July 2020

ENTANGLEMENT ENTROPY IN 1+1d CRITICAL SYSTEMS

Robin Plantey

Laboratoire de physique théorique et hautes énergies, Sorbonne Université

Master Thesis supervised by Yacine Ikhlef & Benoit Estienne

(30 credits)

Contents

1 Introduction

In this report I present the work I did during my internship at the Laboratoire pour la Physique Théorique et Hautes Énergies (LPTHE) at Sorbonne Université. This internship lasted from March to July 2020 and was done under the supervision of Yacine Ikhlef and Benoit Estienne. The topic of the project was Conformal Field Theory (CFT) in the context of 1+1d quantum critical systems. Out of several project ideas I was given, I chose to investigate analytical methods used to compute entanglement entropies. The first half of the internship was spent learning CFT and writing code to simulate the Ising model. After getting used to the CFT formalism I was able to do several analytical computations. Using the program I had written I could compare the CFT description with numerical simulations.

The important theoretical notions encountered during this project are introduced in Sect. 2. In Sect. 3, analytical computations of entanglement entropy in CFT are presented. Numerical computations in the Ising model are described in Sect. 4. We conclude in Sect. 5.

2 Theoretical background

2.1 Notions of Conformal Field Theory

We briefly describe the notions of 2d conformal field theory relevant to this project. [1] and [2] were used as references during the whole project. This section will consist of a compilation of fundamental results from these references described as I understand them. It also serves the purpose of introducing the notation used in the rest of the report.

The conformal group consists of all the angle preserving transformations. In 2 dimensions, conformal covariance is very strong and imposes tight constraints on a field theory. In fact, it leads to large simplifications which make some analytical computations possible.

The building blocks of a CFT are its primary fields which are those fields that transform as

$$\phi'(w,\bar{w}) = \left(\frac{dw}{dz}\right)^{-h}\left(\frac{d\bar{w}}{d\bar{z}}\right)^{-\bar{h}}\phi(z,\bar{z}) \tag{2.1}$$

under any conformal transformation $w = w(z)$. The numbers $h, \bar{h}$ are the conformal dimensions of the primary operator ϕ. In this project we only considered scalar operators with $h = \bar{h}$. For correlation functions of primary field, conformal covariance means that under a conformal transformation $w = w(z)$

$$\langle\phi_1(w_1,\bar{w}_1)...\phi_n(w_n,\bar{w}_n)\rangle_{\mathcal{D}'} = \prod_{i=1}^{n}\left(\frac{dw_i}{dz_i}\right)^{-h_i}\left(\frac{d\bar{w}_i}{d\bar{z}_i}\right)^{-\bar{h}_i}\langle\phi_1(z_1,\bar{z}_1)...\phi_n(z_n,\bar{z}_n)\rangle_{\mathcal{D}} \tag{2.2}$$

where the subscripts $\mathcal{D}$ and $\mathcal{D}'$ indicate the geometry on which the expectation value is taken. Indeed, in CFT one adopts the active point of view and thinks of a conformal

transformation as a deformation of the surface. The functional form of 2-point functions of primary operators is severely constrained by Lorentz invariance and by the scaling implied by Eq.(2.2). On the complex plane, 2-point functions of primary operators are of the form

$$\langle \phi(z_1)\phi(z_2)\rangle_{\mathbb{C}} = |z_1 - z_2|^{-4h}. \tag{2.3}$$

In this project we often considered periodic boundary conditions and hence worked with a cylindrical geometry. Mapping the plane to a cylinder of circumference L, one can transform Eq.(2.3) using Eq.(2.2) and find

$$\langle \phi(w_1)\phi(w_2)\rangle_{\mathbb{R}\times S^1} = \left[\frac{L}{\pi}\sinh\frac{\pi(w_1 - w_2)}{L}\right]^{-2h}\left[\frac{L}{\pi}\sinh\frac{\pi(\bar{w}_1 - \bar{w}_2)}{L}\right]^{-2h}. \tag{2.4}$$

The energy momentum tensor is a very important non-primary operator, it transforms as

$$T'(w) = \left(\frac{dw}{dz}\right)^{-2}\left(\frac{d\bar{w}}{d\bar{z}}\right)^{-2}T(z) + \frac{c}{12}\{w, z\} \tag{2.5}$$

where $\{w, z\} = \frac{\partial_z^3 w}{\partial_z w} - \frac{3}{2}\left(\frac{\partial_z^2 w}{\partial_z w}\right)^2$ is the so-called Schwarzian derivative and the number c, called the central charge, is a characteristic of the theory.

The local operators of a CFT form an algebra whose closure under multiplication is captured by the Operator Product Expansion (OPE)

$$\mathcal{O}_i(w)\mathcal{O}_j(z) = \sum_k C_{ijk}(w - z)O_k(z). \tag{2.6}$$

For example, the OPE between the energy momentum tensor and a primary operator ϕ of dimension h

$$T(w)\phi(z, \bar{z}) = \frac{h}{(w - z)^2}\phi(z, \bar{z}) + \frac{1}{w - z}\partial\phi(z, \bar{z}) + \text{reg.} \tag{2.7}$$

is very useful.

The structure of the Hilbert space of the theory can be derived from an extension of the conformal algebra, the Virasoro algebra (Vir)

$$[L_m, L_n] = (m - n)L_{m+n} + \frac{c}{12}m(m^2 - 1)\delta_{m+n,0}. \tag{2.8}$$

It is the algebra of the modes of the energy momentum tensor

$$T(z) = \sum_{n\in\mathbb{Z}} L_n\, z^{-n-2}. \tag{2.9}$$

There is another copy of this algebra from the anti-holomorphic energy momentum tensor $\bar{T}(\bar{z})$. The Hilbert space is constructed out of representations of $\text{Vir}\times\bar{\text{Vir}}$. A representation

of Vir is built as follows. For any primary operator of conformal dimension h there is a primary state $|h\rangle = \phi(0,0)|0\rangle$ such that

$$\begin{aligned} L_0|h\rangle &= h|h\rangle \\ L_n|h\rangle &= 0 \quad \forall n > 0. \end{aligned} \tag{2.10}$$

In general to each operators $\mathcal{O}$ corresponds a state $\mathcal{O}(0,0)|0\rangle$ and vice versa. Now, given such a primary state, the states

$$\prod_k L_{-n_k}|h\rangle\,, \quad n_k \geq 0 \tag{2.11}$$

form, by construction, a representation $\mathcal{V}_h$ consisting of an infinite tower of states of dimensions $h+N$, $N \in \mathbb{N}$. A null vector at level N is a particular superposition of states at that level that is itself a primary state i.e.

$$\begin{aligned} |\chi\rangle &= (\#L_{-N} + ... + \#L_{-1}^N)|h\rangle \\ L_0|\chi\rangle &= (h+N)|\chi\rangle \\ L_n|\chi\rangle &= 0 \quad \forall n > 0. \end{aligned} \tag{2.12}$$

The so-called Kac parameterization gives the conformal dimensions of the null states at level N. If the central charge is written $c = 1 - 6(b^{-1} - b)^2$ then the null states at level N have dimensions

$$h_{rs} = \frac{(rb^{-1} - sb) - (b^{-1} - b)^2}{4} \tag{2.13}$$

with r, s positive integers satisfying $rs = N$. The scalar product of a null state with any state is easily shown to vanish. This means that any correlation function involving a null state (or equivalently a null operator) vanishes, hence these states are decoupled from the theory. In addition, null states must be quotiented out of $\mathcal{V}_h$ in order for it to be irreducible (for each null state there is an invariant subspace). The Hilbert space is a sum of irreducible representations of Vir $\times$ $\bar{\text{Vir}}$, one for each primary operator ϕ_k

$$H = \oplus_k \frac{\mathcal{V}_{h_k}}{H_{k,\text{null}}} \otimes \frac{\bar{\mathcal{V}}_{h_k}}{\bar{H}_{k,\text{null}}}. \tag{2.14}$$

This knowledge about the structure of the Hilbert space can be used to decompose 4-point functions $\langle\phi_1|\phi_2(1,1)\phi_3(x,\bar{x})|\phi_4\rangle$. By inserting a complete set of states one can show that this correlation function can be written as a quadratic combination of so-called conformal blocks $\mathcal{F}_{12}^{34}(k|z)$

$$\langle\phi_1|\phi_2(1,1)\phi_3(x,\bar{x})|\phi_4\rangle = \sum_k (C_{21}^k)^* C_{34}^k \mathcal{F}_{12}^{34}(k|x)\bar{\mathcal{F}}_{12}^{34}(k|\bar{x}). \tag{2.15}$$

where the index k labels the primary operators of the theory. In Sect.3.3 we discuss how null vector conditions can be used to derive differential equations for the conformal blocks.

2.2 Ising model

The Ising model is a classical model of interacting spins, each spin having two states $s_i = \pm 1$. In the absence of an external field, the energy of a configuration s is

$$E(s) = -J \sum_{\langle ij \rangle} s_i s_j. \tag{2.16}$$

It is an interesting system to study because in 2d it is exactly solvable and exhibits a second-order phase transition. The value of the coupling at critical temperature is known to be

$$J_c = \frac{1}{2} \ln(1 + \sqrt{2}). \tag{2.17}$$

At the critical point, the correlation length ξ diverges and the only remaining characteristic length scale is the lattice spacing a. Therefore all the observables which are insensitive to the lattice spacing, i.e. the infrared observables, can be described by a scale invariant theory. In fact one assumes the stronger conformal invariance and adopts a CFT description. For the critical Ising model, the infrared physics can be captured by a CFT with three primary fields

- identity: $\phi_{11}(z, \bar{z}),\ h_{11} = 0$
- spin: $\phi_{12}(z, \bar{z}) \equiv \sigma(z, \bar{z}),\ h_\sigma = \frac{1}{16}$
- energy: $\phi_{21} \equiv \epsilon(z, \bar{z}),\ h_\epsilon = \frac{1}{2}$

and a central charge $c = \frac{1}{2}$. The central charge can be written in the Kac parameterization Eq.(2.13) with $b = \frac{\sqrt{3}}{2}$.

2.3 Entanglement Entropy

Entanglement is a property of quantum systems which are in a state that is not a product state. Entangled systems exhibit particular correlations which make entanglement a central resource in e.g. quantum information. A convenient measure of entanglement is the entanglement entropy [3]. Given a quantum system partitioned into two subsystems A and B, the entanglement entropy between A and B is defined as

$$S_A = -\mathrm{Tr}\, \rho_A \ln \rho_A \tag{2.18}$$

where $\rho_A = \mathrm{Tr}_B\, |\psi\rangle\langle\psi|$ is the reduced density matrix of A when the system is in the state $|\psi\rangle$. Thus entanglement entropy can be seen as the Von Neumann entropy of subsystem A in a heat bath consisting of B (or vice versa). When A and B are not entangled, the reduced density matrix ρ_A is a projector which means its eigenvalues are either 1 or 0 and therefore $S_A = 0$. Hence A and B entangled corresponds to $S_A \neq 0$ meaning A is observed to be in a mixed state.

