermitian Matrices:

$$e^{Z} = \int e^{-N\,\mathrm{Tr}\left[M^{2} - gM^{3}\right]} dM$$

- Fatgraphs: M_{ij}



- Propagator: $\langle M_{ij} M_{ji} \rangle$

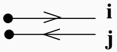


Connection to Pure Gravity - One Matrix Model

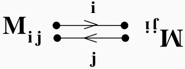
- Consider for $N \times N$ Hermitian Matrices:

$$e^{Z} = \int e^{-N\,\mathrm{Tr}\left[M^2 - gM^3\right]} dM$$

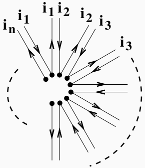
- Fatgraphs: M_{ij}



- Propagator: $\langle M_{ij} M_{ji} \rangle$

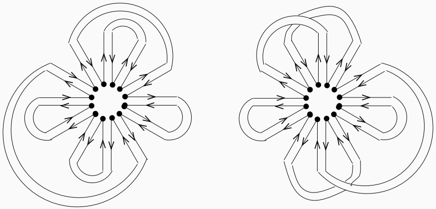


- Vertex: $\mathrm{Tr}[M^n]$



One Matrix Model

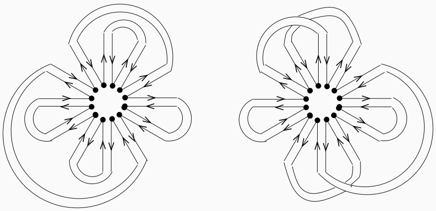
- $\langle \mathrm{Tr}\{M^n\} \rangle \leftrightarrow$ Contractions of Graphs:



(a) **(b)**

One Matrix Model

- $\langle \mathrm{Tr}\{M^n\} \rangle \leftrightarrow$ Contractions of Graphs:

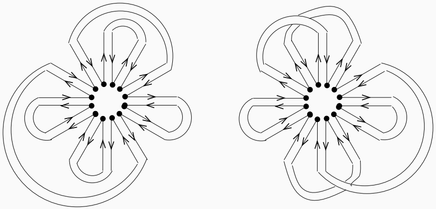


(a) **(b)**

- $Z = \sum_{\text{Connected } \Gamma} \frac{N^{V-E+F}}{|Aut(\Gamma)|} \prod_i g_i^{n_i}$

One Matrix Model

- $\langle \mathrm{Tr}\{M^n\} \rangle \leftrightarrow$ Contractions of Graphs:

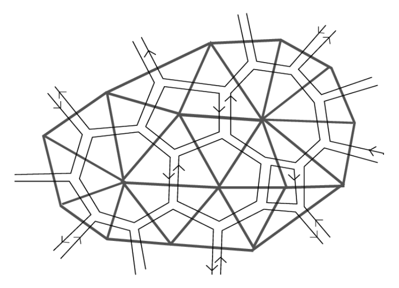


(a) **(b)**

- $Z = \sum_{\text{Connected } \Gamma} \frac{N^{V-E+F}}{|Aut(\Gamma)|} \prod_i g_i^{n_i}$
- Euler Characteristic: $\chi = V - E + F = 2 - 2h \longrightarrow$ Connection to Topology!

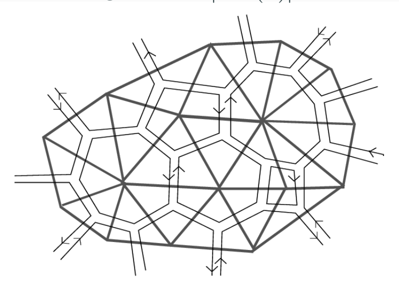
Connection to Gravity

- Assign Vertices $\leftrightarrow$ Faces: Dual Graph, $n_3 = A$



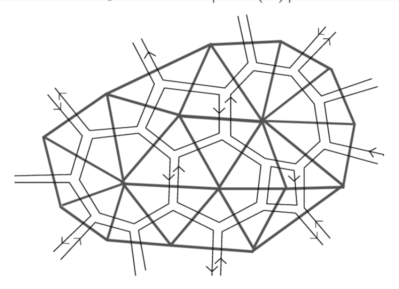
Connection to Gravity

- Assign Vertices $\leftrightarrow$ Faces: Dual Graph, $n_3 = A$
- $Z = \sum_{\text{Connected Triangulations}} \frac{N^{2-2h}}{|Aut(\Gamma)|} g_3^A$



Connection to Gravity

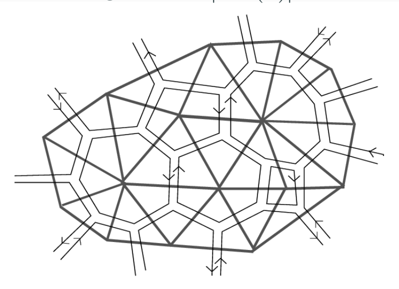
- Assign Vertices $\leftrightarrow$ Faces: Dual Graph, $n_3 = A$
- $Z = \sum_{\text{Connected Triangulations}} \frac{N^{2-2h}}{|Aut(\Gamma)|} g_3^A$



- $g = e^{-\Lambda}$ and $N = e^{\mathcal{N}} \implies Z = \sum_{\text{Connected Triangulations}} e^{-S_E}$

Connection to Gravity

- Assign Vertices $\leftrightarrow$ Faces: Dual Graph, $n_3 = A$
- $Z = \sum_{\text{Connected Triangulations}} \frac{N^{2-2h}}{|Aut(\Gamma)|} g_3^A$



