Practical Fermionic Gaussian States

Fermionic Gaussian States: practical numerical simulations

Author: Jacopo Surace

Institute: Strathclyde, ICFO

Date: February 26, 2020

Version: 1.00

Julia's code on Github

Contents

Chapter 1 The canonical anticommutation relations

1.1 The Hilbert space characterised by the canonical anticommutation relations

Consider a set of operators $\{a_i\}_{i=1}^N$ acting on a Hilbert space $\mathcal{H}$. We say that these operators satisfy the *canonical anticommutation relation* (CAR) when they satisfy

$$\{a_i, a_j^\dagger\} = \mathbb{I}\delta_{i,j}; \qquad \{a_i, a_j\} = 0, \tag{1.1}$$

with $\{a, b\} := ab + ba$ the notation for the anticommutator.
As shown in **?** a number of properties of the set of operators $\{a_i\}_{i=1}^N$ and of the Hilbert space $\mathcal{H}$ can be inferred just by the fact that such operators exist and obey the CAR.
The $a_i^\dagger a_i$ are a set of *commuting, hermitian, positive operators* with eigenvalues $\{0, 1\}$. We denote with $\vec{x} \in \{0, 1\}^N$ a binary string of length N with the i-th elements x_i. With $|\vec{x}\rangle$ we identify one of the 2^N states that is the simultaneous eigenstate of $a_i^\dagger a_i$ for all $i = 1, \ldots, N$ with eigenvalues respectively x_i. The operator a_i acts as a *lowering operator* for $a_i^\dagger a_i$ and $a_i^\dagger$ acts as a *raising operator* for $a_i^\dagger a_i$ in the sense that

1. If $a_i^\dagger a_i|\vec{x}\rangle = |\vec{x}\rangle$, that is, it has the i-th eigenvalue equal to 1. Then the action of a_i on $|\vec{x}\rangle$ lower the corresponding eigenvalue, meaning that $a_i^\dagger a_i(a_i|\vec{x}\rangle) = 0(a_i|\vec{x}\rangle)$.
2. If $a_i^\dagger a_i|\vec{x}\rangle = 0|\vec{x}\rangle$, that is, it has the i-th eigenvalue equal to 0. Then the action of $a_i^\dagger$ on $|\vec{x}\rangle$ raise the corresponding eigenvalue, meaning that $a_i^\dagger a_i(a_i^\dagger|\vec{x}\rangle) = 1(a_i^\dagger|\vec{x}\rangle)$.

We define an *ordering* by explicitly defining $|\vec{x}\rangle := (a_1^\dagger)^{x_1}(a_2^\dagger)^{x_2} \ldots (a_N^\dagger)^{x_N}|\vec{0}\rangle$, where $\vec{0}$ is the string of N zeros. The set $\{|\vec{x}\rangle\}_{\vec{x}\in\{0,1\}^N}$ form an orthonormal basis. If the dimension of the Hilbert space $\mathcal{H}$ is 2^N, then $\{|\vec{x}\rangle\}_{\vec{x}\in\{0,1\}^N}$ is an orthonormal basis of $\mathcal{H}$.
The action of the raising and lowering operators on $|\vec{x}\rangle$ is then

$$a_i|\vec{x}\rangle = \begin{cases} -(-1)^{S_{\vec{x}}^i}|\vec{x}'\rangle \text{ with } x_i' = 0 \text{ and } x_{j\neq i}' = x_{j\neq i}, & \text{if } x_i = 1 \\ 0 & \text{if } x_i = 0 \end{cases}, \tag{1.2}$$

$$a_i^\dagger|\vec{x}\rangle = \begin{cases} 0 & \text{if } x_i = 1 \\ -(-1)^{S_{\vec{x}}^i}|\vec{x}'\rangle \text{ with } x_i' = 1 \text{ and } x_{j\neq i}' = x_{j\neq i}, & \text{if } x_i = 0 \end{cases}, \tag{1.3}$$

with $S_{\vec{x}}^i = \sum_{k=1}^{i-1} x_k$.
In appendix **??** we report some useful equalities valid for operators satisfying the CAR.

1.2 Dirac and Majorana representations

The raising and lowering operators $a_i^\dagger$,a_i are called *Dirac operators* and they represent the action of adding and removing the i-th fermionic mode.

Both a_i and its adjoint $a_i^\dagger$ are not hermitian. The hermitian combinations of the raising and lowering operators

$$x_i = \frac{a_i + a_i^\dagger}{\sqrt{2}}, \qquad p_i = \frac{a_i - a_i^\dagger}{i\sqrt{2}}, \tag{1.4}$$

are called *Majorana operators*.