3 Rényi Entanglement Entropies

The Von Neumann entanglement entropy $S_A = -\mathrm{Tr}\,\rho_A \ln \rho_A$ is difficult to calculate as it requires the knowledge of the full spectrum of the reduced density matrix. On the other hand the Rényi entropies

$$S_A^{(N)} = -\frac{1}{N-1} \ln \mathrm{Tr}\,\rho_A^N \tag{3.1}$$

turn out to be easier to compute and in general behave similarly as the Von Neumann entropy, thus providing an interesting alternative measure of entanglement. In addition, if one computes $S_A^{(N)}$ as an analytic function of N, then the Von Neumann entropy is recovered in the limit $N \to 1$

$$\lim_{N\to 1} S_A^{(N)} = -\lim_{N\to 1} \frac{\partial}{\partial N} \ln \mathrm{Tr}\,\rho_A^N = S_A \tag{3.2}$$

3.1 A Replica Trick

A recipe to calculate $\mathrm{Tr}\,\ln \rho_A^N$ for integer N was given in [4]. The idea is that one can think of each element of the reduced density matrix $(\rho_A)_{\alpha\beta} = \rho_{\alpha\gamma,\beta\gamma}$ as the partition function of a system where the boundaries in B are identified and the boundaries in A are fixed to $|\alpha\rangle$ and $|\beta\rangle \in H_A$. Then taking the trace of ρ_A^N corresponds to the partition function on a surface obtained by cyclically gluing together N copies of the original surface along A. If A consists of k disjoints intervals then let the resulting surface be denoted $\mathcal{R}_{N,k}$. This Riemann surface has N sheets connected by $2k$ branch points corresponding to the endpoints $[u_i, v_i]$ of each interval (see Fig.1). We will refer to this surface as the replicated surface. Let us denote the partition function on the replicated surface by Z_N then

$$\ln \mathrm{Tr}\,\rho_A^N = Z_N \tag{3.3}$$

provided the states are normalized such that the partition function on the initial surface is unity.

Instead of considering the theory on a complicated Riemann surface, there is another useful, equivalent, point of view. One can consider the theory on the original surface but with each degree of freedom replicated N times and an internal $\mathbb{Z}_N$ symmetry. Such a theory is referred to as a $\mathbb{Z}_N$-orbifold. Each copy can be thought of as living on one sheet of the replicated surface while the quotient by $\mathbb{Z}_N$ ensures cyclic invariance. To complete the equivalence, such a theory must be equipped with so-called twist fields $\tau_N, \tilde{\tau}_N$. These operators implement the branch cuts by allowing degrees of freedom from consecutive replicas to interact.

The correspondence between the two points of view is given by

$$\langle O_i(z,\bar{z})\rangle_{\mathcal{L},\mathcal{R}_{N,k}} = \frac{\langle \tau_N(u_1,\bar{u}_1)\tilde{\tau}_N(v_1,\bar{v}_1)...\tau_N(u_k,\bar{u}_k)O_i(z,\bar{z})\rangle_{\mathcal{L}^{(N)}}}{\langle \tau_N(u_1,\bar{u}_1)\tilde{\tau}_N(v_1,\bar{v}_1)...\tau_N(u_k,\bar{u}_k)\tilde{\tau}_N(v_k,\bar{v}_k)\rangle_{\mathcal{L}^{(N)}}} \tag{3.4}$$

where the subscript $\mathcal{L}, \mathcal{R}_{N,k}$ means that the expectation value is to be taken with the original degrees of freedom on the replicated surface and the subscript $\mathcal{L}^{(N)}$ means that the expectation value is to be taken with the replicated degrees of freedom on the original surface. We will omit these subscripts when there is no ambiguity. Here O_i is an operator constructed from degrees of freedom of the i^{th} sheet/replica. The partition function on $\mathcal{R}_{N,k}$ corresponds to a $2k$-point correlation function of twist operators

$$Z_N = \langle \tau_N(u_1, \bar{u}_1)\tilde{\tau}_N(v_1, \bar{v}_1)...\tau_N(u_k, \bar{u}_k)\tilde{\tau}_N(v_k, \bar{v}_k)\rangle_{\mathcal{L}^{(N)}} \tag{3.5}$$

Analytical computations performed within this framework are presented in the remainder of this section.

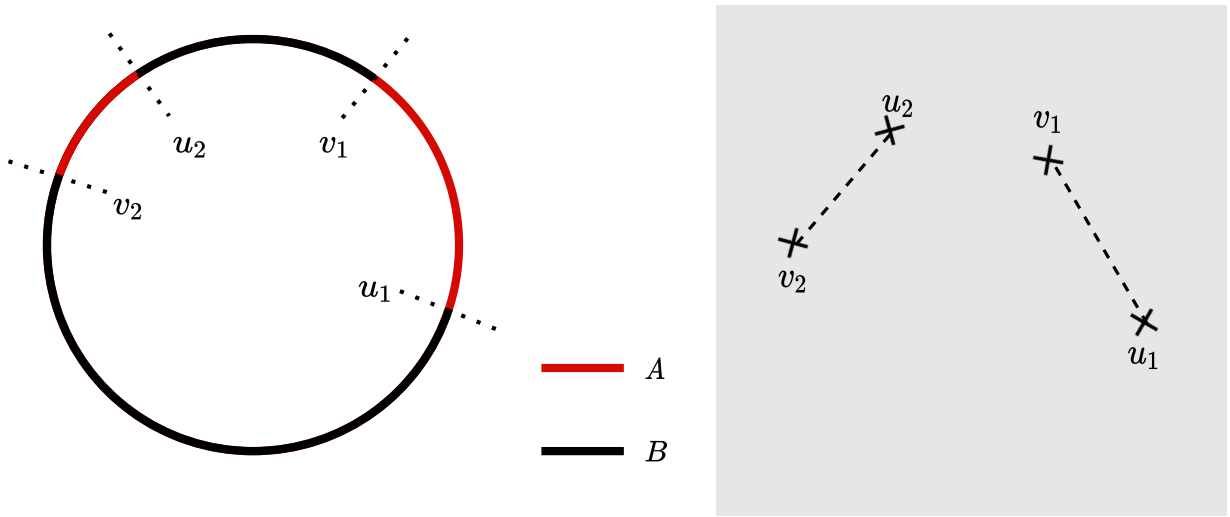

Figure 1: (Left) Example of a partition of the 1d quantum system where A consists of 2 intervals $[u_1, v_1]$ and $[u_2, v_2]$. (Right) Replicated surface $\mathcal{R}_{N,2}$ corresponding to such a partition. The points u_1, v_1, u_2, v_2 are branch points of order N and the dashed lines represent branch cuts.

3.1.1 Rényi entropies of a single interval in the vacuum

According to Eq.(3.5), the Rényi entropies of a single interval $x \in [0, il]$ in the vacuum can be calculated as a 2-point function of twist fields. On a cylinder of circumference L, using Eq.(2.4), we have

$$\langle \tau_N(0,0)\tilde{\tau}_N(il, -il)\rangle_{\mathbb{R}\times S^1} = \left[\frac{L}{\pi}\sin\frac{\pi l}{L}\right]^{-4h_N} \tag{3.6}$$

and thus it suffices to calculate the conformal dimension h_N of the twist fields $\tau_N, \tilde{\tau}_N$. This can be done by considering the vacuum expectation value of the energy momentum tensor

on the replicated surface

$$\langle T\rangle_{\mathcal{R}_{N,1}} = \frac{\langle \tau_N(u,\bar{u})\tilde{\tau}_N(v,\bar{v})t_i(w)\rangle_{\mathbb{C}}}{\langle \tau_N(u,\bar{u})\tilde{\tau}_N(v,\bar{v})\rangle_{\mathbb{C}}}, \quad i = 1...N \tag{3.7}$$

where $t_i(w)$ is the energy momentum tensor of the i^{th} replica. The complete energy momentum tensor of the replicated theory $T(w) = \sum_{i=1}^{N} t_i(w)$ can be used as a probe for the conformal dimension. Indeed if we let $w \to u, v$, the operator product expansion

$$T(w)\tau_N(u,\bar{u}) = \Big(\frac{h_N}{(w-u)^2} + \frac{1}{w-u}\frac{\partial}{\partial u}\Big)\tau_N(u,\bar{u}) \tag{3.8}$$

and similarly for v, gives

$$\langle \tau_N(u,\bar{u})\tilde{\tau}_N(v,\bar{v})T(w)\rangle = \Big(\frac{h_N}{(w-u)^2} + \frac{1}{w-u}\frac{\partial}{\partial u} + \frac{h_N}{(w-v)^2} + \frac{1}{w-v}\frac{\partial}{\partial v}\Big)\langle \tau_N(u,\bar{u})\tilde{\tau}_N(v,\bar{v})\rangle. \tag{3.9}$$

Using that on the plane $\langle \tau_N(u,\bar{u})\tilde{\tau}_N(v,\bar{v})\rangle_{\mathbb{C}} = (u-v)^{-2h_N}(\bar{u}-\bar{v})^{-2h_N}$, we have

$$\langle T\rangle_{\mathcal{R}_{N,1}} = \frac{h_N}{N}\Big(\frac{1}{(w-u)^2} - \frac{2}{(w-u)} + \frac{1}{(w-v)^2} + \frac{2}{(w-v)}\Big) \tag{3.10}$$

$$= \frac{h_N}{N}\frac{(u-v)^2}{(w-u)^2(w-v)^2} \tag{3.11}$$

On the other hand, $\langle T\rangle_{\mathcal{R}_{N,1}}$ can be calculated directly by performing a conformal transformation from $\mathbb{C}$ to $\mathcal{R}_{N,1}$. If z is the coordinate on the complex plane, the transformation

$$\xi = \Big(\frac{z-u}{z-v}\Big)^{\frac{1}{N}} \tag{3.12}$$

conformally maps $\mathbb{C}$ onto $\mathcal{R}_{N,1}$. Under such a transformation the energy momentum tensor transforms as

$$T(\xi) = \Big(\frac{dz}{d\xi}\Big)^2 T(w) + \frac{c}{12}\{z,\xi\} \tag{3.13}$$

Since $\langle T(z)\rangle = 0$ (otherwise translational and rotational invariance would be broken) we have

$$\langle T\rangle_{\mathcal{R}_{N,1}} = \frac{c}{12}\{z,\xi\} = \frac{c}{24}\Big(1 - \frac{1}{N^2}\Big)\frac{(u-v)^2}{(w-u)^2(w-v)^2} \tag{3.14}$$

