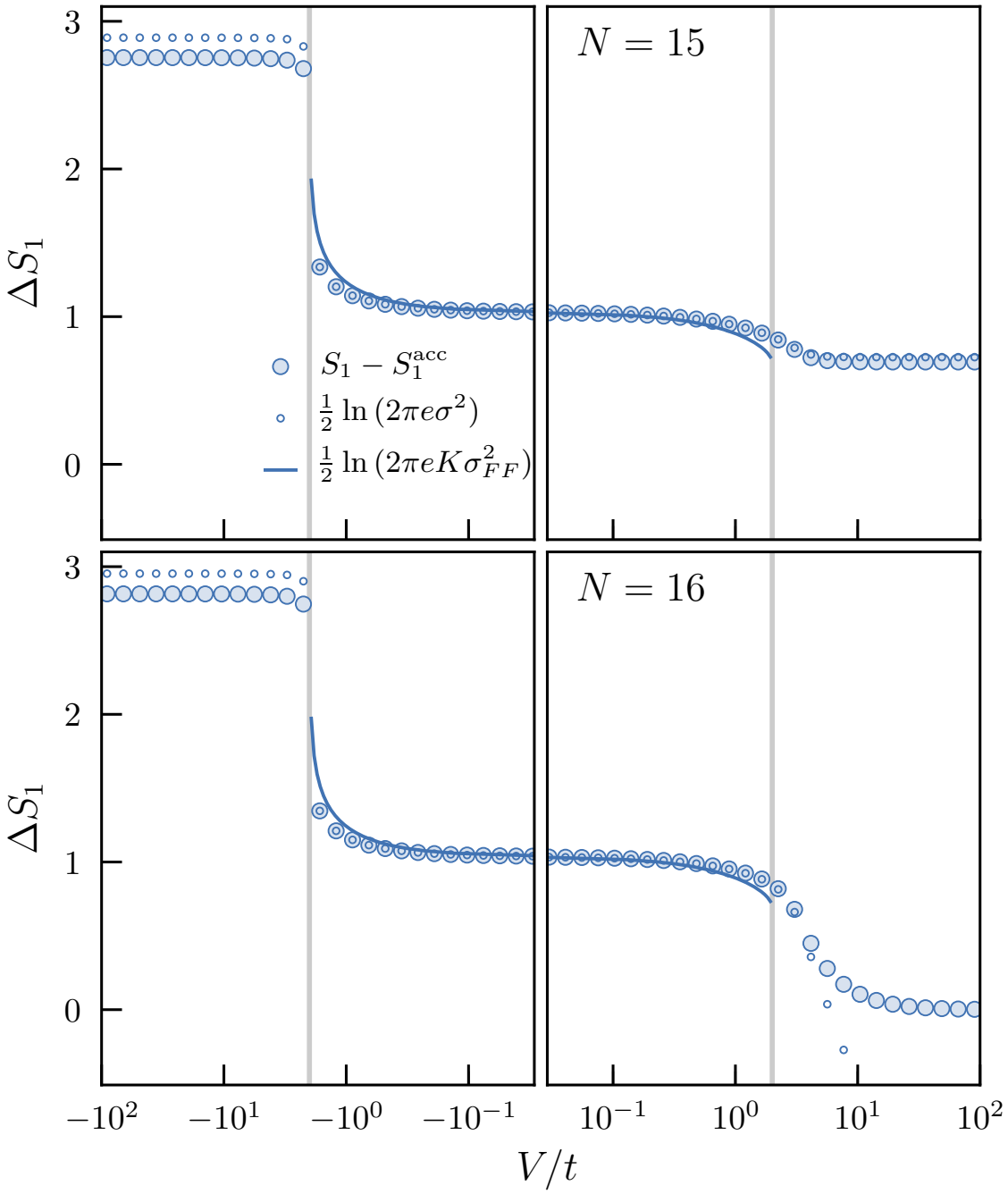

Figure 3.3: Difference between the von Neumann and accessible entanglement entropies $S_1 - S_1^{acc}$ and $\frac{1}{2}\ln 2\pi e\sigma^2$ as functions of interaction strength V/t. The latter expression is the well known differential entropy of a Gaussian distribution. In TLL phase ($-2 < V/t < 2$), the probability distribution is Gaussian, as can be seen from the agreement between the two results. The solid lines use the theoretical variance of particle number in A inside the LL phase, $K\sigma_{FF}^2$, where K is the Luttinger parameter and is a function of V/t and σ_{FF}^2 is the exact variance for free-fermions ($V/t = 0$).