The inverse transformations are:

$$a_i = \frac{x_i + ip_i}{\sqrt{2}}, \qquad a_i^\dagger = \frac{x_i - ip_i}{\sqrt{2}}. \tag{1.5}$$

In terms of Majorana operators the CARs read as

$$\{x_i, x_j\} = \{p_i, p_j\} = \delta_{i,j}, \qquad \{x_i, p_j\} = 0. \tag{1.6}$$

Remark Majorana operators labelled by i correspond Dirac operators labelled by i. Moving between Majorana and Dirac operators does not mix modes.

1.2.1 Vector notation

We can collect the Dirac operators of a system with N modes in the vector $\vec{\alpha}$ of length $2N$ defined as

$$\vec{\alpha} = \begin{pmatrix} a_0^\dagger \\ \vdots \\ a_{N-1}^\dagger \\ a_0 \\ \vdots \\ a_{N-1} \end{pmatrix}, \quad \vec{\alpha}^\dagger = \begin{pmatrix} a_0 & \dots & a_{N-1} & a_0^\dagger & \dots & a_{N-1}^\dagger \end{pmatrix}. \tag{1.7}$$

Analogously we can collect the Majorana operators in the vector $\vec{r}$ defined as

$$\vec{r} = \begin{pmatrix} x_0 \\ \vdots \\ x_{N-1} \\ p_0 \\ \vdots \\ p_{N-1} \end{pmatrix}, \tag{1.8}$$

in terms of $\vec{r}$ the CAR are conveniently written as

$$\{r_i, r_j\} = \delta_{i,j}. \tag{1.9}$$

We define the unitary matrix Ω as

$$\Omega = \frac{1}{\sqrt{2}}\begin{pmatrix} \mathbb{I} & \mathbb{I} \\ i\mathbb{I} & -i\mathbb{I} \end{pmatrix} \quad , \quad \Omega^\dagger = \Omega^{-1} = \frac{1}{\sqrt{2}}\begin{pmatrix} \mathbb{I} & -i\mathbb{I} \\ \mathbb{I} & i\mathbb{I} \end{pmatrix} . \tag{1.10}$$

Such a matrix, applied to the vector of the Dirac operators $\vec{\alpha}$, returns the vector of Majorana operators $\vec{r} = \Omega\vec{\alpha}$.

Remark The $\{r_i\}_{i=1,\dots,2N}$ are hermitian, traceless and form a Clifford algebra C_{2N} characterised by the MCAR (**??**).

1.2.2 Fermionic transformation

A transformation O is said to respect the CAR in the Majorana representation if maps a vector of Majorana operators $\vec{r}$ to a new one $\vec{s} = O\vec{r}$. That is for all $i, j = 1, 2, \dots, N$ it must hold

$$\delta_{i,j} = \{s_i, s_j\} = \sum_{k,l} O_{i,k}O_{j,l}\{r_k, r_l\} = (OO^T)_{i,j}, \tag{1.11}$$

that corresponds to the request for O to be an orthogonal matrix.

A transformation U of a vector of Dirac operators $\vec{\alpha}$ to a new one $\vec{\beta} = U\vec{\alpha}$ that respects the CAR, we call it a *fermionic transformation*, has the form of $U = \Omega^\dagger O\Omega$ with O an orthogonal matrix.

Chapter 2 Fermionic Quadratic Hamiltonians

2.1 Dirac Representation

The general *fermionic quadratic hamiltonians* on a finite lattice of N sites in the Dirac operators representation can be written as

$$\hat{H} = \frac{1}{2}\sum_{i,j=1}^{N-1}\left(A_{i,j}a_i^\dagger a_j - \bar{A}_{i,j}a_i a_j^\dagger + B_{i,j}a_i a_j - \bar{B}_{i,j}a_i^\dagger a_j^\dagger\right), \tag{2.1}$$

where A is an *hermitian*, $A^\dagger = A$, and B is a *skew-symmetric*, $B^T = -B$, complex matrix. Defining the matrix

$$H = \begin{pmatrix} -\bar{A} & B \\ -\bar{B} & A \end{pmatrix}, \tag{2.2}$$

the compact form of equation (**??**) reads

$$\hat{H} = \frac{1}{2}\vec{\alpha}^\dagger H \vec{\alpha}. \tag{2.3}$$

2.2 Majorana Representation

The Majorana representation of the generic fermionic quadratic hamiltonian reads as

$$\hat{H} = \frac{i}{2}\vec{r}^\dagger h \vec{r}, \tag{2.4}$$

where

$$ih = \Omega H \Omega^\dagger = i\begin{pmatrix} \Im\{A+B\} & \Re\{A+B\} \\ \Re\{B-A\} & \Im\{A-B\} \end{pmatrix}. \tag{2.5}$$

Where $\Im\{\cdot\}$ and $\Re\{\cdot\}$ are respectively the imaginary and the real part of their argument. Using the properties of matrices A and B, it is easy to see that matrix $\Omega H \Omega^\dagger$ is purely imaginary and hermitian, therefore matrix h is real and skew-symmetric.