Comparing Eq.(3.14) with Eq.(3.10) yields the conformal dimension of the twist operators for a theory replicated N times

$$h_N = \frac{c}{24}\Big(N - \frac{1}{N}\Big) \tag{3.15}$$

Hence the Rényi entropies on the cylinder for a single interval $A = [0, l]$ in the vacuum is

$$S_A^{(N)} = -\frac{1}{N-1} \ln Z_N = \frac{c}{6}\left(1 + \frac{1}{N}\right) \ln \left[\frac{L}{\pi} \sin\left(\frac{\pi l}{L}\right)\right] \tag{3.16}$$

In this simple case, one can even compute the Von Neumann entropy explicitly

$$S_A = \lim_{N\to 1} S_A^{(N)} = -\lim_{N\to 1} \frac{\partial}{\partial N} \ln Z_N = \frac{c}{3} \ln \left[\frac{L}{\pi} \sin\left(\frac{\pi l}{L}\right)\right]. \tag{3.17}$$

3.1.2 Rényi entropies of a single interval in an excited state

Here we compute the Rényi entanglement entropy when the system is in a primary state $|\phi\rangle$ of conformal dimension $h = \bar{h}$. Instead of taking vacuum expectation values of twist operators we must now take expectation values in the state $|\phi\rangle$ hence

$$Z_N = \lim_{R\to\infty} \frac{\langle \Phi(-R,-R)\tau_N(0,0)\tilde{\tau}_N(il,-il)\Phi(R,R)\rangle_{\mathbb{R}\times S^1}}{\langle \Phi(-R,-R)\Phi(R,R)\rangle_{\mathbb{R}\times S^1}} \tag{3.18}$$

where $\Phi = \otimes^N \phi$ is a primary field of conformal dimension Nh in the orbifold. Using the conformal transformation $\xi = \frac{L}{2\pi} \ln \omega$ we can express Z_N in terms of correlation functions on the complex plane. Letting $M \equiv \frac{2\pi}{L} R$ and $\alpha \equiv 2\pi \frac{l}{L}$, we have

$$Z_N = \lim_{M\to\infty} \left(\frac{L}{2\pi}\right)^{-4h_N} \frac{\langle \Phi(e^{-M}, e^{-M})\tau_N(1,1)\tilde{\tau}_N(e^{i\alpha}, e^{-i\alpha})\Phi(e^M, e^M)\rangle_{\mathbb{C}}}{\langle \Phi(e^{-M}, e^{-M})\Phi(e^M, e^M)\rangle_{\mathbb{C}}} \tag{3.19}$$

The denominator is simply a 2-point function

$$\langle \Phi(e^{-M}, e^{-M})\Phi(e^M, e^M)\rangle_{\mathbb{C}} = (e^{-M} - e^M)^{-4Nh} = (2\sinh M)^{-4Nh} \tag{3.20}$$

Using Eq.(3.4) we can express the numerator in terms of a correlation function in the original theory but now considered on the replicated surface $\mathcal{R}_{N,1}$.

$$\begin{aligned} &\langle \Phi(e^{-M}, e^{-M})\tau_N(1,1)\tilde{\tau}_N(e^{i\alpha}, e^{-i\alpha})\Phi(e^M, e^M)\rangle_{\mathbb{C}} \\ &= \langle \Phi(e^{-M}, e^{-M})\Phi(e^M, e^M)\rangle_{\mathcal{R}_{N,1}} \langle \tau_N(1,1)\tilde{\tau}_N(e^{i\alpha}, e^{-i\alpha})\rangle_{\mathbb{C}} \end{aligned} \tag{3.21}$$

The twist field 2-point function was calculated in the previous section. To calculate the remaining Φ correlation function we can conformally map $\mathcal{R}_{N,1}$ onto the complex plane with

$$z = \left(\frac{\xi - 1}{\xi - e^{i\alpha}}\right)^{\frac{1}{N}} \iff \xi = \frac{1 - e^{i\alpha} z^N}{1 - z^N} \tag{3.22}$$

This transformation also maps the branch points $\xi_U = 1$ and $\xi_V = e^{i\alpha}$ to $z_U = 0$ and $z_V = \infty$, respectively. Under this conformal map, the correlation function transforms covariantly

$$\begin{aligned}&\langle\Phi(e^{-M},e^{-M})\Phi(e^{M},e^{M})\rangle_{\mathcal{R}_{N,1}}\\&=\prod_{\substack{k=1\dots N\\a=+,-}}\left(\frac{d\xi}{dz}\right)^{-h}_{z_{ka}}\left(\frac{d\bar{\xi}}{d\bar{z}}\right)^{-h}_{\bar{z}_{ka}}\langle\phi(z_{1-},\bar{z}_{1-})...\phi(z_{N-},\bar{z}_{N-})\phi(z_{1+},\bar{z}_{1+})...\phi(z_{N+},\bar{z}_{N+})\rangle_{\mathbb{C}}\end{aligned}\tag{3.23}$$

where $\frac{d\xi}{dz} = (e^{i\alpha}-1)\frac{Nz^{N-1}}{(1-z^N)^2}$ and the points z_{k-}, z_{k+} are the $2N$ images of e^{-M}, e^{+M} under the conformal map

$$z_{k-} = \left(\frac{e^{-M}-1}{e^{-M}-e^{i\alpha}}\right)^{\frac{1}{N}}, \qquad z_{k+} = \left(\frac{e^{M}-1}{e^{M}-e^{i\alpha}}\right)^{\frac{1}{N}}\tag{3.24}$$

In the end we will be interested in the limit $M\to\infty$ of $\left(\frac{d\xi}{dz}\frac{d\bar{\xi}}{d\bar{z}}\right)_{z_{k\pm}}$. This limit is harmless for the points z_{k-} so we can take it at once and obtain

$$z_{k-}\to e^{-i\frac{\alpha}{N}}r_k\,, \qquad \left(\frac{d\xi}{dz}\frac{d\bar{\xi}}{d\bar{z}}\right)_{z_{k-}}\to\frac{N^2}{4\sin^2\frac{\alpha}{2}}\tag{3.25}$$

where the $r_k = e^{i2\pi\frac{k-1}{N}}$ are the N^{th} roots of unity. The points z_{k+} require some additional care. One can find the following asymptotic expression

$$z_{k+}\to r_k\,, \qquad \left(\frac{d\xi}{dz}\frac{d\bar{\xi}}{d\bar{z}}\right)_{z_{k+}}\simeq\frac{N^2}{4\sin^2\frac{\alpha}{2}}e^{-4M}\tag{3.26}$$

Therefore we have that

$$\begin{aligned}&\langle\Phi(e^{-M},e^{-M})\Phi(e^{M},e^{M})\rangle_{\mathcal{R}_{N,1}}\\&\simeq\left[\frac{4}{N}\sin^2\frac{\alpha}{2}\right]^{2Nh}e^{4NhM}\langle\phi(e^{-i\frac{\alpha}{N}}r_1,e^{i\frac{\alpha}{N}}\bar{r}_1)...\phi(e^{-i\frac{\alpha}{N}}r_n,e^{i\frac{\alpha}{N}}\bar{r}_N)\phi(r_1,\bar{r}_1)...\phi(r_N,\bar{r}_N)\rangle_{\mathbb{C}}\end{aligned}\tag{3.27}$$

Putting everything together we find

$$\begin{aligned}Z_N=\lim_{M\to\infty}(2e^{-M}\sinh M)^{4Nh}&\left[\frac{L}{\pi}\sin\pi\frac{l}{L}\right]^{-4h_N}\times\left[\frac{4}{N}\sin^2\pi\frac{l}{L}\right]\\&\times\langle\phi(e^{-i\frac{\alpha}{N}}r_1,e^{i\frac{\alpha}{N}}\bar{r}_1)...\phi(e^{-i\frac{\alpha}{N}}r_N,e^{i\frac{\alpha}{N}}\bar{r}_N)\phi(r_1,\bar{r}_1)...\phi(r_N,\bar{r}_N)\rangle_{\mathbb{C}}\end{aligned}\tag{3.28}$$

The limit $M\to\infty$ can now be taken safely and we obtain

$$\begin{aligned}Z_N=&\left[\frac{L}{\pi}\sin\pi\frac{l}{L}\right]^{-4h_N}\times\left[\frac{4}{N}\sin^2\pi\frac{l}{L}\right]^{2Nh}\\&\times\langle\phi(e^{-i\frac{\alpha}{N}}r_1,e^{i\frac{\alpha}{N}}\bar{r}_1)...\phi(e^{-i\frac{\alpha}{N}}r_N,e^{i\frac{\alpha}{N}}\bar{r}_N)\phi(r_1,\bar{r}_1)...\phi(r_N,\bar{r}_N)\rangle_{\mathbb{C}}\end{aligned}\tag{3.29}$$

In the case of the Ising model the $2N$-point functions of the energy are known to be [1]

$$\langle \epsilon(z_1,\bar{z}_1)...\epsilon(z_{2N},\bar{z}_{2N})\rangle_{\mathbb{C}} = \left|\det\left(\frac{1}{z_i-z_j}\right)\right| \tag{3.30}$$

Our final expression for the Ising model on the cylinder is therefore

$$S_A^{(N)} = \frac{1}{N-1}\left(\frac{1}{12}\left(N-\frac{1}{N}\right)\ln\left[\frac{L}{\pi}\sin\pi\frac{l}{L}\right] - N\ln\left[\frac{4}{N}\sin^2\pi\frac{l}{L}\right] - \ln\left|\det\left(\frac{1}{z_i-z_j}\right)\right|\right). \tag{3.31}$$

3.1.3 2nd Rényi entropy of 2 intervals in the vacuum

One could partition the 1d quantum system into the union of 2 disjoint intervals and its complement. In that case, the replica approach becomes significantly more difficult since the genus of the covering surface increases with N. For $N = 2$, the covering surface is a torus which already introduces technical difficulties. Another complication is that computing Rényi entropies now requires computing 4-point functions of twist operators. In this section we compute the 2nd Rényi entropy of a generic CFT by considering a toroidal geometry. The derivation follows what was done in [5] for the $\mathbb{Z}_2$-orbifold of a bosonic string.