- $g = e^{-\Lambda}$ and $N = e^{\mathcal{N}} \implies Z = \sum_{\text{Connected Triangulations}} e^{-S_E}$
- $S_E = \Lambda A + \mathcal{N}\chi = \Lambda \int_\Sigma \sqrt{g} d^2 + \mathcal{N} \int_\Sigma \sqrt{g} R d^2$ [3]

Discretization on Quantum Computer

Step 1: N= 1, Gaussian Distribution

- Gaussian Distribution:

$$\rho(x,\sigma) = e^{-\frac{x^2}{\sigma^2}}$$

Step 1: N= 1, Gaussian Distribution

- Gaussian Distribution:

$$\rho(x,\sigma) = e^{-\frac{x^2}{\sigma^2}}$$

- Discretize into intervals of length a, centre points $\pm a(\frac{1}{2}+k),\ k \in \mathbb{N}^+$

Step 1: N= 1, Gaussian Distribution

- Gaussian Distribution:

$$\rho(x,\sigma) = e^{-\frac{x^2}{\sigma^2}}$$

- Discretize into intervals of length a, centre points $\pm a(\frac{1}{2}+k),\ k \in \mathbb{N}^+$
- State: $|\pm k\rangle \equiv |k_\pm, k_0, k_1, k_2, \ldots\rangle$ s.t. $k = \sum_{m=0}^{\infty} k_m 2^m$, with $k_m \in \{0,1\}$

Step 1: N= 1, Gaussian Distribution

- Gaussian Distribution:

$$\rho(x,\sigma) = e^{-\frac{x^2}{\sigma^2}}$$

- Discretize into intervals of length a, centre points $\pm a(\frac{1}{2} + k),\ k \in \mathbb{N}^+$
- State: $|\pm k\rangle \equiv |k_\pm, k_0, k_1, k_2, \ldots\rangle$ s.t. $k = \sum_{m=0}^{\infty} k_m 2^m$, with $k_m \in \{0,1\}$
- $\hat{k} = \sigma_z^\pm \sum_{m=0}^{\infty} \hat{k}_m 2^m$, $\hat{k}_m := \frac{1-\sigma_z^{(m)}}{2} \implies \hat{k}\,|\pm k\rangle = \pm k\,|\pm k\rangle$

Step 1: N= 1, Gaussian Distribution

- Gaussian Distribution:

$$\rho(x,\sigma) = e^{-\frac{x^2}{\sigma^2}}$$

- Discretize into intervals of length a, centre points $\pm a(\frac{1}{2} + k),\ k \in \mathbb{N}^+$
- State: $|\pm k\rangle \equiv |k_\pm, k_0, k_1, k_2, \ldots\rangle$ s.t. $k = \sum_{m=0}^{\infty} k_m 2^m$, with $k_m \in \{0, 1\}$
- $\hat{k} = \sigma_z^\pm \sum_{m=0}^{\infty} \hat{k}_m 2^m$, $\hat{k}_m := \frac{1-\sigma_z^{(m)}}{2} \implies \hat{k}\,|\pm k\rangle = \pm k\,|\pm k\rangle$
- $\hat{x} = a\left(\frac{1}{2}\sigma_z^{(\pm)} + \hat{k}\right)$

Step 1: N= 1, Gaussian Distribution

- Gaussian Distribution:

$$\rho(x,\sigma) = e^{-\frac{x^2}{\sigma^2}}$$

- Discretize into intervals of length a, centre points $\pm a(\frac{1}{2}+k),\ k \in \mathbb{N}^+$
- State: $|\pm k\rangle \equiv |k_\pm, k_0, k_1, k_2, \ldots\rangle$ s.t. $k = \sum_{m=0}^{\infty} k_m 2^m$, with $k_m \in \{0,1\}$
- $\hat{k} = \sigma_z^{\pm} \sum_{m=0}^{\infty} \hat{k}_m 2^m$, $\hat{k}_m := \frac{1-\sigma_z^{(m)}}{2} \implies \hat{k}\,|\pm k\rangle = \pm k\,|\pm k\rangle$
- $\hat{x} = a\left(\frac{1}{2}\sigma_z^{(\pm)} + \hat{k}\right)$
- Hamiltonian: $H = \hat{x}^2$ with $H\,|k\rangle = a^2(\frac{1}{2}+k)^2\,|k\rangle$

Step 1: N= 1, Gaussian Distribution

- Create Thermal State st. $\beta = \frac{1}{\sigma^2}$:

$$\rho_\beta = e^{-\beta H}$$

Step 1: N= 1, Gaussian Distribution

- Create Thermal State st. $\beta = \frac{1}{\sigma^2}$:

$$\rho_\beta = e^{-\beta H}$$

- Integral approximated by:

$$\int dx \rho(x, \sigma) \approx \sum_k a\, e^{-\frac{a^2 k^2}{\sigma^2}} = a\, \mathrm{Tr} \rho_{\frac{1}{\sigma^2}}$$

Step 1: N= 1, Gaussian Distribution

- Create Thermal State st. $\beta = \frac{1}{\sigma^2}$:

$$\rho_\beta = e^{-\beta H}$$

- Integral approximated by:

$$\int dx \rho(x, \sigma) \approx \sum_k a\, e^{-\frac{a^2 k^2}{\sigma^2}} = a\,\mathrm{Tr} \rho_{\frac{1}{\sigma^2}}$$