2.3 Diagonalisation

2.3.1 Hamiltonian diagonal form with Dirac operators

Given a particular fermionic quadratic hamiltonian H in the general form (**??**) it is always possible to find a new set of Dirac operators $\{b_k\}_{k=1}^N$ such that H in terms of $\{b_k\}_{k=1}^N$ reads as

$$\hat{H} = \frac{1}{2}\sum_{k=1}^{N} \epsilon_k (b_k^\dagger b_k - b_k b_k^\dagger), \tag{2.6}$$

with $\epsilon_k \in \mathbb{R}$ for all $k = 1, 2, \ldots, N$ **?**.

We call hamiltonians in this form free-free fermion hamiltonians.

In compact form

$$\hat{H} = \frac{1}{2}\vec{\beta}^\dagger H_D \vec{\beta} \tag{2.7}$$

with

$$H_D = U^\dagger H U = \begin{pmatrix} -\epsilon_1 & 0 & \ldots & & \ldots & 0 \\ 0 & \ddots & \ddots & & & \vdots \\ \vdots & \ddots & -\epsilon_N & & & \\ & & & \epsilon_1 & \ddots & \vdots \\ \vdots & & & \ddots & \ddots & 0 \\ 0 & \ldots & & & \ldots 0 & \epsilon_N \end{pmatrix}, \tag{2.8}$$

where U is a fermionic tranformation.

We will always order the eigenvalues in descending order ($\epsilon_1 \geq \epsilon_2 \geq \cdots \geq \epsilon_N$).

2.3.2 Hamiltonian diagonal form with Majorana operators

In terms of Majorana operators the diagonal form of a generic fermionic quadratic hamiltonian reads as

$$\hat{H} = \frac{i}{2}\sum_{i=1}^{N} \lambda_i (\tilde{x}_i \tilde{p}_i - \tilde{p}_i \tilde{x}_i). \tag{2.9}$$

for a set of Majorana operators $\{\tilde{x}_i\}_i$, $\{\tilde{p}_i\}_i$. In compact form

$$\hat{H} = \frac{i}{2}\vec{s}^\dagger h_d \vec{s} \tag{2.10}$$

with

$$h_d = O^T h O = \bigoplus_{i=1}^{N} \begin{pmatrix} 0 & \lambda_i \\ -\lambda_i & 0 \end{pmatrix} \tag{2.11}$$

a block diagonal matrix and O an orthogonal transformation .

2.4 Numerical diagonalisation

We want to find the orthogonal transformation O that diagonalise a fermionic quadratic hamiltonian in Majorana representation as in (**??**). For a more physical approach we refer to **?**, here we focus on the numerical approach.

The following theorem is a standard result in matrix theory **??**

Theorem 2.1. Block diagonal form of real, skew-symmetric matrices

Let h be $2N \times 2N$ a real, skew-symmetric matrix. There exist an orthogonal matrix O such that

$$h = O\Sigma h_D O^T, \tag{2.12}$$

with h_D a block diagonal matrix of the form

$$h_D = \bigoplus_{i=1}^{N} \begin{pmatrix} 0 & \lambda_i \\ -\lambda_i & 0 \end{pmatrix} \tag{2.13}$$

for real, positive definite $\{\lambda_i\}_{i=1,\ldots,N}$. The non-zero eigenvalues of matrix h are the imaginary numbers $\{\pm i\lambda_i\}_{i=1,\ldots,N}$. ♡

Matrix h in (**??**) is real, skew-symmetric, thus, using theorem **??** we find the orthogonal transformation O that diagonalise the matrix

$$\hat{H} = \frac{i}{2}\vec{r}^{\dagger} h \vec{r} = \frac{i}{2}\left(\vec{r}^{\dagger} O^{\dagger}\right)\left(O \bigoplus_{i=1}^{N} \begin{pmatrix} 0 & \lambda_i \\ -\lambda_i & 0 \end{pmatrix} O^{\dagger}\right)\left(O\vec{r}\right) = \tag{2.14}$$

$$= \frac{i}{2}\vec{s}^{\dagger}\left(O \bigoplus_{i=1}^{N} \begin{pmatrix} 0 & \lambda_i \\ -\lambda_i & 0 \end{pmatrix} O^{\dagger}\right)\vec{s} = \frac{i}{2}\sum_{i=0}^{N-1} \lambda_i (\tilde{x}_i \tilde{p}_i - \tilde{p}_i \tilde{x}_i) \tag{2.15}$$