We would like to compute

$$Z_2 = \langle \tau_2(u_1,\bar{u}_1)\tilde{\tau}_2(v_1,\bar{v}_1)\tau_2(u_2,\bar{u}_2)\tilde{\tau}_2(v_2,\bar{v}_2)\rangle_{\mathbb{C}}. \tag{3.32}$$

whose logarithm will give $S_A^{(2)}$. By applying the global conformal transformation $w = \frac{v_1-u_1}{v_1-v_2}\frac{z-v_2}{z-u1}$ one can map the points u_1, v_1, u_2, v_2 to $\infty, 1, x = \frac{v_1-u_1}{v_1-v_2}\frac{u_2-v_2}{u_2-u_1}, 0$. So without loss of generality we may only consider the following standard 4-point function

$$\langle \tau_2(\infty,\infty)\tilde{\tau}_2(1,1)\tau_2(x,\bar{x})\tilde{\tau}_2(0,0)\rangle_{\mathbb{C}} \equiv F(x). \tag{3.33}$$

The derivation proceeds in a similar manner as for the Rényi entropy of a single interval. By considering the vacuum expectation value of the energy momentum tensor on the replicated surface, one can extract the logarithmic derivative of $F(x)$. Indeed, using the OPE Eq.(2.7), we have

$$\begin{aligned}\langle t(z)\rangle_{\mathcal{R}_{2,2}} = \frac{\langle \tau_2(\infty,\infty)\tilde{\tau}_2(1,1)\tau_2(x,\bar{x})t(z)\tilde{\tau}_2(0,0)\rangle_{\mathbb{C}}}{\langle \tau_2(\infty,\infty)\tilde{\tau}_2(1,1)\tau_2(x,\bar{x})\tilde{\tau}_2(0,0)\rangle_{\mathbb{C}}} &= \frac{1}{F(x)}\left(\frac{h_2}{(z-x)^2}+\frac{1}{(z-x)}\frac{\partial}{\partial x}+\text{reg.}\right)F(x)\\ &= \frac{h_2}{(z-x)^2}+\frac{1}{z-x}\frac{\partial \ln F(x)}{\partial x}+\text{reg.}.\end{aligned} \tag{3.34}$$

As before, the vacuum expectation value of the energy momentum tensor on the replicated plane and on the torus can be related by performing a conformal transformation. The

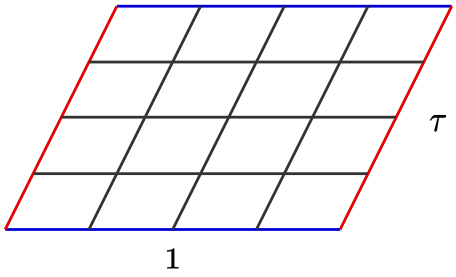

Figure 2: A torus with modular parameter τ. Opposites sides are to be identified.

Weierstrass $\wp$ function provides a conformal map between the replicated plane $\mathcal{R}_{2,2}$ and the torus T^2

$$z(\omega) = \frac{\wp(\omega) - e_1}{e_2 - e_1}\,, \quad x = \frac{e_3 - e_1}{e_2 - e_1} \tag{3.35}$$

where z and ω are the coordinates on the replicated plane and on the torus. Topologically, a torus is a parallelogram with opposite edges identified. One may take the length of the sides of the parallelogram to be 1 and τ (see Fig.2). In Eq.(3.35), the numbers e_1, e_2 and e_3 are given by $\wp(\frac{1}{2})$, $\wp(\frac{1}{2}\tau)$ and $\wp(\frac{1}{2}(1+\tau))$, respectively, and they satisfy $e_1+e_2+e_3=0$. In our case the modular parameter τ is related to the branch point x by the following relation

$$x = \left(\frac{\theta_3(\tau)}{\theta_4(\tau)}\right)^4 = \prod_{n=1}^{\infty}\left(\frac{1+q^{n-\frac{1}{2}}}{1-q^{n-\frac{1}{2}}}\right)^8\,, \quad q = e^{2i\pi\tau}. \tag{3.36}$$

We can use the conformal transformation Eq.(3.35) to relate the energy momentum tensor on the replicated plane and on the torus

$$\langle T\rangle_{T^2} = \left(\frac{dz}{d\omega}\right)^2\langle t(z)\rangle + \frac{c}{12}\{z,\omega\}. \tag{3.37}$$

This can be put into a more useful form by using some properties of the $\wp$ function. First, the Weierstrass $\wp$ function satisfies a differential equation from which we can express the derivative of z [6]

$$\frac{d\wp}{d\omega} = \sqrt{4(\wp-e_1)(\wp-e_2)(\wp-e_3)} \implies \frac{dz}{d\omega} = 2(e_2-e_1)^{\frac{1}{2}}\sqrt{(z-1)(z-x)z}. \tag{3.38}$$

The above relation between $\wp$ and $\wp'$ can be used to simplify the Schwarzian derivative $\{z,\omega\} = \{\wp,\omega\}$. By differentiating Eq.(3.38) one finds

$$\begin{aligned}\wp'' &= 6\wp^2 + 2(e_1e_2 + e_2e_3 + e_3e_1)\\ \wp''' &= 12\wp\wp'\end{aligned} \tag{3.39}$$

Using these relations, the term involving the Schwarzian derivative can be written

$$\frac{c}{12}\{z,\omega\} = c\wp - \frac{c}{8}\left(\frac{d}{d\omega}\ln\wp'\right)^2 = c(e_2-e_1)z + ce_1 - \frac{c}{8}\left(\frac{d}{d\omega}\ln z'\right)^2 \tag{3.40}$$

On the torus, we also have the following expression for the vacuum expectation value of the energy momentum tensor

$$\langle T\rangle_{T^2} = 2i\pi\frac{\partial}{\partial\tau}Z_\tau \tag{3.41}$$

where Z_τ is the partition function on the torus. Putting everything together we can express the vacuum expectation value of the energy momentum tensor on the plane as

$$\begin{aligned}\langle T(z)\rangle &= 2\Big(\frac{d\omega}{dz}\Big)^2\Big[\langle T\rangle_{T^2} - \frac{c}{12}\{z,\omega\}\Big]\\ &= 2\Big(\frac{d\omega}{dz}\Big)^2\Big[2i\pi\frac{\partial}{\partial\tau}Z_\tau - c(e_2-e_1)z - ce_1 + \frac{c}{4}\Big(\frac{d}{d\omega}\ln z'\Big)^2\Big]\\ &= \frac{2i\pi\frac{\partial}{\partial\tau}Z_\tau - c(e_2-e_1)z - ce_1}{2(e_2-e1)(z-1)(z-x)z} + \frac{c}{16}\Big(\frac{1}{z-1}+\frac{1}{z-x}+\frac{1}{z}\Big)^2\end{aligned} \tag{3.42}$$

Near $z = x$ we have

$$\langle T(z)\rangle = \frac{c}{16}\frac{1}{(z-x)^2} + \frac{1}{z-x}\Big[\frac{2i\pi\frac{\partial}{\partial\tau}Z_\tau - c(e_2-e_1)z - ce_1}{2(e_2-e1)(x-1)x} + \frac{c}{8}\Big(\frac{1}{x-1}+\frac{1}{x}\Big)\Big] + \text{reg..} \tag{3.43}$$

Comparing with Eq.(3.34), one finds for the logarithmic derivative of the twist 4-point function

$$\frac{d}{dx}\ln F(x) = \frac{2i\pi\frac{\partial}{\partial\tau}Z_\tau - c(e_2-e_1)z - ce_1}{2(e_2-e1)(x-1)x} + \frac{c}{8}\Big(\frac{1}{x-1}+\frac{1}{x}\Big) \tag{3.44}$$

and of course, the conformal dimension of the twist field $h_2 = \frac{c}{16}$ is recovered. As it is there is some implicit x dependence in Eq.(3.43) that comes from e_1, e_2 and Z_τ. This x dependence can be made explicit by using the identities

$$\begin{aligned}\frac{1}{4(e_2-e_1)(x-1)x} &= \frac{1}{4i\pi}\frac{d\tau}{dx}\\ \frac{e_2}{e_1} &= \frac{x-2}{x+1}\end{aligned} \tag{3.45}$$

Finally one arrives at a simple differential equation for the twist 4-point function

$$\frac{d}{dx}\ln F(x) = -\frac{c}{24}\Big(\frac{1}{x-1}+\frac{1}{x}\Big) + \frac{d}{dx}\ln Z_\tau. \tag{3.46}$$

After an elementary integration this yields the twist 4-point function

$$\langle\tau_2(\infty,\infty)\tilde\tau_2(1,1)\tau_2(x,\bar x)\tilde\tau_2(0,0)\rangle = cst\times Z_\tau\times|x|^{-\frac{c}{12}}|x-1|^{-\frac{c}{12}}. \tag{3.47}$$

Now obtaining the Rényi entropy $S_A^{(2)}$ is only a matter of algebra. All that is needed is to re-express this result in terms of the end points of the two intervals u_1, v_1, u_2, v_2 with a conformal transformation and to take the logarithm.

3.2 Geometrical picture : conical singularities

One could compute twist correlation functions by directly computing the partition function on the relevant Riemann surface. The Riemann surface one has to consider has conical singularities corresponding to twist field insertions in addition to any singularities of the original surface (e.g. the Riemann sphere has a conical singularity at ∞). An algorithm to regularize these singularities and calculate the partition function was presented in [7]

- Cut out holes around the conical singularities
- Conformally map the cut surface onto the covering surface (a surface obtained by "unfolding" the sheets)
- Fill the images of the holes with flat patches
- Relate the resulting metric to the flat metric by a Weyl transformation: $ds^2 = e^{\phi} d\hat{s}^2$
- Compute the Liouville action $S_L[\phi] = \frac{c}{96\pi} \int_M d^2z \, \hat{g}^{\mu\nu} \partial_\mu \phi \partial_\nu \phi$.

The desired partition function is then given by

$$Z_g = e^{S_L[\phi]} Z_{\hat{g}} \tag{3.48}$$

where g is the metric on the image of the cut surface and $\hat{g}$ is the flat metric on the covering surface. This approach thus relies on the known transformation of the partition function under a Weyl transformation.