- Expectation Values:

$$\int dx f(x) \rho(x, \sigma) \approx \sum_k a\, f(ak) e^{-\frac{a^2 k^2}{\sigma^2}} = a \mathrm{Tr} f(\hat{x}) \rho_{\frac{1}{\sigma^2}}$$

Step 2: One Matrix Model

- Ensemble of Hermitian Random Matrices with $V(M) = M^2 + \sum_{k\geq 3} \alpha_k M^k$. How to discretisize?

$$Z = \int dM \exp\{-tr(V(M))\}$$

Step 2: One Matrix Model

- Ensemble of Hermitian Random Matrices with $V(M) = M^2 + \sum_{k \geq 3} \alpha_k M^k$. How to discretisize?

$$Z = \int dM \exp\{-tr(V(M))\}$$

- Goal: Parametrise by Eigenvalues λ_k and Unitary Matrices U: $M = U^\dagger \Lambda U$

Step 2: One Matrix Model

- Ensemble of Hermitian Random Matrices with $V(M) = M^2 + \sum_{k\geq 3} \alpha_k M^k$. How to discretisize?

$$Z = \int dM \exp\{-tr(V(M))\}$$

- Goal: Parametrise by Eigenvalues λ_k and Unitary Matrices U: $M = U^\dagger \Lambda U$
- Forbenius norm:
  $\|M\|^2 = tr(MM^\dagger) \longrightarrow tr(d\Lambda^2 - [\Lambda, dT][\Lambda, dT])$,
  T Hermitian

One Matrix Model: Reducing to $\mathbb{R}^n$

- $tr(dMdM^\dagger) = \sum_k (d\lambda_k)^2 - \sum_{i,j}(\lambda_i - \lambda_j)^2 |dT_{ij}|^2$

One Matrix Model: Reducing to $\mathbb{R}^n$

- $tr(dMdM^\dagger) = \sum_k (d\lambda_k)^2 - \sum_{i,j}(\lambda_i - \lambda_j)^2 |dT_{ij}|^2$
- Read the Jacobi Factor:

$$G = \prod_{i \neq j} (\lambda_i - \lambda_j)^2$$

$$\sqrt{G} = \prod_{i<j} (\lambda_i - \lambda_j)^2 = \Delta(\Lambda) \quad (= \text{Vandermonde Determinant})$$

One Matrix Model: Reducing to $\mathbb{R}^n$

- $tr(dMdM^\dagger) = \sum_k (d\lambda_k)^2 - \sum_{i,j} (\lambda_i - \lambda_j)^2 |dT_{ij}|^2$
- Read the Jacobi Factor:

$$G = \prod_{i \neq j} (\lambda_i - \lambda_j)^2$$

$$\sqrt{G} = \prod_{i<j} (\lambda_i - \lambda_j)^2 = \Delta(\Lambda) \quad (= \text{Vandermonde Determinant})$$

- Tranformation:

$$Z = \int \prod_i^N d\lambda_i \Delta(\Lambda)^2 \exp\left\{(-\sum_i V(\lambda_i))\right\}$$

One Matrix Model: Implementation

- Define: $|\pm\lambda\rangle \equiv |\lambda_\pm, \lambda_0, \lambda_1, \lambda_2, \ldots\rangle$

$$\lambda = \sigma_z^{\pm} \sum_{m=0}^{\infty} \lambda_m 2^m \quad \text{with} \quad \lambda_m := \frac{1 - \sigma_z^{(m)}}{2}$$

$$\hat{x}_\lambda = a\left(\frac{1}{2}\sigma_z^{(\pm)} + \lambda\right)$$

One Matrix Model: Implementation

- Define: $|\pm\lambda\rangle \equiv |\lambda_\pm, \lambda_0, \lambda_1, \lambda_2, \ldots\rangle$

$$\lambda = \sigma_z^\pm \sum_{m=0}^{\infty} \lambda_m 2^m \quad \text{with} \quad \lambda_m := \frac{1-\sigma_z^{(m)}}{2}$$

$$\hat{x}_\lambda = a\left(\frac{1}{2}\sigma_z^{(\pm)} + \lambda\right)$$

- For an $N \times N$ random matrix, we need $D \times N$ qubits :

$$|\psi\rangle = \underbrace{\left|\lambda_\pm^{(1)}, \lambda_0^{(1)}, \lambda_1^{(1)}, \lambda_2^{(1)}, \ldots\right\rangle}_{\text{D Qubits}} \otimes \left|\lambda_\pm^{(2)}, \lambda_0^{(2)}, \lambda_1^{(2)}, \lambda_2^{(2)}, \ldots\right\rangle \otimes \ldots$$

$$\cdots \otimes \left|\lambda_\pm^{(N)}, \lambda_0^{(N)}, \lambda_1^{(N)}, \lambda_2^{(N)}, \ldots\right\rangle$$