We defined the new collection of Majorana operators $\vec{s} = O\vec{r}$

$$\vec{s} = \begin{pmatrix} \tilde{x}_0 \\ \tilde{p}_0 \\ \tilde{x}_1 \\ \tilde{p}_1 \\ \vdots \\ \tilde{x}_{N-1} \\ \tilde{p}_{N-1} \end{pmatrix}. \tag{2.16}$$

2.4.1 Block-diagonal form of real skew-symmetric matrices

The matrix decomposition (**??**) of theorem **??** is numerically obtained in 3 steps

1. Compute numerically a Schur decomposition (Schur triangularisation as in **?**) of the skew-symmetric matrix h such that: $h = \tilde{O}\tilde{h}_D\tilde{O}^T$. The matrix $\tilde{h}_D$ should be a block-diagonal

matrix with each block in the anti-diagonal form

$$\begin{pmatrix} 0 & \tilde{\lambda}_i \\ -\tilde{\lambda}_i & 0 \end{pmatrix}, \tag{2.17}$$

it is not guaranteed that $\tilde{\lambda}_i$ is positive for each i. It is necessary to reorder it.

2. Build the ortogonal matrix $S = \bigoplus_{i=1}^{\lfloor N/2 \rfloor} s_i$ with

$$s_i = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{2.18}$$

if $\tilde{\lambda}_i < 0$ or

$$s_i = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \tag{2.19}$$

if $\tilde{\lambda}_i > 0$.

3. The final orthogonal transformation is $O = \tilde{O}S$ such that $h = Oh_D O^T$.

F-utilities Routine 2.1. `Diag_Real_Skew(h)` $\longrightarrow h_D, O$

This function implements the algorithm for the block diagonalisation of h a generic skew-symmetric real matrix. h_D is the block-diagonal matrix of (**??**) *and has the following property: it is in the block diagonal form, each 2×2 block is skew-symmetric with the upper-right element positive and real and h_D is in ascending order for the upper diagonal. O is an orthogonal matrix such that: $h = Oh_D O^T$.* ♠

2.4.2 Diagonalisation with Dirac operators

Considering the Dirac representation one is interested in a transformation from Dirac modes $\vec{\alpha}$ in Dirac modes $\vec{\beta}$ such that the generic quadratic Hamiltonian (**??**) is mapped to the free-free fermions Hamiltonian $\hat{H} = \frac{1}{2}\sum_{k=0}^{N-1} \epsilon_k \left(b_k^\dagger b_k - b_k b_k^\dagger \right)$.
To do so it is sufficient to map the Majoranas $\vec{s}$ back to Fermions to obtain (**??**). The values $\tilde{h}_i$ of (**??**) and ϵ_k of (**??**) are related by

$$\epsilon_k = \lambda_k. \tag{2.20}$$

In order to obtain the diagonal form of H one can use the following algorithm:

1. Obtain matrix h from H as $h = -i\Omega H \Omega^\dagger$.
2. Apply the Majorana diagonalisation of section **??** to h to obtain its block diagonal form and the orthogonal transformation O such that $\vec{s} = O\vec{r}$.
3. In order to move back to the Dirac representation one has to pay attention to how the Majorana operators are ordered in $\vec{s}$. In fact, in $\vec{s}$ the order is xp (see (**??**)), so we cannot transform back to the Dirac representation simply applying Ω again. To reorder $\vec{s}$ to order

xx (see (**??**)) we define the $2N \times 2N$ matrix

$$F_{xp\to xx} = \begin{array}{c} i=0 \\ i=1 \\ \vdots \\ \vdots \\ i=N \\ i=N+1 \\ \vdots \\ i=2N+1 \end{array} \begin{pmatrix} 1 & 0 & 0 & 0 & \dots & \dots & 0 & 0 \\ 0 & \vdots & 1 & \vdots & & & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & & & 0 & \vdots \\ \vdots & 0 & 0 & 0 & & & 1 & \vdots \\ \vdots & 1 & 0 & 0 & & & 0 & \vdots \\ \vdots & 0 & \vdots & 1 & & & \vdots & \vdots \\ \vdots & \vdots & \vdots & 0 & & & \vdots & 0 \\ 0 & 0 & 0 & \vdots & \dots & \dots & 0 & 1 \end{pmatrix} \tag{2.21}$$

so that $F_{xp\to xx}\vec{s}$ has the order xx as $\vec{r}$ and we get

$$\tilde{h}_D = F_{xp\to xx} h_D F^T_{xp\to xx} = \begin{pmatrix} 0 & \dots & 0 & \lambda_0 & 0 & 0 \\ \vdots & \ddots & \vdots & 0 & \ddots & 0 \\ 0 & \dots & 0 & 0 & 0 & \lambda_{\lfloor N/2 \rfloor} \\ -\lambda_0 & 0 & 0 & 0 & \dots & 0 \\ 0 & \ddots & 0 & \vdots & \ddots & \vdots \\ 0 & 0 & -\lambda_{\lfloor N/2 \rfloor} & 0 & \dots & 0 \end{pmatrix} \tag{2.22}$$