Let us illustrate this procedure for the calculation of the twist 4-point function

$$\langle \tau_2(z_\infty, \bar{z}_\infty) \tilde{\tau}_2(1,1) \tau_2(x, \bar{x}) \tilde{\tau}_2(0,0) \rangle \tag{3.49}$$

where the limit $z \to z_\infty$ will be taken at the end of the calculation. In that case it boils down to computing the partition function on a 2-sheeted Riemann surface which has conical singularities at $z_\infty, 1, x$ and 0. We cut out holes of radius ϵ around these points. There is also the conical singularity of the Riemann sphere at ∞. We cut a hole of radius δ around this point. Now if z is the coordinate on this surface, the map $z(\omega)$ that performs the uniformization is given by

$$\frac{dz}{d\omega} = \alpha \Big[(z - z_\infty)(z-1)(z-x)z \Big]^{\frac{1}{2}} \tag{3.50}$$

The covering surface parametrized by ω is a torus and, in the limit $z_\infty \to \infty$, $z(\omega)$ is given as in Eq.(3.35) by the Weierstrass $\wp$ function. The torus thus obtained inherits the metric $ds^2 = dz d\bar{z} = \left| \frac{dz}{d\omega} \right|^2 d\omega d\bar{\omega}$ outside the holes and a flat metric inside the holes. This metric is related to the flat metric on the torus by a Weyl transformation e^{ϕ} where

$$\phi = \ln \frac{dz}{d\omega} + \ln \frac{d\bar{z}}{d\bar{\omega}} \tag{3.51}$$

outside the holes. We can now compute the terms of the Liouville action $S_L[\phi]$ that contribute to the 4-point function. Since ϕ is the sum of a holomorphic and an anti-holomorphic function $\hat{g}^{\mu\nu}\partial_\mu\partial_\nu\phi = 4\partial_{\bar{\omega}}\partial_\omega\phi = 0$. Hence, using Stokes theorem, the only contribution to the Liouville action comes from the boundaries of the surface i.e. the boundaries of the holes

$$S_L[\phi] = \frac{c}{96\pi}\int_{\partial M} dl_\mu \hat{g}^{\mu\nu}\phi\partial_\nu\phi \tag{3.52}$$

Expressing everything in complex coordinates for a hole of radius r around 0 we find

$$\begin{aligned} g^{\mu\nu}dl_\mu\partial_\nu &= rd\theta\Big(-\frac{z}{|z|}\partial_z - \frac{\bar{z}}{\bar{z}}\partial_{\bar{z}}\Big) \\ d\theta &= -i\frac{dz}{z} = i\frac{d\bar{z}}{\bar{z}} \\ \rightarrow S_L[\phi] &= \frac{ic}{96\pi}\Big[\int_{\partial M} dz\phi\partial_z\phi - c.c\Big] \end{aligned} \tag{3.53}$$

To continue we only need the dominant part of ϕ near each of the singularities. We show the calculation for the hole around $z = x$.

$$\begin{aligned} \frac{dz}{d\omega} &\approx \alpha\Big[(z-x)(x-z_\infty)(x-1)x\Big]^{\frac{1}{2}} \\ \phi &\approx \ln\Big(|\alpha|^2|z-x||x-z_\infty||x-1||x|\Big) \\ \partial_z\phi &\approx \frac{1}{2(z-x)} \end{aligned} \tag{3.54}$$

Writing $z = x + \epsilon e^{i\theta}$ we have the following contribution to the Liouville action

$$\begin{aligned} S_L(z=x) &= 2\times\frac{ic}{96\pi}\int_0^{4\pi} i\epsilon e^{i\theta}d\theta\frac{1}{2\epsilon e^{i\theta}}\ln\Big(|\alpha|^2\epsilon|x-z_\infty||x-1||x|\Big) \\ &= -\frac{c}{24}\ln\Big(|\alpha|^2\epsilon|x-z_\infty||x-1||x|\Big) \end{aligned} \tag{3.55}$$

where the integration is between 0 and 4π because it takes two turns of the z variable to go around the hole on the covering surface. The same calculation gives the contribution from the other singularities at $0, 1$ and z_∞

$$\begin{aligned} S_L(z=0) &= -\frac{c}{24}\ln\Big(|\alpha|^2\epsilon|z_\infty||x|\Big) \\ S_L(z=1) &= -\frac{c}{24}\ln\Big(|\alpha|^2\epsilon|1-z_\infty||1-x|\Big) \\ S_L(z=z_\infty) &= -\frac{c}{24}\ln\Big(|\alpha|^2\epsilon|z_\infty-1||z_\infty-x||z_\infty|\Big). \end{aligned} \tag{3.56}$$

There is also a contribution from the singularity at $z = \infty$. For $z \to \infty$ we have

$$\begin{aligned} \frac{dz}{d\omega} &\approx \alpha z^2 \\ \phi &\approx \ln(|\alpha|^2|z|^4) \\ \partial_z\phi &\approx \frac{2}{z} \end{aligned} \tag{3.57}$$

and therefore, parameterizing the contour as $\frac{1}{z} = \delta e^{i\theta}$, we find

$$\begin{aligned} S(z=\infty) &= 2 \times \frac{ic}{96\pi} \int_0^{4\pi} \frac{-i}{\delta} e^{-i\theta} \frac{2\delta}{e^{-i\theta}} \ln(|\alpha|^2 \delta^{-4}) \\ &= \frac{c}{6} \ln(|\alpha|^2 \delta^{-4}) \end{aligned} \tag{3.58}$$

The complete Liouville action is thus

$$\begin{aligned} S_L[\phi] &= S(z=z_\infty) + S(z=1) + S(z=x) + S(z=0) + S(z=\infty) \\ &= -\frac{c}{12} \ln|x||1-x| - \frac{c}{6} \ln\epsilon - \frac{2}{3} \ln\delta - \frac{c}{4} \ln|z_\infty| \end{aligned} \tag{3.59}$$

There are several regulation dependent terms in this equation. The terms involving ϵ and z_∞ disappear from the 4-point function after renormalization of the twist fields and of the vacuum state. The divergence in δ, which comes from the conical singularity of the Riemann sphere, is precisely the same as that of the partition function on the sphere. It therefore cancels when taking the ratio with the partition function on the sphere Z_δ. The final result is

$$\langle \tau_2(\infty,\infty) \tilde{\tau}_2(1,1) \tau_2(x,\bar{x}) \tilde{\tau}_2(0,0) \rangle = e^{S_L[\phi]} \frac{Z_\tau}{Z_\delta^2} = cst \times |x|^{-\frac{c}{12}} |1-x|^{-\frac{c}{12}} \times Z_\tau. \tag{3.60}$$

So using this completely geometric approach we were able to recover the previously derived twist 4-point function.

3.3 Algebraic picture : null vectors

There is another approach that can be used to compute twist 4-point functions. The idea is to fully exploit the algebraic structure of the orbifold CFT [8]. In the same way the Virasoro algebra is used to construct the Hilbert space of a CFT, the larger orbifold Virasoro algebra can be used to construct the Hilbert space of the orbifold CFT. In particular null vectors equations can also be used to derive ordinary differential equations for 4-point functions. In the Ising model, the identity operator ϕ_{11} has a null vector at level 1 and level 6 as can be seen from the Kac parametrization Eq.(2.13). The null vector conditions are

$$\begin{aligned} L_{-1}\phi_{11} &= 0 \\ (108L_{-6} + 264L_{-4}L_{-2} - L_{-3}^2 - 64L_{-2}^3)\phi_{11} &= 0 \end{aligned} \tag{3.61}$$

To these null vectors correspond null vectors for the twist field in the orbifold. Using the differential action of orbifold Virasoro operators inside the twist 4-point function, the following ordinary differential equation was derived in [8]

$$\begin{aligned} \Big((x-1)^3\partial_x^3 + \frac{33(2x-1)}{11x}(x-1)^2\partial_x^2 + \frac{3(192x(x-1)+5)}{256x^2}(x-1)\partial_x \\ + \frac{15(2x-1)}{4096x^3} \Big) G(x) = 0 \end{aligned} \tag{3.62}$$

The three linearly independent solutions of this equation are the three conformal blocks $\mathcal{F}^{\tau\tau}_{\tau\tau}(1|x)$, $\mathcal{F}^{\tau\tau}_{\tau\tau}(\epsilon|x)$ and $\mathcal{F}^{\tau\tau}_{\tau\tau}(\sigma|x)$. In the previous sections, we have found $\langle\tau\tau\tau\tau\rangle \propto |x|^{-\frac{1}{24}}|1-x|^{-\frac{1}{24}}Z_\tau$ so by making the change of function $G(x) = x^{-\frac{1}{48}}(1-x)^{-\frac{1}{48}}\chi(x)$ we can access the building blocks of the partition function on the torus, the Virasoro characters $\chi_{r,s}$

$$\begin{aligned} Z_\tau &= \chi_{1,1}\bar{\chi}_{1,1} + \chi_{1,2}\bar{\chi}_{1,2} + \chi_{2,1}\bar{\chi}_{2,1} \\ \chi_{r,s}(q) &= \mathrm{Tr}_{\mathcal{V}_{r,s}} q^{L_0-\frac{1}{48}} = q^{-\frac{1}{48}+h_{r,s}} \sum_{n\geq 0} \mathcal{N}_n^{(r,s)} q^n \end{aligned} \tag{3.63}$$

where $q(x) = e^{2i\pi\tau(x)}$. As can be seen from Eq.(3.63), $\mathcal{N}_n^{(r,s)}$ is the number of (physical) states at level n in $\mathcal{V}_{r,s}$. The equation satisfied by the characters was also given in [8]

$$\begin{aligned} \Big((x-1)^3\partial_x^3 + \frac{2(2x-1)}{x}(x-1)^2\partial_x^2 + \frac{391x(x-1)+7}{192x^2}(x-1)\partial_x \\ + \frac{23(2-x)(x+1)(2x-1)}{24^3x^3}\Big)\chi(x) = 0 \end{aligned} \tag{3.64}$$

We solve this equation around the regular singularity $x = 1$ by looking for a power series solution as in [2]. To do so it is more convenient to express Eq.(3.64) in terms of the differential operator $\theta \equiv (x-1)\partial_x$ whose eigenfunctions are $(x-1)^\alpha$.

$$\begin{aligned} &\Big(\theta^3 + q_1(x)\theta^2 + q_2(x)\theta + q_3(x)\Big)C(x) = 0 \\ &q_1(x) = \frac{2(2x-1)}{x} - 3 \\ &q_2(x) = \frac{391x(x-1)+7}{192x^2} - \frac{2(2x-1)}{x} + 2 \\ &q_3(x) = \frac{23(2-x)(x+1(2x-1))}{24^3x^3} \end{aligned} \tag{3.65}$$