Notice that raising Eq. 3.71 to either $1/\alpha$ or K on both sides should get rid of the α or K dependence of the exponential factor, respectively, within the TLL regime. The square root factor will still pick up the dependence on either of the exponents. In other words, raising by $1/\alpha$ or K should give:

$$P_{n,\alpha}^{1/\alpha} \approx \sqrt{\frac{\pi\alpha}{2K\ln\ell}}^{1/\alpha} e^{\frac{-\pi^2(n-\langle n\rangle^2)}{2K\ln\ell}} \tag{3.72}$$

and

$$P_{n,\alpha}^{K} \approx \sqrt{\frac{\pi\alpha}{2K\ln\ell}}^{K} e^{\frac{-\alpha\pi^2(n-\langle n\rangle^2)}{2\ln\ell}} \tag{3.73}$$

Figure 3.6 shows the distribution $A_\alpha P_{n,\alpha}^{1/\alpha}$ for various interaction strengths V/t and, thus, K. The constant A_α is the inverse of the square root factor in Eq. 3.72. Cancelling the square root factor allows for a direct comparison of the exponential factor for each of the α values used. The middle plots confirm that this exponential factor indeed is independent of α, illustrated by the fact that the distributions become the same for $\alpha = \{1, 2, 5, 10\}$, when inside the TLL regime of $-2 < V/t < 2$.

Figure 3.7 shows the distribution $A_\alpha P_{n,\alpha}^{K}$ with the Rényi index fixed at $\alpha = 2$ and at various interaction strengths V/t and corresponding Luttinger parameters K. In this case, the factor A_α is the inverse of the square root factor in Eq. 3.73. All of the interaction strengths fall within the TLL regime and as such, all the distributions should become the same for the various V/t and, thus, K values. This collapse of the distributions at various K is evident from looking at regions near the middle of the graph. Although it may not be apparent at first glance due to the scale, the tails of the distribution are all essentially zero.

3.6 Conclusion

In this chapter, the operationally accessible Rényi entanglement entropy was introduced in both it's original and generalized form. Analytical values of the entanglement entropy were obtained at various special cases of the tV model and then confirmed via exact diagonalization. A maximum value in accessible entanglement was observed and evidence seems to support that it follows an inverse power law scaling in total particle number with scaling exponent of -0.3066.

The difference in full and accessible entanglement entropies was also computationally determined and it was confirmed that in the TLL phase of the tV model, it is equal to the the Rényi entropy of a Normal Distribution of local particle number. Finally, it was then proposed theoretically and confirmed computationally, that getting rid of its Rényi index and Luttinger parameter dependence, the exponential part of these Normal Distributions depend exclusively on local particle number fluctuations.