4. . Move from Majoranas to Dirac representation

$$H_D = i\Omega^\dagger \tilde{h}_D \Omega = \begin{pmatrix} \lambda_0 & & & & & \\ & \ddots & & & & \\ & & \lambda_{\lfloor N/2 \rfloor} & & & \\ & & & -\lambda_0 & & \\ & & & & \ddots & \\ & & & & & -\lambda_{\lfloor N/2 \rfloor} \end{pmatrix} \tag{2.23}$$

5. . The final unitary transformation $\vec{\alpha} = U\vec{\beta}$ read as

$$U = \Omega^\dagger \cdot O \cdot F^\dagger_{xp\to xx} \cdot \Omega \tag{2.24}$$

6. . Finally we have $H = UH_DU^\dagger$.

F-utilities Routine 2.2. `Diag_ferm`$(H) \to H_D, U$

This function diagonalise H with the fermionic algorithm of section **??**. *H_D is the diagonal form with the first half diagonal negative and the second one positive. U is the fermionic transformation such that: $M = U_f M_f U_f^\dagger$.*

♠

Appendix Extended calculations

A.1 Eigenvalues of Γ and H_α

We consider the state $\rho = \frac{e^{-\frac{1}{2}\vec{\alpha}^\dagger M_\rho \vec{\alpha}}}{Z}$, we diagonalise M_ρ changing the basis to $\vec{\beta} = U^\dagger \vec{\alpha}$. Thus we have

$$\rho = \frac{e^{-\frac{1}{2}\vec{\beta}^\dagger M_D \vec{\beta}}}{Z} = \frac{e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^\dagger b_k - b_k b_k^\dagger\right)}}{Z}. \tag{A.1}$$

We change the basis of the correlation matrix too

$$\Gamma^b_{i,j} = \left(U^\dagger \Gamma U\right)_{i,j} = Tr\left[\rho \vec{\beta}_i \vec{\beta}_j^{\,\dagger}\right]. \tag{A.2}$$

Now we want to explicitly compute the elements of Γ^b. First of all we compute the normalisation constant

$$Z = Tr\left[e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^\dagger b_k - b_k b_k^\dagger\right)}\right] = \prod_{k=1}^{N-1}\left(2\cosh\left(\frac{\epsilon_k}{2}\right)\right). \tag{A.3}$$

To compute the numerator part this equalities will result useful

- For $x \neq y$

$$\begin{aligned}
Tr\left[e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^\dagger b_k - b_k b_k^\dagger\right)} b_x^\dagger b_y\right] &= \sum_{v\in\{0,1\}^N}\langle v| e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^\dagger b_k - b_k b_k^\dagger\right)} b_x^\dagger b_y |v\rangle = \\
&= \sum_{v\in\{0,1\}^N}\langle v| e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^\dagger b_k - b_k b_k^\dagger\right)} |\tilde{v}\rangle = \\
&= \sum_{v\in\{0,1\}^N} e^{-\frac{1}{2}\sum_{k=0}^{N-1}(-1)^{v_k+1}\epsilon_k}\langle v|\tilde{v}\rangle = 0
\end{aligned} \tag{A.4}$$

$$Tr\left[e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^\dagger b_k - b_k b_k^\dagger\right)} b_x b_y^\dagger\right] = 0 \tag{A.5}$$

-

- $\forall x, y$

$$Tr\left[e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^\dagger b_k - b_k b_k^\dagger\right)} b_x b_y\right] = 0 \tag{A.6}$$

$$Tr\left[e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^\dagger b_k - b_k b_k^\dagger\right)} b_x^\dagger b_y^\dagger\right] = 0 \tag{A.7}$$