Writing the solution $\chi(x) = (x-1)^\lambda \sum_{n\geq 0} a_n(x-1)^n$, the allowed exponents λ are the solutions of the characteristic equation

$$\begin{aligned} Q(\lambda) \equiv \lambda^3 + q_1(1)\lambda^2 + q_2(1)\lambda + q_3(1) = \lambda^3 - \lambda^2 + \frac{7}{192}\lambda + \frac{23}{6912} = 0 \\ \lambda_1 = \frac{-1}{24}\,, \quad \lambda_2 = \frac{1}{12}\,, \quad \lambda_3 = \frac{23}{24}. \end{aligned} \tag{3.66}$$

For each exponent, the coefficients of the power series are solutions of the following triangular linear system of equations

$$\sum_{n=0}^{k} Q_{k-n}(\lambda+n)a_n = 0\,, \quad k \in \mathbb{N} \tag{3.67}$$

where $Q_0(\lambda) \equiv R(\lambda)$ and for $n \geq 1$, $Q_n(\lambda) \equiv \frac{1}{n!}\frac{d^n}{dx^n}\Big(q_1(x)\lambda^2 + q_2(x)\lambda + q_3(x)\Big)\Big|_{x=1}$. Since the first equation is automatically satisfied $R(\lambda)a_0 = 0$, we can take $a_0 = 1$. For each exponent, we solved this system of equations in `Mathematica` for the first eight coefficients and the results are shown below

$\boldsymbol{\lambda = \frac{23}{24}}$

$$\chi_{2,1}(x) = (x-1)^{\frac{23}{24}}\Big[1 - \frac{23}{48}(x-1) + \frac{1357}{768}(x-1)^2 - \frac{12029}{6912}(x-1)^3 + \frac{989}{576}(x-1)^4 - \frac{1955}{1152}(x-1)^5 + \frac{11615}{1912}(x-1)^6 - \frac{3841}{2304}(x-1)^7 + \frac{1909}{1152}(x-1)^8 + ...\Big] \tag{3.68}$$

$\boldsymbol{\lambda = \frac{1}{12}}$

$$\chi_{1,2}(x) = (x-1)^{\frac{1}{12}}\Big[1 - \frac{1}{24}(x-1) + \frac{25}{1152}(x-1)^2 - \frac{1225}{82944}(x-1)^3 + \frac{89425}{7962624}(x-1)^4 - \frac{1734845}{191102976}(x-1)^5 + \frac{209916245}{27518828544}(x-1)^6 - \frac{4348265075}{660451885056}(x-1)^7 + \frac{734856797675}{126806761930752}(x-1)^8 + ...\Big] \tag{3.69}$$

$\boldsymbol{\lambda = -\frac{1}{24}}$

$$\chi_{1,1}(x) = (x-1)^{-\frac{1}{24}}\Big[1 + \frac{1}{48}(x-1) - \frac{19}{2304}(x-1)^2 + \frac{799}{165888}(x-1)^3 - \frac{52255}{15925248}(x-1)^4 + \frac{928739}{382205952}(x-1)^5 - \frac{104386667}{55037657088}(x-1)^6 + \frac{2027593997}{1320903770112}(x-1)^7 - \frac{323535808277}{253613523861504}(x-1)^8 + ...\Big] \tag{3.70}$$

We can find the first few coefficients of the characters $N_n^{(r,s)}$ by expressing these solutions in terms of q using Eq.(3.36). Expanding $x(q)$ around $q = 0$ gives

$$x(q) - 1 = 16q^{\frac{1}{2}}\left(1 + 8q^{\frac{1}{2}} + 44q + 192q^{\frac{3}{2}} + 718q^2 + 2400q^{\frac{5}{2}} + 7352q^3 + 20992q^{\frac{7}{2}} + 56549q^4 + ...\right) \tag{3.71}$$

Performing the substitution at order $\mathcal{O}(q^4)$ we recover the Virasoro characters of the Ising model as given in e.g. [1].

$$\begin{aligned}\chi_{1,1}(q) &= 1 + q^2 + q^3 + 2q^4 + ... \\ \chi_{1,2}(q) &= 1 + q + q^2 + 2q^3 + 2q^4 + ... \\ \chi_{2,1}(q) &= 1 + q + q^2 + q^3 + 2q^4 + ...\end{aligned} \tag{3.72}$$

4 Numerical simulations

We now present a numerical simulation of the critical Ising model using the transfer matrix method. We consider the Ising model on a cylindrical lattice of a circumference La and length Ma, with a the lattice spacing. Now to calculate expectation values in this 2d statistical system we need to be able to perform Boltzmann-weighted sums over all configurations. It is very useful to think of the longitudinal direction as imaginary time and work with a 1d quantum system consisting of a loop of L spins. The space of states of this 1d system is 2^L dimensional and we write the states as

$$|s\rangle\,, \quad s \in \{0...2^L - 1\} \tag{4.1}$$

A one-to-one correspondence between the integer label s and a configuration of the spins is naturally given by the binary expression of s.

The transfer matrix T is the (imaginary) time evolution operator of the 1d system

$$T_{ss'} = \left(e^{-\beta H}\right)_{ss'} = \prod_{k=0}^{L} e^{\frac{J}{2}s_j s_{j+1}} e^{\frac{J}{2}s'_j s'_{j+1}} e^{J s_j s'_j}. \tag{4.2}$$

Where the inverse temperature β has been absorbed in the coupling constant J. T acts on all the quantum states of the L spins so it is a $2^L \times 2^L$ matrix. The quantum Hamiltonian H is that of the transverse Ising chain. With this choice each matrix multiplication involving T performs the desired Boltzmann-weighted sum over the states of one slice. The partition function can then be computed as

$$Z = \langle b_l | T^M | b_r \rangle \tag{4.3}$$

for left and right boundary conditions $|b_l\rangle$ and $|b_r\rangle$ or $Z = \operatorname{Tr} T^M$ for periodic boundary conditions. Expectation values of operators in the 2d classical system are calculated in the 1d quantum system using

$$\langle \mathcal{O}(m,n) \rangle = \frac{\langle b_l | T^{\frac{M}{2}} \mathcal{O}(m,n) T^{\frac{M}{2}} | b_r \rangle}{\langle b_l | T^M | b_r \rangle} \tag{4.4}$$

where $\mathcal{O}(m,n) = T^{-m}\mathcal{O}(0,n)T^m$ is an operator on the m^{th} slice acting on the spin n.

As L grows, it quickly becomes computationally expensive to construct and handle the transfer matrix. The situation can be improved by noting that the knowledge of the full transfer matrix is not needed but only its action on an arbitrary vector. This greatly improves the memory usage by letting the program only use objects of size 2^L instead of 2^{2L}. The action of the transfer matrix can be broken down into the action of simple elementary matrices as

$$T|s\rangle = \Big(\prod_{j=0}^{L} p_j\Big)\Big(\prod_{j=0}^{L} q_j\Big)\Big(\prod_{j=0}^{L} p_j\Big)|s\rangle \tag{4.5}$$

with

$$p_j|s\rangle = e^{\frac{J}{2}s_j s_{j+1}}|s\rangle$$
$$q_j|s\rangle = e^{J}|s\rangle + e^{-J}|\tilde{s}^{(j)}\rangle \tag{4.6}$$

and $|\tilde{s}^{(j)}\rangle$ denotes the state s with the j^{th} spin flipped. The transfer matrix method implemented by elementary matrices is the core of the program written for this project (the code is available in Appendix A).

4.1 Vacuum energy and the central charge

From the eigenvalues of the transfer matrix one gets access to the energy spectrum of the quantum model. The two largest eigenvalues, corresponding to the first two energy eigenvalues were extracted using algorithms from [9]. The corresponding eigenstates were also computed. The algorithm is the following

- Generate a normalized random state $|x_0\rangle$
- $\frac{T|x_n\rangle}{||T|x_n\rangle||} = |x_{n+1}\rangle \quad , \quad \lambda_0^{(n)} = ||T|x_n\rangle||$
- Repeat until $||\,|x_{n+1}\rangle - |x_n\rangle\,||$ reaches the desired precision.

$\lambda_0^{(n)}$ and $|x_n\rangle$ converge to the largest eigenvalue and corresponding eigenvector λ_0, $|0\rangle$. To find the second eigenvector the algorithm is the same except that instead of iterating the transfer matrix T we project onto the subspace orthogonal to $|0\rangle$ at each step by iterating TP where $P = 1 - |0\rangle\langle 0|$. Using projectors, it is also possible to compute eigenvectors in the even or odd $\mathbb{Z}_2$ sector. If we let R be the matrix that flips all the spins then $\frac{1\pm R}{2}$ is the projector onto the even/odd sector.

From the finite-size scaling of the vacuum energy one can extract the central charge of the underlying CFT. This is a good consistency check, making sure that the simulated system is indeed described by the expected CFT. The following relation is expected from CFT [1]

$$-\ln \lambda_0(L) = f_0 L - \frac{\pi c}{6L} \tag{4.7}$$

We computed λ_0 for systems of size between $L = 10$ and $L = 16$ with a convergence criterion $||\,|x_{n+1}\rangle - |x_n\rangle\,|| < 10^{-8}$. The values are presented in Table 1. Fitting these values with Eq.(4.7), we find $c \approx 0.503$. This is perfectly consistent with the CFT description of the critical Ising model which has central charge $c = \frac{1}{2}$.

4.2 2-point functions and conformal dimensions

One can also estimate the conformal dimensions of operators by computing their 2-point functions. Indeed, on the cylinder, the CFT prediction for the 2-point function of an operator ϕ with conformal dimension $h = \bar{h}$ is

L	10	11	12	13	14	15	16
$\hat{E}_0 = -\ln \lambda_0$	-0.509	-0.555	-0.602	-0.648	-0.695	-0.742	-0.790
$\hat{E}_1 = -\ln \lambda_1$	0.114	0.011	-0.081	-0.167	-0.248	-0.325	-0.398
$\hat{h}_\epsilon = -\frac{L}{4\pi} \ln \frac{\lambda_1}{\lambda_0}$	0.496	0.496	0.497	0.498	0.498	0.498	0.498

Table 1: Numerical results for the energies of the vacuum and of the first excited states for different system sizes. The third row shows an estimation of the conformal dimension of the energy operator.

$$\langle \phi(il, -il)\phi(0,0)\rangle = \left[\frac{L}{\pi} \sin \frac{\pi l}{L}\right]^{-4h}. \tag{4.8}$$

The spin 2-point function was computed for various separations l and is shown in Fig.3. Fitting with the CFT prediction Eq.(4.8) we find $h_\sigma \approx 0.063$ which is about 1% from the theoretical value $h_\sigma = \frac{1}{16}$. To compare results for different sizes we remove the expected size dependent multiplicative constant $(\frac{L}{\pi})^{-4h_\sigma}$. Thus it can be seen that the CFT description is excellent except for $\frac{l}{L} \approx 0 \cong 1$ which corresponds to separations comparable to the lattice spacing. This is expected since the CFT description is valid as long as the lattice spacing can be neglected.