One Matrix Model: Implementation

- Define: $|\pm\lambda\rangle \equiv |\lambda_{\pm}, \lambda_0, \lambda_1, \lambda_2, \ldots\rangle$

$$\lambda = \sigma_z^{\pm} \sum_{m=0}^{\infty} \lambda_m 2^m \quad \text{with} \quad \lambda_m := \frac{1 - \sigma_z^{(m)}}{2}$$

$$\hat{x}_\lambda = a\left(\frac{1}{2}\sigma_z^{(\pm)} + \lambda\right)$$

- For an $N \times N$ random matrix, we need $D \times N$ qubits :

$$|\psi\rangle = \underbrace{\left|\lambda_{\pm}^{(1)}, \lambda_0^{(1)}, \lambda_1^{(1)}, \lambda_2^{(1)}, \ldots\right\rangle}_{\text{D Qubits}} \otimes \left|\lambda_{\pm}^{(2)}, \lambda_0^{(2)}, \lambda_1^{(2)}, \lambda_2^{(2)}, \ldots\right\rangle \otimes \ldots$$

$$\cdots \otimes \left|\lambda_{\pm}^{(N)}, \lambda_0^{(N)}, \lambda_1^{(N)}, \lambda_2^{(N)}, \ldots\right\rangle$$

- Hamiltonian: $H = \sum_{i=0}^{N-1} \mathbb{1}^{\otimes i} \otimes V(\hat{x}_\lambda) \otimes \mathbb{1}^{\otimes((N-1)-i)}$

Step 3: Multi-Matrix Model (For Future)

- Consider with
  $V(M^{(\alpha)}) = \sum_{\alpha=1}^{\nu-1} V_\alpha M^{(\alpha)} - \sum_{\alpha=1}^{\nu-2} c_\alpha M^{(\alpha)} M^{(\alpha+1)}$

$$Z = \int \prod_{\alpha=1}^{\nu-1} dM^{(\alpha)} e^{-\operatorname{Tr}\left[V(M^{(\alpha)})\right]}$$

Step 3: Multi-Matrix Model (For Future)

- Consider with
  $V(M^{(\alpha)}) = \sum_{\alpha=1}^{\nu-1} V_\alpha M^{(\alpha)} - \sum_{\alpha=1}^{\nu-2} c_\alpha M^{(\alpha)} M^{(\alpha+1)}$

$$Z = \int \prod_{\alpha=1}^{\nu-1} dM^{(\alpha)} e^{-\operatorname{Tr}\left[V(M^{(\alpha)})\right]}$$

- For $\alpha = 2$:

$$Z = \int dM_1 dM_2 e^{-[V(M_1)+V(M_2)-\beta tr(M_1 M_2)]}$$

Step 3: Multi-Matrix Model (For Future)

- Consider with
  $V(M^{(\alpha)}) = \sum_{\alpha=1}^{\nu-1} V_\alpha M^{(\alpha)} - \sum_{\alpha=1}^{\nu-2} c_\alpha M^{(\alpha)} M^{(\alpha+1)}$

$$Z = \int \prod_{\alpha=1}^{\nu-1} dM^{(\alpha)} e^{-\operatorname{Tr}\left[V(M^{(\alpha)})\right]}$$

- For $\alpha = 2$:

$$Z = \int dM_1 dM_2 e^{-[V(M_1)+V(M_2)-\beta tr(M_1 M_2)]}$$

- Transform:

$$Z \propto \int \prod^{N} d\lambda_{1,i} d\lambda_{2,j} \Delta^2(\Lambda_1)\Delta^2(\Lambda_2) e^{[-(V(\Lambda_1)+V(\Lambda_2))]} I(\Lambda_1, \Lambda_2)$$

with $I(M_1, M_2; \beta) \propto \beta^{\frac{-N(N-1)}{2}} \frac{det(e^{(\beta\lambda_{1,i}\lambda_{2,j})})}{\Delta(\Lambda_1)\Delta(\Lambda_2)}$ [4]

Multi Matrix Model

- Complete transformation [2]:

$$Z = \int \prod_{\alpha=1...\nu-1; i=1...N} d\lambda_i^{(\alpha)} \Delta(\Lambda^{(1)}) e^{-\sum_i V(\lambda_i^{(\alpha)})} \Delta(\Lambda^{(\nu-1)}).$$

Multi Matrix Model

- Complete transformation [2]:

$$Z = \int \prod_{\alpha=1\ldots\nu-1; i=1\ldots N} d\lambda_i^{(\alpha)} \Delta(\Lambda^{(1)}) e^{-\sum_i V(\lambda_i^{(\alpha)})} \Delta(\Lambda^{(\nu-1)}).$$

- Adds Extra Features: Matter [2] , Casual Structure [1] ...

Multi Matrix Model

- Complete transformation [2]:

$$Z = \int \prod_{\alpha=1...\nu-1; i=1...N} d\lambda_i^{(\alpha)} \Delta(\Lambda^{(1)}) e^{-\sum_i V(\lambda_i^{(\alpha)})} \Delta(\Lambda^{(\nu-1)}).$$

- Adds Extra Features: Matter [2] , Casual Structure [1] ...
- Implementation similar

Creating Thermal Distributions: General Hamiltonians

- Maximally mixed state:

$$\rho^{\max} = \mathbb{1} = \sum_{k=-2^\Lambda}^{2^\Lambda} |k\rangle \langle k|$$

Creating Thermal Distributions: General Hamiltonians

- Maximally mixed state:

$$\rho^{\max} = \mathbb{1} = \sum_{k=-2^{\Lambda}}^{2^{\Lambda}} |k\rangle \langle k|$$

- Imaginary Time Evolution:

$$\rho_\beta = \frac{e^{-\frac{\beta}{2}H}\mathbb{1}e^{-\frac{\beta}{2}H}}{Tr[e^{-\beta H}]}$$