Thus the numerator can be explicitly written as

$$Tr\left[e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^\dagger b_k - b_k b_k^\dagger\right)} \vec{\alpha}_i \vec{\alpha}_j^{\,\dagger}\right] = \tag{A.8}$$

$$= \sum_{l=1}^{2N}\sum_{m=1}^{2N} U_{i,l}U^{\dagger}_{m,j} Tr\left[e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^{\dagger}b_k - b_k b_k^{\dagger}\right)}\vec{\beta}_l\vec{\beta}_m^{\dagger}\right] =$$

$$= \sum_{l=1}^{N} U_{i,l}U^{\dagger}_{l,j} Tr\left[e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^{\dagger}b_k - b_k b_k^{\dagger}\right)}b_l^{\dagger}b_l\right] + \sum_{l=1}^{N} U_{i,l+N}U^{\dagger}_{l+N,j} Tr\left[e^{-\frac{1}{2}\sum_{k=0}^{N-1}\epsilon_k\left(b_k^{\dagger}b_k - b_k b_k^{\dagger}\right)}b_l b_l^{\dagger}\right] =$$

$$= \sum_{l=1}^{N} U_{i,l}U^{\dagger}_{l,j} e^{-\frac{\epsilon_l}{2}} \prod_{k\neq l} 2\cosh(\frac{\epsilon_k}{2}) + \sum_{l=1}^{N} U_{i,l+N}U^{\dagger}_{l+N,j} e^{\frac{\epsilon_l}{2}} \prod_{k\neq l} 2\cosh(\frac{\epsilon_k}{2})$$

I can divide by Z and obtain

$$\begin{aligned}
\Gamma_{i,j} &= \sum_{l=1}^{N} U_{i,l}U^{\dagger}_{l,j}\frac{e^{-\frac{\epsilon_l}{2}}}{e^{\frac{\epsilon_l}{2}} + e^{-\frac{\epsilon_l}{2}}} + \sum_{l=1}^{N} U_{i,l+N}U^{\dagger}_{l+N,j}\frac{e^{\frac{\epsilon_l}{2}}}{e^{\frac{\epsilon_l}{2}} + e^{-\frac{\epsilon_l}{2}}} \\
&= \sum_{l=1}^{N} U_{i,l}U^{\dagger}_{l,j}\frac{1}{1+e^{\epsilon_l}} + \sum_{l=1}^{N} U_{i,l+N}U^{\dagger}_{l+N,j}\frac{1}{1+e^{-\epsilon_l}} = \\
&= \sum_{l=1}^{2N} U_{i,l}U^{\dagger}_{l,j}\frac{1}{1+e^{\epsilon_l}} = U\Gamma^D U^{\dagger}.
\end{aligned} \tag{A.9}$$

So the same transformation U that moves to the free Hamiltonian M_D is also the transformation that diagonalise the correlation matrix. The eigenvalues ν_i of the correlation matrix Γ are related to the eigenvalues of the Hamiltonian M_ρ by

$$\nu_i = \frac{1}{1+e^{\epsilon_i}}, \tag{A.10}$$

$$\epsilon_i = ln\left(\frac{1-\nu_i}{\nu_i}\right), \tag{A.11}$$

since $\nu_i \in [0,1]$ the eigenvalues $\epsilon_i \in [-\infty, +\infty]$.

A.2 Purity

From the previous subsection we have:

$$Z^2 = \prod_{k=1}^{N-1}\left(2\cosh\left(\frac{\epsilon_k}{2}\right)\right)^2 \tag{A.12}$$

and

$$Tr\left[e^{-\sum_{k=0}^{N-1}\epsilon_k\left(b_k^{\dagger}b_k - b_k b_k^{\dagger}\right)}\right] = \prod_{k=1}^{N-1}\left(2\cosh\left(\epsilon_k\right)\right). \tag{A.13}$$

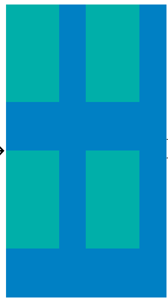
Thus the purity is:

$$\text{Purity} = \prod_{k=1}^{N-1} \frac{1}{\operatorname{sech}(\epsilon_k) + 1} \tag{A.14}$$