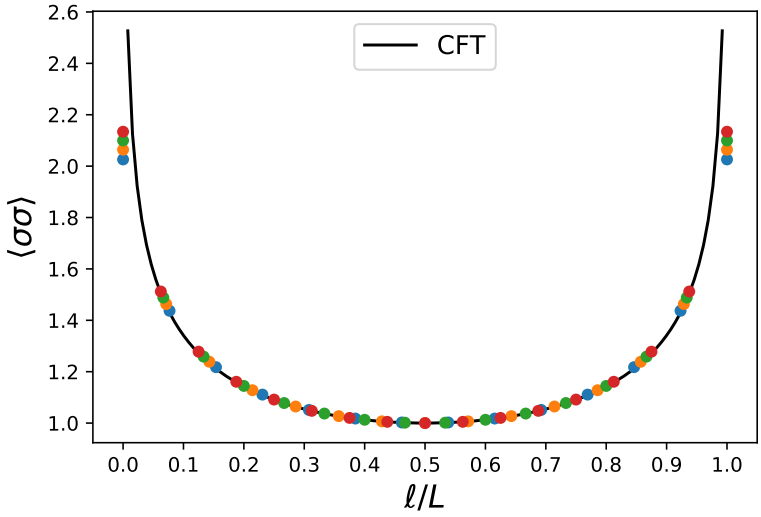

Figure 3: Comparison of the CFT prediction Eq.(4.8) (black line) and numerical computation (dots) for the spin 2-point function on the circumference of a cylinder of size L. Different colored dots correspond to different system sizes. Results for $L = 13$ (blue), 14 (yellow), 15 (green) and 16 (red) are shown on this figure.

The conformal dimension of the energy operator could be computed in a similar fashion from the energy 2-point function. Alternatively it can be computed from the finite-size

scaling of the gap between the vacuum energy and the first excited state. Indeed, the first excited state in the even $\mathbb{Z}_2$ sector is $|\epsilon\rangle = \epsilon(0,0)|0\rangle$. From the quantum Hamiltonian on the cylinder $\mathcal{H} = \frac{2\pi}{L}(L_0 + \bar{L}_0 - \frac{c}{12})$ we have

$$E_1 - E_0 = -\ln \frac{\lambda_1}{\lambda_0} = \frac{4\pi}{L} h_\epsilon \tag{4.9}$$

where we have just used $L_0|\epsilon\rangle = \bar{L}_0|\epsilon\rangle = h_\epsilon|\epsilon\rangle$. Performing the computation using data from Table 1 yields $h_\epsilon \approx 0.497$ consistent with the expected theoretical value $h_\epsilon = \frac{1}{2}$.

4.3 Entanglement entropies

In this section we describe the numerical computation of entanglement entropies in the Ising model. First we partition the Hilbert space as $A \times B$. In these computations we only considered the case where A consists of the states of a single interval of size l. If we let $\{|a_i\rangle\}_{i=0\ldots 2^l}$ and $\{|b_i\rangle\}_{i=0\ldots 2^{L-l}}$ be spin bases in A and B then we can write states as

$$|x\rangle = \sum_s x_s |s\rangle = \sum_{i=0}^{2^l} \sum_{k=0}^{2^{L-l}} x_{ik} |a_i b_k\rangle \tag{4.10}$$

with the components being related by $x_{ik} = x_s \iff s = i + 2^l k$. We calculate ρ_A directly as

$$(\rho_A)_{ij} = \mathrm{Tr}_B |x\rangle\langle x| = \sum_k \langle a_i b_k|x\rangle\langle x|a_j b_k\rangle = \sum_k x_{ik} x^*_{jk}. \tag{4.11}$$

This is of course more efficient than computing the full density matrix and then tracing over B. Having computed the reduced density matrix we can compute the entanglement entropies using the formulas given in Sect.3.

Numerical computations of the 2$^{\text{nd}}$ Rényi entropy in the vacuum and in the excited state $|\epsilon\rangle$ are shown in Fig. 4 and 5. To allow the comparison of results for different sizes we subtract the L dependent constant expected from the CFT predictions Eq.(3.16) and (3.31). In both cases, the agreement with the CFT prediction is excellent. This is quite remarkable since (i) the size of the lattice is rather modest ($\leq 16 \times 16$ sites) and (ii) a priori the CFT model we are using describes the low-energy physics of the critical Ising model on an infinite square lattice.

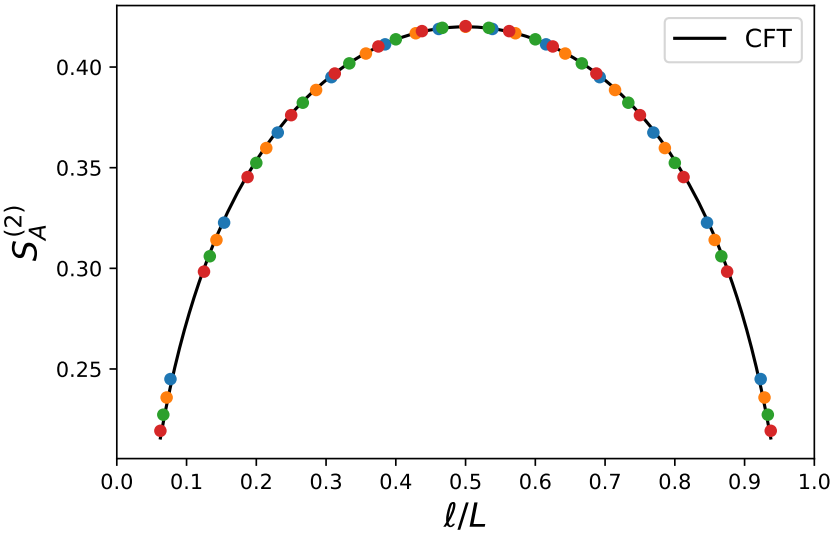

Figure 4: Comparison of the CFT prediction Eq.(3.16) (black line) and numerical computation (dots) for the 2nd Rényi entropy in the vacuum state. Results for $L = 13$ (blue), 14 (yellow), 15 (green) and 16 (red) are shown on this figure.

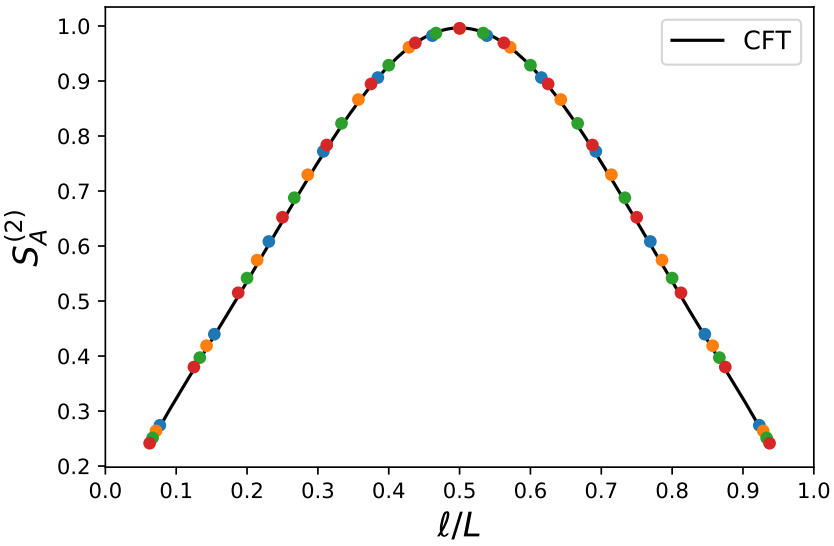

Figure 5: Comparison of the CFT prediction Eq.(3.31) (black line) and numerical computation (dots) for the 2nd Rényi entropy in the state $|\epsilon\rangle$. Results for $L = 13$ (blue), 14 (yellow), 15 (green) and 16 (red) are shown on this figure.

5 Conclusion

The rich structure of 2d CFTs enables one to choose between several analytical methods to compute Rényi entanglement entropies. We explored the replica approach under different lights, -with a geometrical method formulated in terms of Riemann surfaces with conical singularities, -with an algebraic method taking advantage of the orbifold Virasoro algebra and with a hybrid formalism alternating the geometric and orbifold points of view. The results obtained within CFT agree very well with simulations of the Ising model on a lattice, even for relatively small systems. Such a good agreement is quite remarkable since CFT only captures the infrared physics of the infinite Ising model.

With more time it would have been interesting to further explore the algebraic picture and orbifold algebras. One could also consider numerically computing entanglement entropies in other systems and for more complicated partitions.

Acknowledgements

I am thankful to my supervisors, Yacine and Benoit, for coming up with this interesting project and for their continuous support despite the difficult circumstances.

References

[1] P. Di Francesco, P. Mathieu and D. Senechal, *Conformal Field Theory*, Graduate Texts in Contemporary Physics. Springer-Verlag, New York, 1997, 10.1007/978-1-4612-2256-9.

[2] B. Estienne and Y. Ikhlef, *Lecture notes on conformal field theory for 2d Statistical Mechanics*. 2020.

[3] P. Calabrese, J. Cardy and B. Doyon, *Entanglement entropy in extended quantum systems*, *Journal of Physics A: Mathematical and Theoretical* **42** (2009) 500301.

[4] P. Calabrese and J. Cardy, *Entanglement entropy and conformal field theory*, *Journal of Physics A: Mathematical and Theoretical* **42** (2009) 504005.

[5] L. J. Dixon, D. Friedan, E. J. Martinec and S. H. Shenker, *The Conformal Field Theory of Orbifolds*, *Nucl. Phys. B* **282** (1987) 13.

[6] P. K. Kythe, *Handbook of conformal mappings and applications*. CRC Press, 2019.

[7] O. Lunin and S. D. Mathur, *Correlation functions for M^N/S_N orbifolds*, *Commun. Math. Phys.* **219** (2001) 399 [hep-th/0006196].

[8] T. Dupic, B. Estienne and Y. Ikhlef, *Entanglement entropies of minimal models from null-vectors*, *SciPost Phys.* **4** (2018) 031 [1709.09270].

[9] Y. Saad, *Numerical methods for large eigenvalue probelms*. Manchester University Press, 1992, Available here.