Creating Thermal Distributions: General Hamiltonians

- Maximally mixed state:

$$\rho^{\max} = \mathbb{1} = \sum_{k=-2^{\Lambda}}^{2^{\Lambda}} |k\rangle \langle k|$$

- Imaginary Time Evolution:

$$\rho_{\beta} = \frac{e^{-\frac{\beta}{2}H} \mathbb{1} e^{-\frac{\beta}{2}H}}{Tr[e^{-\beta H}]}$$

- Problem: $U(\theta)\mathbb{1}U(\theta)^{\dagger} = \mathbb{1}$

Creating Thermal Distributions: General Hamiltonians

- Maximally mixed state:

$$\rho^{\max} = \mathbb{1} = \sum_{k=-2^{\Lambda}}^{2^{\Lambda}} |k\rangle \langle k|$$

- Imaginary Time Evolution:

$$\rho_{\beta} = \frac{e^{-\frac{\beta}{2}H}\mathbb{1}e^{-\frac{\beta}{2}H}}{Tr[e^{-\beta H}]}$$

- Problem: $U(\theta)\mathbb{1}U(\theta)^{\dagger} = \mathbb{1}$
- Maximally Entangled State:

$$|\psi(0)\rangle = V(\theta(\vec{0}))\,|0\rangle^{\otimes 2N} = \left|\phi^{+}\right\rangle^{\otimes N}$$

Creating Thermal Distributions: General Hamiltonians

- Maximally mixed state:

$$\rho^{\max} = \mathbb{1} = \sum_{k=-2^{\Lambda}}^{2^{\Lambda}} |k\rangle \langle k|$$

- Imaginary Time Evolution:

$$\rho_{\beta} = \frac{e^{-\frac{\beta}{2}H}\mathbb{1}e^{-\frac{\beta}{2}H}}{Tr[e^{-\beta H}]}$$

- Problem: $U(\theta)\mathbb{1}U(\theta)^{\dagger} = \mathbb{1}$
- Maximally Entangled State:

$$|\psi(0)\rangle = V(\theta(\vec{0}))\,|0\rangle^{\otimes 2N} = \left|\phi^{+}\right\rangle^{\otimes N}$$

- $\rho^{Gibbs} = \mathrm{Tr}_B\,|\psi\rangle\langle\psi| \approx \frac{e^{-\beta H}}{Z}$

Generating Thermal Distributions: Diagonal Hamiltonian

- Disadvantage: Double Qubits.

Generating Thermal Distributions: Diagonal Hamiltonian

- Disadvantage: Double Qubits.
- Diagonal Hamiltonian:

$$e^{\beta H}\frac{1}{\sqrt{2^N}}\left|000...\right\rangle + \left|100...\right\rangle + ... \left|111...\right\rangle)$$
$$= \frac{1}{\sqrt{2^N}}(e^{\beta E_{000...}}\left|000...\right\rangle + e^{\beta E_{100...}}\left|100...\right\rangle + ... + e^{\beta E_{111..}}\left|111...\right\rangle)$$

Generating Thermal Distributions: Diagonal Hamiltonian

- Disadvantage: Double Qubits.
- Diagonal Hamiltonian:

$$e^{\beta H}\frac{1}{\sqrt{2^N}}\left|000...\right\rangle+\left|100...\right\rangle+...\left|111...\right\rangle)$$
$$=\frac{1}{\sqrt{2^N}}(e^{\beta E_{000...}}\left|000...\right\rangle+e^{\beta E_{100...}}\left|100...\right\rangle+...+e^{\beta E_{111..}}\left|111...\right\rangle)$$

- Works with pure state!

Variational Imaginary Time Evolution (VarQITE)

- Schrödinger Equation:

$$\frac{d\,|\psi(\tau)\rangle}{d\tau} = -(H - E_\tau)\,|\psi(\tau)\rangle$$

Variational Imaginary Time Evolution (VarQITE)

- Schrödinger Equation:

$$\frac{d\left|\psi(\tau)\right\rangle}{d\tau} = -(H - E_\tau)\left|\psi(\tau)\right\rangle$$

- Evolution

$$\left|\psi(\tau)\right\rangle = \frac{1}{\sqrt{\left\langle\psi(0)\right| e^{-2H\tau}\left|\psi(0)\right\rangle}} e^{-H\tau}\left|\psi(0)\right\rangle$$

Variational Imaginary Time Evolution (VarQITE)

- Schrödinger Equation:

$$\frac{d\left|\psi(\tau)\right\rangle}{d\tau} = -(H - E_\tau)\left|\psi(\tau)\right\rangle$$

- Evolution

$$\left|\psi(\tau)\right\rangle = \frac{1}{\sqrt{\left\langle\psi(0)\right| e^{-2H\tau}\left|\psi(0)\right\rangle}} e^{-H\tau}\left|\psi(0)\right\rangle$$

- Approximate:

$$\left|\phi(\tau)\right\rangle = V(\theta(\vec{\tau}))\left|\phi(0)\right\rangle \approx \left|\psi(\tau)\right\rangle \text{ [6]}$$

VarQITE

- After a timestep:

$$|\psi(\tau + \delta\tau)\rangle \approx \left|\phi(\theta(\vec{\tau}))\right\rangle - \delta\tau(H - E_\tau)\left|\phi(\theta(\vec{\tau}))\right\rangle$$