Appendix F-utilities

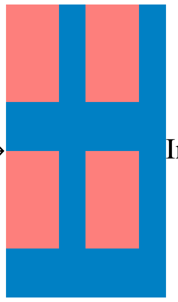
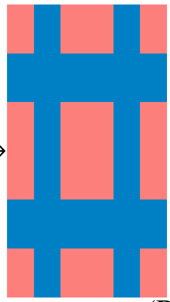
1. `Print_matrix(title, M)`: Print a graphical representation of matrix `M` in a figure called title.
2. `Build_Omega(N)`$\longrightarrow \Omega$: This function return the $2N \times 2N$ matrix Ω (**??**).
3. `Build_FxxTxp(N)`$\longrightarrow F_{xx\to xp}$: This function return the $2N \times 2N$ matrix $F_{xx\to xp}$ (**??**).
4. `Build_FxpTxx(N)`$\longrightarrow F_{xp\to xx}$: This function return the $2N \times 2N$ matrix $F^T_{xx\to xp}$.
5. `Diag_Real_Skew(h)`$\longrightarrow h_D, O$: This function implements the algorithm for the block diagonalisation of h a generic skew-symmetric real matrix. h_D is the block-diagonal matrix of (**??**) and has the following property: it is in the block diagonal form, each 2×2 block is skew-symmetric with the upper-right element positive and real and h_D is in ascending order for the upper diagonal. O is an orthogonal matrix such that: $h = O h_D O^T$.
6. `Diag_ferm(M)`$\longrightarrow M_f, U_f$:This function implement the fermionic algorithm of section **??** with $M = H_a$. M_f is the matrix we called H_D and it is in diagonal form with the first half diagonal negative and the second one positive. U_f is the orthogonal matrix that we called U and it is a unitary matrix such that: $M = U_f M_f U_f^\dagger$.
7. `Diag_Gamma(M)`$\longrightarrow M_f, U$ This function returns M_f, the the diagonal form (**??**) of the Dirac correlation matrix M and U the fermionic transformation such that $M = U M_f U\dagger$.
8. `Purity(M)`$\longrightarrow p$: This function takes as input the correlation matrix Γ and return a Float p from 0 to 1 that is the purity.
9. `Evolve_gamma(M,D,U,t)`$\longrightarrow M_t$: This function evolve for a time `t` (last argument) the correlation matrix Γ (first argument), `D` is the Dirac Hamiltonian diagonalised, U is the unitary transformation that change basis from the diagonal one to the original one. e.g. If I want to evolve the correlation matrix M with H_a for a time t I would write `M_t=Evolve_gamma(M,Diag_ferm(`H_a`),`t`)`.
10. `Evolve_gamma_imag(M,D,U,t)`$\longrightarrow M_t$: This function evolve for a time `t` (last argument) the pure state correlation matrix Γ (first argument) with the evolution **??**, `D` is the Dirac Hamiltonian diagonalised, U is the unitary transformation that change basis from the diagonal one to the original one .N.B. `M` must be a pure state (Purity(M) = 1), `Evolve_gamma_imag(M,D,U,t)` is a pure state.
11. `Energy_fermion(M,D,U)`$\longrightarrow e$: This function return the energy of the correlation matrix M with respect of the Hamiltonian in the diagonal form M where U is the change of basis from the diagonal one to the space one. e.g. If I want to compute the energy of the correlation matrix Γ with respect to the generic quadratic Hamiltonian represented by H I would write `Energy_fermion(`Γ`,Diag_ferm(`H`))`.
12. `Reduce_gamma(M,N_partition,first_index)` $\longrightarrow M_r$: This function return the corre-

lation matrix of a subsystem with `N_partition` size starting from the site at `first_index`. e.g. `Reduce_gamma()` return the green the element of the matrix $M_{6\times6}$

Reduce_gamma($M_{6\times6}$,2,1)⟶  Reduce_gamma($M_{6\times6}$,2,3)⟶ 

(B.1)

13. `Inject_gamma(gamma, injection, first_index)` $\to M_T$:This function overwrite the subsystem of gamma starting at `first_index` with the system with correlation matrix injection. The system of injection has to be smaller then the one of gamma. The returned system has same dimension of gamma. e.g. `Inject_gamma()` return the red and blue matrix where the elements in red are the one of the matrix injection.

Inject_gamma($M_{6\times6}$,2,1)⟶  Inject_gamma($M_{6\times6}$,2,3)⟶ 

(B.2)

14. `Eigenvalues_of_rho(M)` $\to \vec{e}$: This function take as input a correlation matrix Γ and return the vector with all its eigenvalues. N.B. The number of eigenvalues grows exponentially with the size of Γ.
15. `VN_Entropy(Γ)` $\to S$ This function return S, the `Float64` value of the Von Neumann Entropy of the state described by the Dirac correlation matrix Γ.

Appendix Useful relations

C.1 Pauli Matrices

1. $\sigma^+ = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$, $\sigma^- = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$, $\sigma^z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$, $\sigma^y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$, $\sigma^x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $|+\rangle_x = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix}$,
$|-\rangle_x = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -1 \end{pmatrix}$, $|+\rangle_y = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ i \end{pmatrix}$, $|-\rangle_y = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ -i \end{pmatrix}$, $|0_-\rangle_z = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, $|1_+\rangle_z = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$
2. $\sigma^z\sigma^- = -\sigma^-$
3. $\sigma^z\sigma^+ = \sigma^+$
4. $\sigma^-\sigma^z = \sigma^-$
5. $\sigma^+\sigma^z = -\sigma^+$
6. $\sigma^+\sigma^- = \frac{\sigma^z+\mathbb{I}}{2}$
7. $\sigma^-\sigma^+ = \frac{\mathbb{I}-\sigma^z}{2}$