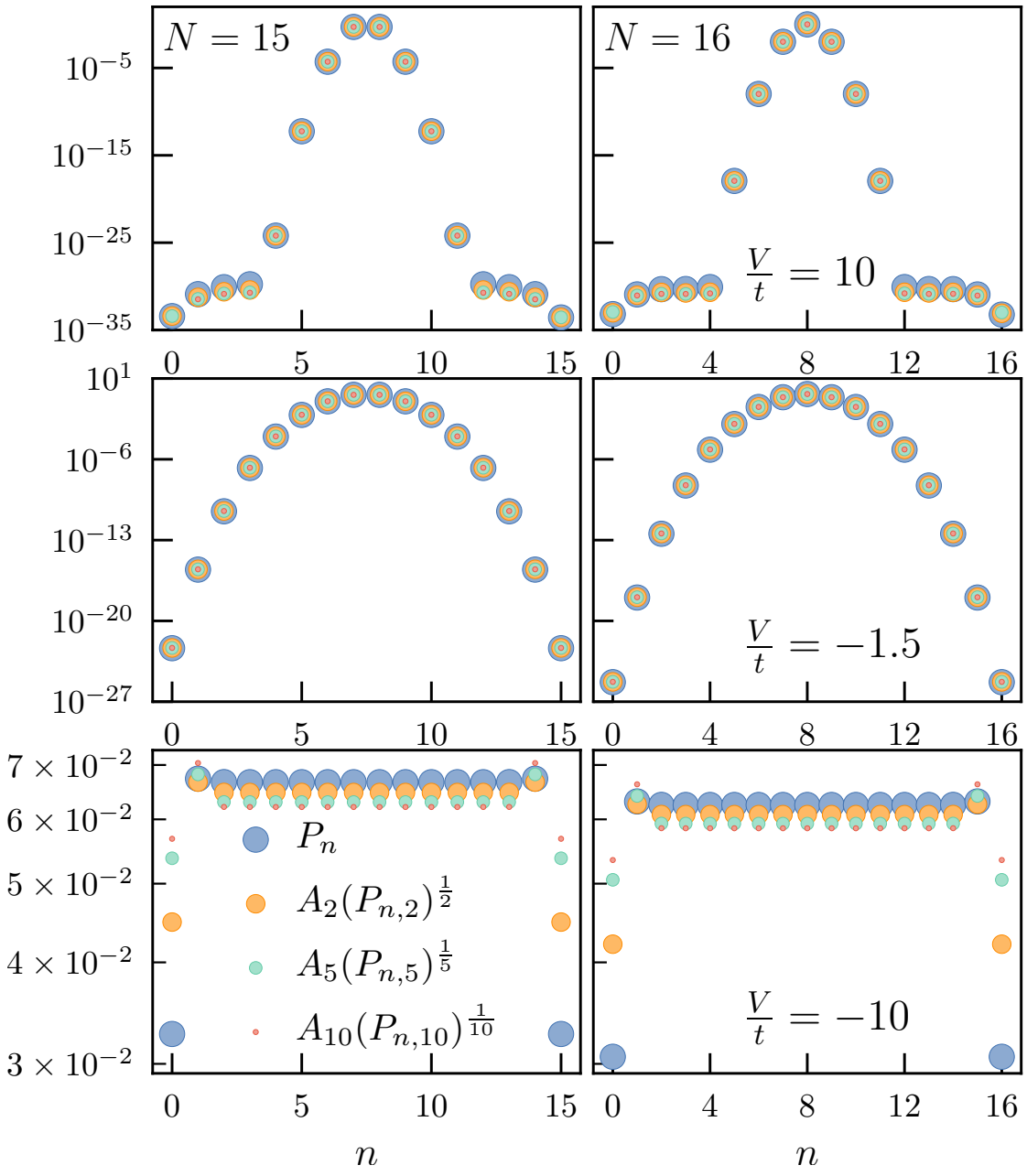

Figure 3.6: Probabilities of measuring a state with n particles in subregion A, as a function of n. The probabilities in the TLL regime are known to be Gaussian, as seen from Eq.6. Here, they have been raised to $1/\alpha$ in order to cancel out the α dependence of the exponential part. For the middle plot, the interaction strength lies in the TLL regime and, consequently, the probabilities collapse to the same values in all the range after the α dependence has been cancelled. The top and bottom plots show results outside of the TLL regime, where the probabilities are not Gaussian.

Appendix A

Appendix

A.1 Lanczos Algorithm

A.1.1 Introduction

The Lanczos Algorithm, takes as input a Hermitian Matrix and iteratively builds a similarity transform that makes it tridiagonal. Due to similarity, the solution of the eigenvalue problem of the tridiagonal matrix is the same as that of the original matrix. Nevertheless, some methods can exploit the tridiagonality to find the eigendecompositio more easily. In condensed matter physics, the input matrix is usually a Hamiltonian. The eigenvalues and eigenvectors of the Hamiltonian represent the energies and the associated quantum states of the system.