VarQITE

- After a timestep:

$$|\psi(\tau + \delta\tau)\rangle \approx \left|\phi(\theta(\vec{\tau}))\right\rangle - \delta\tau(H - E_\tau)\left|\phi(\theta(\vec{\tau}))\right\rangle$$

- Parameter Space:

$$|\phi(\tau + \delta\tau)\rangle \approx \left|\phi(\theta(\vec{\tau}))\right\rangle + \sum_j \frac{\partial\left|\phi(\theta(\vec{\tau}))\right\rangle}{\partial\theta_j}\delta\theta_j$$

VarQITE

- After a timestep:

$$|\psi(\tau + \delta\tau)\rangle \approx \left|\phi(\theta(\vec{\tau}))\right\rangle - \delta\tau(H - E_\tau)\left|\phi(\theta(\vec{\tau}))\right\rangle$$

- Parameter Space:

$$|\phi(\tau + \delta\tau)\rangle \approx \left|\phi(\theta(\vec{\tau}))\right\rangle + \sum_j \frac{\partial \left|\phi(\theta(\vec{\tau}))\right\rangle}{\partial\theta_j}\delta\theta_j$$

- Approximation reduces to:

$$\sum_j \frac{\partial \left|\phi(\theta(\vec{\tau}))\right\rangle}{\partial\theta_j}\delta\theta_j \approx -\delta\tau(H - E_\tau)\left|\phi(\theta(\vec{\tau}))\right\rangle \text{ [5]}$$

VarQITE

- After a timestep:

$$|\psi(\tau+\delta\tau)\rangle \approx \left|\phi(\theta(\vec{\tau}))\right\rangle - \delta\tau(H-E_\tau)\left|\phi(\theta(\vec{\tau}))\right\rangle$$

- Parameter Space:

$$|\phi(\tau+\delta\tau)\rangle \approx \left|\phi(\theta(\vec{\tau}))\right\rangle + \sum_j \frac{\partial \left|\phi(\theta(\vec{\tau}))\right\rangle}{\partial\theta_j}\delta\theta_j$$

- Approximation reduces to:

$$\sum_j \frac{\partial \left|\phi(\theta(\vec{\tau}))\right\rangle}{\partial\theta_j}\delta\theta_j \approx -\delta\tau(H-E_\tau)\left|\phi(\theta(\vec{\tau}))\right\rangle \text{ [5]}$$

- Minimize the distance with $\left\|\left|\phi(\theta(\vec{\tau}))\right\rangle\right\| = 1$

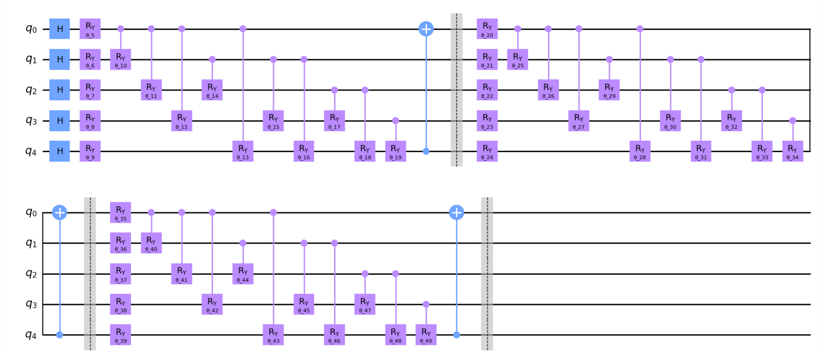
Results

Ansatz

- Generate Combinations:

$$H^{\otimes n}\left|0\right\rangle^{\otimes n} = \sum_{i=0}^{2^n-1} \left|i\right\rangle = \frac{1}{\sqrt{2^n}} \left|000...\right\rangle + \left|100...\right\rangle + ... \left|111...\right\rangle$$

- Parameterized Circuit:



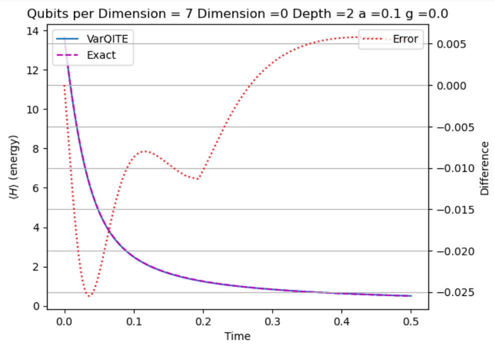
What are we Calculating?

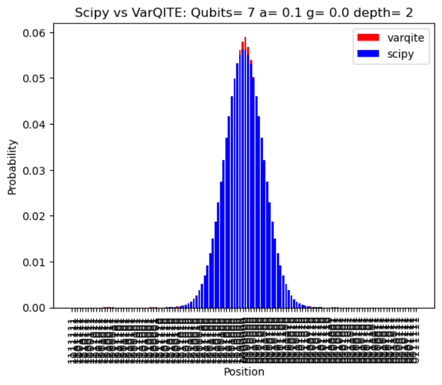
- Transformed:

$$Z = \int \prod_{i}^{N} d\lambda_i \Delta(\Lambda)^2 \exp\left\{(-\sum_i V(\lambda_i))\right\}$$