C.2 Operators obeying CAR

1. $\{a_i, a_j^\dagger\} = \mathbb{I}\delta_{i,j} \qquad \{a_i, a_j\} = \{a_i^\dagger, a_j^\dagger\} = 0$
2. $a_i a_j = -a_j a_i; \qquad a_i^\dagger a_j^\dagger = -a_j^\dagger a_i^\dagger$
3. $a_i^2 = \left(a_j^\dagger\right)^2 = 0$
4. $a_i a_j^\dagger = \delta_{i,j} - a_j^\dagger a_i$
5. $a_i a_j = \frac{a_i a_j - a_j a_i}{2}$
6. $a_i a_j^\dagger = \frac{a_i a_j^\dagger - a_j^\dagger a_i}{2} + \frac{\delta_{i,j}}{2}$
7. $a_i^\dagger a_j = \frac{a_i^\dagger a_j - a_j a_i^\dagger}{2} + \frac{\delta_{i,j}}{2}$

Commutators

1. $[a_i^\dagger, a_j] = \delta_{i,j} - 2a_j a_i^\dagger = a_i^\dagger a_j - \delta_{i.j}$
2. $[a_i, a_j^\dagger] = \delta_{i,j} - 2a_j^\dagger a_i = a_i a_j^\dagger - \delta_{i,j}$
3. $[a_i, a_j] = 2a_i a_j$
4. $[a_i^\dagger, a_j^\dagger] = 2a_i^\dagger a_j^\dagger$

Majorana operators

1. $x_i^2 = p_i^2 = \frac{1}{2}$
2. $a^\dagger a = \frac{i}{2}(xp - px) + \frac{1}{2} = ixp + \frac{1}{2}$
3. $aa^\dagger = \frac{i}{2}(px - xp) + \frac{1}{2} = ipx + \frac{1}{2}$
4. $xp = -\frac{i}{2}\left(a^\dagger a - aa^\dagger\right) = -i\left(a^\dagger a - \frac{1}{2}\right)$

C.3 Jordan-Wigner Transformations

C.3.1 spinless fermions → spins

1. $a_j = -\bigotimes_{k=1}^{j-1} \sigma_k^z \otimes \sigma_j^- \bigotimes_{k=j+1}^{N} \mathbb{I}_k$
2. $a_j^\dagger = -\bigotimes_{k=1}^{j-1} \sigma_k^z \otimes \sigma_j^+ \bigotimes_{k=j+1}^{N} \mathbb{I}_k$
3. $a_j^\dagger a_j = \bigotimes_{k=1}^{j-1} \otimes \frac{\sigma_j^z + \mathbb{I}_j}{2} \bigotimes_{k=j+1}^{N} \mathbb{I}_k$

C.3.2 spins → spinless fermions

1. $\sigma_j^z = a_j^\dagger a_j - a_j a_j^\dagger$
2. $\sigma_j^x = -\bigotimes_{k=1}^{j-1} \sigma_j^z \otimes (a_j + a_j^\dagger) \bigotimes_{k=j+1}^{N} \mathbb{I}_j$
3. $\sigma_j^x = i \bigotimes_{k=1}^{j-1} \sigma_j^z \otimes (a_j^\dagger - a_j) \bigotimes_{k=j+1}^{N} \mathbb{I}_j$
4. $\sigma_j^x \sigma_{j+1}^x = (a_j^\dagger - a_j)(a_{j+1} + a_{j+1}^\dagger)$
5. $\sigma_j^y \sigma_{j+1}^y = -(a_j^\dagger + a_j)(a_{j+1}^\dagger - a_{j+1})$
6. $\sigma_j^x \sigma_{j+1}^y = i(a_j^\dagger - a_j)(a_{j+1}^\dagger + a_{j+1})$
7. $\sigma_j^y \sigma_{j+1}^x = i(a_j^\dagger + a_j)(a_{j+1}^\dagger + a_{j+1})$

Bibliography

Horn, Roger A. and Charles R. Johnson, "Matrix Analysis by Roger A. Horn," /core/books/matrix-analysis/9CF2CB491C9E97948B15FAD835EF9A8B December 1985.

Lieb, Elliott, Theodore Schultz, and Daniel Mattis, "Two Soluble Models of an Antiferromagnetic Chain," *Annals of Physics*, December 1961, *16* (3), 407–466.

Nielsen, Michael A, "The Fermionic Canonical Commutation Relations and the Jordan-Wigner Transform," p. 8.

Zumino, Bruno, "Normal Forms of Complex Matrices," *Journal of Mathematical Physics*, September 1962, *3* (5), 1055–1057.