A Appendix: Python Code

```
from __future__ import division
import numpy as np
from numpy import random as rd
from scipy import linalg

##### Basic setup of the Ising model on the cylinder #####

 #Criticial value of the coupling
J = 0.5*np.log(1+np.sqrt(2))

#Spin chain object with a configuration and size attributes
class Col:
    def __init__(self, size, config):
        self.size = size
        self.config = config

#Configurations are labeled by an integer k=0,...,2^N-1
# Conversion of a configuration label k into column vector of
   up/down spins
# uses the binary expression of k
# Ex: N=6 spins, configuration 42=101010 -> (+-+-+-)
    def give_config(self):
        form = '{:0>%s}' %str(self.size)
        c = form.format(format(self.config,'b'))
        binconfig = map(int,c)
        for n, i in enumerate(binconfig):
            if i == 0:
                binconfig[n] = -1
        return binconfig

    def qstate(self):
# returns the quantum state corresponding to a given
   configuration
        qstate = np.zeros(2**self.size)
        qstate[self.config] = 1
        return qstate

    def spinflip(self,site):
#flips one spin of a given quantum state
        if self.give_config()[site]==-1:
```

```
39              self.config+=2**(self.size-site-1)
40          else:
41              self.config+=-2**(self.size-site-1)
42          return self.qstate()
43
44
45  def interact(s1,s2):
46  #Interaction function. Energies are shifted by 1J for
        convenience
47      if s1*s2==1:
48          E = 0
49      else:
50          E = -2
51      return E
52
53  #####  Transfer matrix #####
54
55  def p_on_x(size, site, x):
56  #Action of the elementary matrix p
57  #Recall Boltzmann weights: (1,exp(-2J))
58      px=np.zeros(2**size)
59      c=Col(size,0)
60      for i in range(2**size):
61          c.config=i
62          s=c.give_config()
63          px[i]=x[i]*np.exp(0.5*J*interact(s[site],s[(site+1)%
        size]))
64      return px
65
66  def q_on_x(size, site, x):
67  #Action of the elementary matrix q
68      qx=np.zeros(2**size)
69      c=Col(size,0)
70      for i in range(2**size):
71          c.config=i
72          qx[i]+=x[i]*(np.exp(J*interact(1,1)))+x[i-c.
        give_config()[site]*2**(c.size-site-1)]*np.exp(interact
        (1,-1)*J)
73      return qx
74
75  def pinv_on_x(size, site, x):
76      px=np.zeros(2**size)
77      c=Col(size,0)
```

```
78      for i in range(2**size):
79          c.config=i
80          s=c.give_config()
81          px[i]+=x[i]*np.exp(-0.5*J*interact(s[site],s[(site+1)%
    size]))
82      return px
83
84  def qinv_on_x(size, site, x):
85      qx=np.zeros(2**size)
86      c=Col(size,0)
87      for i in range(2**size):
88          c.config=i
89          qx[i]+=x[i]*(1/(1-np.exp(2*J*interact(1,-1))))*(np.exp
    (J*interact(1,1))) - x[i-c.give_config()[site]*2**(c.size-
    site-1)]*np.exp(interact(1,-1)*J)
90      return qx
91
92  def T_on_x(size,x):
93  #Action of the transfer matrix from the elementary matrices p,
    q
94      for i in range(size):
95          x = p_on_x(size, i, x)
96      for j in range(size):
97          x = q_on_x(size, j, x)
98      for k in range(size):
99          x = p_on_x(size, k, x)
100     return x
101
102 def Tinv_on_x(size,x):
103     for i in range(size):
104         x = pinv_on_x(size, size-1-i, x)
105     for j in range(size):
106         x = qinv_on_x(size, size-1-j, x)
107     for k in range(size):
108         x = pinv_on_x(size, size-1-k, x)
109     return x
110
111 def sigma_on_x(size, site, x):
112 #Action of the spin operator S(0, site)
113     sx=np.zeros(2**size)
114     c=Col(size,0)
115     for i in range(2**size):
116         c.config=i
```

```
117         sx[i]=x[i]*c.give_config()[site]
118     return sx
119 
120 def S_on_x(size,m,n,x):
121 #Heisenberg picture spin operator S(m,n)
122     sx = x
123     c = Col(size,0)
124     for i in range(m):
125         sx = T_on_x(size,sx)
126     for j in range(2**size):
127         c.config = j
128         sx[j] = x[j]*c.give_config()[n]
129     for k in range(m):
130         sx = Tinv_on_x(size, sx)
131     return sx
132 
133 def Z2_on_x(size, x):
134 #Z_2 inversion, all the spins are flipped
135     xflipped=np.zeros(2**size)
136     for k in range(2**size):
137         xflipped[k]=x[-k-1]
138     return xflipped
139 
140 
141 def dom_eval(size, prec, Z2project):
142 #Returns the largest eigenvalue of the transfer matrix and its
    eigenvector
143 
144     new_vect = rd.rand(2**size)
145     delta = 1
146     iterations = 0
147 
148     while delta > prec:
149         old_vect = new_vect
150         if Z2project==True:
151 #Projection onto the even Z_2 sector
152             new_vect = 0.5*new_vect + 0.5*Z2_on_x(size,
    new_vect)
153 
154         new_vect = T_on_x(size, new_vect)
155         lambda0 = np.sqrt(np.dot(new_vect, new_vect))
156         new_vect = (1/np.sqrt(np.dot(new_vect, new_vect)))*
    new_vect
```

```
157         delta = np.sqrt(np.dot((new_vect - old_vect), (
    new_vect - old_vect)))
158         iterations+=1
159 
160     print(np.log(lambda0)/size, delta, iterations)
161     return lambda0, new_vect
162 
163 def subdom_eval(size, prec, Z2project):
164  #Returns the first two largest eigenvalue of the transfer
     matrix  and their eigenvectors
165 
166     delta = 1
167     iterations = 0
168     new_vect = rd.rand(2**size)
169     lambda0, u0 = dom_eval(size, prec, Z2project)
170 
171     while delta > prec:
172         old_vect = new_vect
173         new_vect = T_on_x(size, new_vect)
174         if Z2project==True:
175             new_vect = 0.5*new_vect + 0.5*Z2_on_x(size,
    new_vect)
176 
177 
178         new_vect = new_vect - np.dot(u0, new_vect)*u0
179         lambda1 = np.sqrt(np.dot(new_vect, new_vect))
180         new_vect = (1/lambda1)*new_vect
181 
182         delta = np.sqrt(np.dot((new_vect-old_vect), (new_vect
    - old_vect)))
183 
184         iterations+=1
185     print(np.log(lambda1)/size, delta, iterations)
186     return lambda1, new_vect, lambda0, u0
187 
188 ##### Spin 2-point functions in the vacuum #####
189 
190 def calc_Z(L, M, prec):
191 #returns the partition function and the vacuum state
192     if M%2==1:
193         print("error M must be even")
194     else:
195         v = u = dom_eval(L, prec, False)[1]
```

```
196         for i in range(M):
197             u = T_on_x(L, u)
198         Z = np.vdot(v,u)
199     return Z, v
200 
201 def twopfunc(L, M, i, j, k, l, prec):
202 #Computes <0|S(i,j)S(k,l)|0> for a cylinder of circumference L
        and length M
203     if M%2==1:
204         print("error M must be even")
205     else:
206         Z, v = calc_Z(L, M, prec)
207         u = v
208         for a in range(M//2):
209             u = T_on_x(L,u)
210         u = S_on_x(L, k, l, u)
211         u = S_on_x(L, i, j, u)
212         for b in range(M//2):
213             u = T_on_x(L, u)
214         c = (1/Z)*np.vdot(v, u)
215     return c
216 
217 def fulltwopfunc(L, M, prec):
218 #Computes <0|S(0,0)S(0,l)|0> along a circumference of the
       cylinder
219     if M%2==1:
220         print("error M must be even")
221     else:
222         c=np.zeros(L)
223         Z, v = calc_Z(L, M, prec)
224         for l in range(L):
225             u = v
226             for a in range(M//2):
227                 u = T_on_x(L,u)
228             u = S_on_x(L, 0, l, u)
229             u = S_on_x(L, 0, 0, u)
230             for b in range(M//2):
231                 u = T_on_x(L, u)
232             c[l] = (1/Z)*np.vdot(v, u)
233     return c
234 
235 #####  Reduced density matrix #####
236 
```

```
237 # partition of H = 0...m...L-1 in AxB with A=0...m-1, B=m...L
    -1 ,  dim(A)=m, dim(B)=L-m,   rho_A: 2**m x 2**m
238
239 def reduced_x(L, sizeofA, x, i):
240 #Given a state x, only keeps the components x_ik pour k
    =0...2**L-m
241     x_red=[]
242     for j in range(2**L):
243         if (j-i)%2**sizeofA==0:
244             x_red.append(x[j])
245     return np.array(x_red)
246
247 def red_dens_mat(x, m):
248 #Compute the reduced density matrix rho_A
249     L = int(np.log(np.size(x))/np.log(2))
250     rhoA = np.zeros([2**m, 2**m])
251
252     for i in range(2**m):
253         for j in range(2**m):
254             x1 = reduced_x(L, m, x, i)
255             x2 = reduced_x(L, m, x, j)
256             rhoA[i,j] = np.dot(x1, x2)
257     return rhoA
258
259 #####  Von Neumann entropy #####
260
261 def vnentropy(state, l):
262 #Computes the VN entropy from the eigenvalues of rhoA
263     L=int(np.log(np.size(state))/np.log(2))
264     rho = red_dens_mat(state, l)
265     ev = linalg.eigvalsh(rho)
266     condition = ev >= 0
267 #Some eigenvalues of rhoA maybe slightly negative due to
    machine precision
268 #these are discarded
269     S = np.sum(-ev[condition]*np.log(ev[condition]))
270     return S
271
272 def plotneumann(state):
273     L = int(np.log(np.size(state))/np.log(2))
274     S = np.zeros(L-1)
275     for l in range(1, L//2+1):
276         S[l-1] = S[L-l-1] = vnentropy(state, l)
```

```
277     return S
278 
279 #####  Renyi entropies #####
280 
281 def rentropy(state, l, n):
282 #Compute the nth Renyi entropy
283     L = int(np.log(np.size(state))/np.log(2))
284     rho = red_dens_mat(state, l)
285     S = (1/(1-n))*np.log(np.trace(np.linalg.matrix_power(rho,n
    )))
286     return S
287 
288 def plotrenyi(state, n):
289     L = int(np.log(np.size(state))/np.log(2))
290     S = np.zeros(L-1)
291     for l in range(1, L//2+1):
292         S[l-1] = S[L-l-1] = rentropy(state, l, n)
293     return S
```