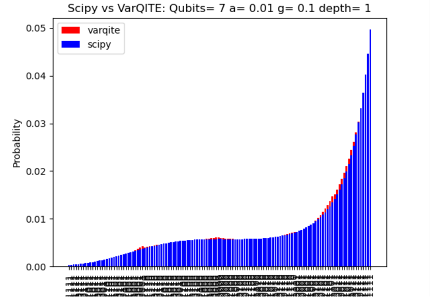
- $Z_{phys} \longrightarrow Z_{QC} = \left\langle \Delta(\Lambda)^2 \right\rangle$
- Expectation Values: $\langle f(ak) \rangle_{phys} \longrightarrow \left\langle \Delta(\Lambda)^2 f(ak) \right\rangle$

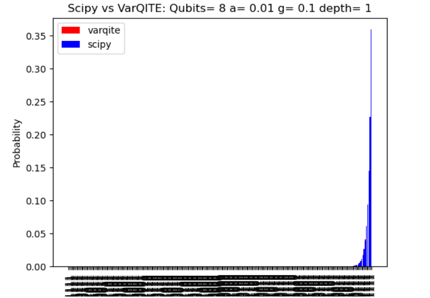
Gaussian Distribution $V(x) = x^2$

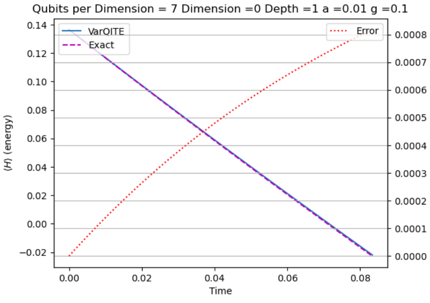


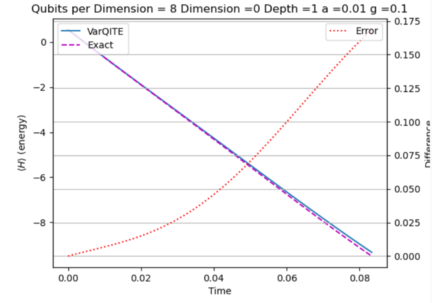


with gx^3 Term

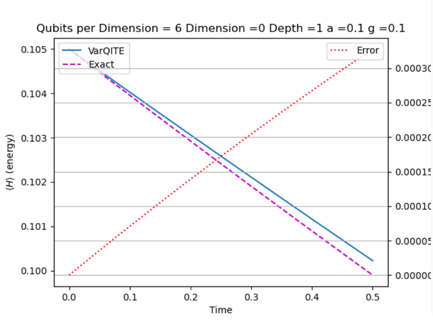


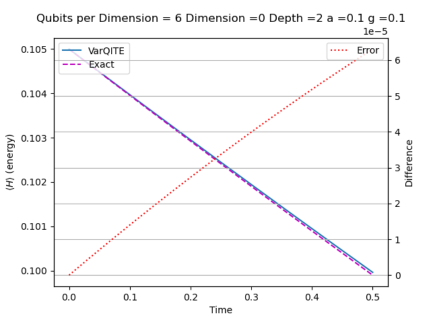


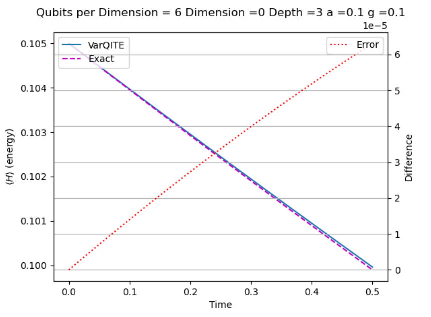




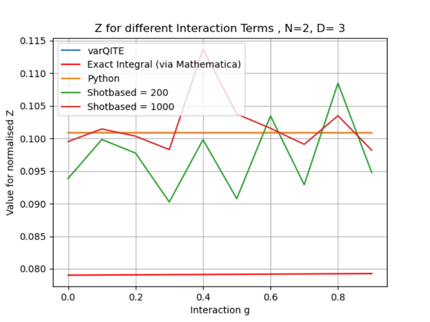
with gx^3 Term

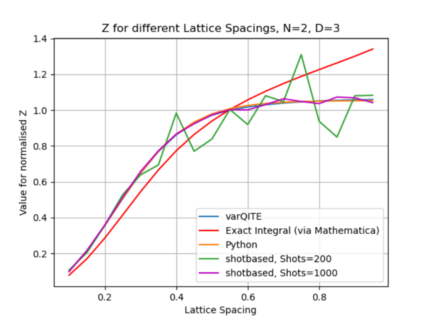


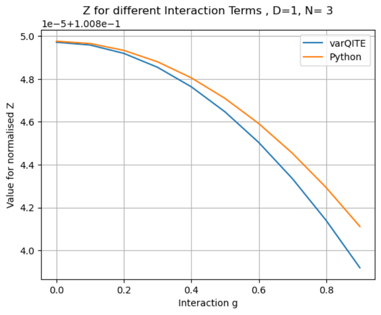




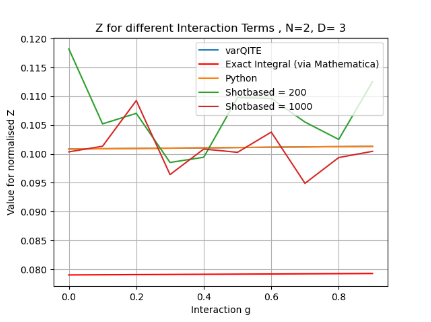
One Matrix Model: $V(M) = M^2 - gM^3$

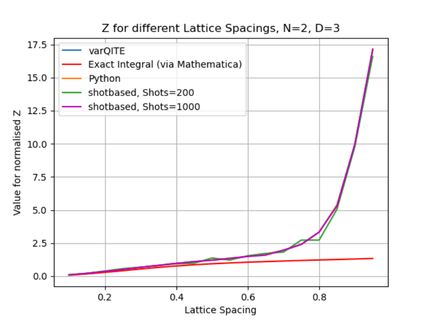


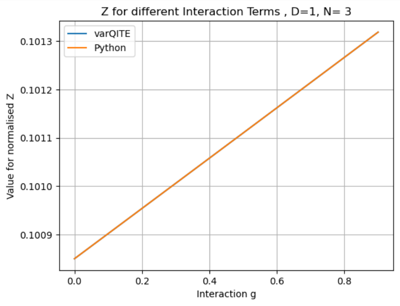




One Matrix Model: with $V(M) = M^2 - gM^4$







Conclusion and Outlook

Conclusion

- Simulating Matrix Models $\leftrightarrow$ 2D Quantum Gravity

Conclusion

- Simulating Matrix Models $\leftrightarrow$ 2D Quantum Gravity
- Discretization needs $D \times N$ Qubits.

Conclusion

- Simulating Matrix Models $\leftrightarrow$ 2D Quantum Gravity
- Discretization needs $D \times N$ Qubits.
- VarQITE for Thermal States

Conclusion

- Simulating Matrix Models $\leftrightarrow$ 2D Quantum Gravity
- Discretization needs $D \times N$ Qubits.
- VarQITE for Thermal States
- Results $\implies$ Method Scalable for Higher Qubits

Outlook

- Efficiency Calculation: Is this really worth doing? Comparison with Monte Carlo.

Outlook

- Efficiency Calculation: Is this really worth doing? Comparison with Monte Carlo.
- Is there a better (more quantum) algorithm?

Outlook

- Efficiency Calculation: Is this really worth doing? Comparison with Monte Carlo.
- Is there a better (more quantum) algorithm?
- Put to real Quantum Computer.

Outlook

- Efficiency Calculation: Is this really worth doing? Comparison with Monte Carlo.
- Is there a better (more quantum) algorithm?
- Put to real Quantum Computer.
- Actually doing physics:
  - Find and Calculate interesting Expectation Values (eg. $\langle \mathrm{Tr}\{M^n\}\rangle$

Outlook

- Efficiency Calculation: Is this really worth doing? Comparison with Monte Carlo.
- Is there a better (more quantum) algorithm?
- Put to real Quantum Computer.
- Actually doing physics:
  - Find and Calculate interesting Expectation Values (eg. $\langle \mathrm{Tr}\{M^n\} \rangle$
  - Large N Limit

Outlook

- Efficiency Calculation: Is this really worth doing? Comparison with Monte Carlo.
- Is there a better (more quantum) algorithm?
- Put to real Quantum Computer.
- Actually doing physics:
  - Find and Calculate interesting Expectation Values (eg. $\langle \mathrm{Tr}\{M^n\} \rangle$
  - Large N Limit
  - Double Scaling Limit $N \to \infty$, $g \to 0$

Outlook

- Efficiency Calculation: Is this really worth doing? Comparison with Monte Carlo.
- Is there a better (more quantum) algorithm?
- Put to real Quantum Computer.
- Actually doing physics:
  - Find and Calculate interesting Expectation Values (eg. $\langle \mathrm{Tr}\{M^n\} \rangle$
  - Large N Limit
  - Double Scaling Limit $N \to \infty$, $g \to 0$
  - Investigate different Coupling Constants g

Outlook

- Efficiency Calculation: Is this really worth doing? Comparison with Monte Carlo.
- Is there a better (more quantum) algorithm?
- Put to real Quantum Computer.
- Actually doing physics:
    - Find and Calculate interesting Expectation Values (eg. $\langle \mathrm{Tr}\{M^n\} \rangle$
    - Large N Limit
    - Double Scaling Limit $N \to \infty$, $g \to 0$
    - Investigate different Coupling Constants g
    - Different Matrix Models

Outlook

- Efficiency Calculation: Is this really worth doing? Comparison with Monte Carlo.
- Is there a better (more quantum) algorithm?
- Put to real Quantum Computer.
- Actually doing physics:
    - Find and Calculate interesting Expectation Values (eg. $\langle \mathrm{Tr}\{M^n\} \rangle$
    - Large N Limit
    - Double Scaling Limit $N \to \infty$, $g \to 0$
    - Investigate different Coupling Constants g
    - Different Matrix Models
    - Search for new Dualities

References

Juan L. A. Abranches, Antonio D. Pereira, and Reiko Toriumi.
Dually weighted multi-matrix models as a path to causal gravity-matter systems.
10 2023.

P Di Francesco.
2d quantum gravity, matrix models and graph combinatorics, 2004.

P Di Francesco, P Ginsparg, and J Zinn-Justin.
2d gravity and random matrices, 1994.

C. Itzykson and J. B. Zuber.
The planar approximation. ii.
Journal of Mathematical Physics, 21:411–421, 1979.

Xiao Yuan, Suguru Endo, Qi Zhao, Ying Li, and Simon Benjamin.
Theory of variational quantum simulation.
12 2018.

Christa Zoufal, Aurélien Lucchi, and Stefan Woerner.
Variational quantum boltzmann machines.
6 2020.