In the following section, the Lanczos Algorithm will be derived. Next, some methods for approximating the eigenvalues and eigenvectors will be discussed. Finally, a hopefully simple implementation of the algorithm in Python will be linked and some results will be shown.

A.1.2 Tridiagonalization of the original matrix

Let A be a Hermitian matrix of size nxn. An orthonormal transform matrix Q is needed such that:

$$T = Q^T A Q$$

where T is a tridiagonal and Hermitian matrix similar to A.

The idea is to obtain a recursive relation, starting from the known fact that T is tridiagonal and that the columns of the transform Q are mutually orthonormal. The matrix T has the form:

$$T = \begin{pmatrix} \alpha_1 & \beta_1 & & & & & 0 \\ \beta_1 & \alpha_2 & \beta_2 & & & & \\ & \beta_2 & \alpha_3 & \beta_3 & & & \\ & & \beta_3 & \ddots & \ddots & & \\ & & & \ddots & \alpha_{n-2} & \beta_{n-2} & \\ & & & & \beta_{n-2} & \alpha_{n-1} & \beta_{n-1} \\ 0 & & & & & \beta_{n-1} & \alpha_n \end{pmatrix}$$

$$q_{k+1} = \frac{r_k}{\beta_k} \tag{A.5}$$

where $\beta_k \neq 0$ and, since q_{k+1} is an orthonormal vector, $\beta_k = ||r_k||_2$, such that q_{k+1} is normalized.

Note that the α_k and β_k terms of the three-term recursion relation have been accounted for. As for the β_{k-1}, a "bottom rung" for the recursion has to be set. The tridiagonal matrix T does not have a β_{k-1} term. Thus, for $k = 1$, the $\beta_{k-1}q_{k-1}$ term is set to $\beta_0 q_0 = 0$. Now the columns of Q can be built by iterating from $k = 1$ to $k = n$.

A.1.3 Algorithm

1. Set $r_0 = q_1$,$\beta_0 = 1$ and $q_0 = 0$
2. For k=1,2,3,...,n:
3. $q_{k+1} = \frac{r_k}{\beta_k}$
4. $\alpha_k = q_k^T A q_k$
5. $r_k = (A - \alpha_k I)q_k - \beta_{k-1}q_{k-1}$
6. $\beta_k = ||r_k||_2$
7. Reorthonormalize $\{q_i\}_{i=1}^k$ if necessary
8. Approximate Eigenvalues and Eigenvectors (Can be done after the loop instead)

Line 1: β_0 is set to 1 since it is the norm of r_0 and $r_0 = q_1$, where q_1 is a normalized vector.

Line 2: The for loop runs from $k = 1$ all the way up to $k = n$, where n is the total number of columns. Depending of the eigenvalues desired, this loop can instead be a while loop that ends whenever the eigenvalues have reached a desired tolerance.

Line 7: Due to finite precision errors, the set of supposedly mutually orthonormal vectors $\{q_i\}_{i=1}^k$ will actually lose their orthonormality at later Lanczos steps. When this happens, a reorthonormalization scheme, such as the Grahm-Schmidt Process, has to be employed.

Line 8: Again, depending on the problem and the desired eigenpairs, the approximation can be done for the current version of the tridiagonal matrix at step k, call it T_k. Alternatively, it could be done after the for loop has finished and the full tridiagonal matrix has been T built. There is no strict requirement on which iterative method should be used to find the eigendecomposition (QR Method, Power Iteration, Inverse Power Iteration, etc...).

A.1.4 Code

An implementation of the Lanczos Algorithm in Python can be found in: https://github.com/ecasian
The code generates a random, sparse, hermitian matrix of specified size, finds a tridiagonal representation via Lanczos and calculates the full eigendecomposition via QR Algorithm or finds the smallest eigenvalue via Inverse Power Iteration. A blackbox

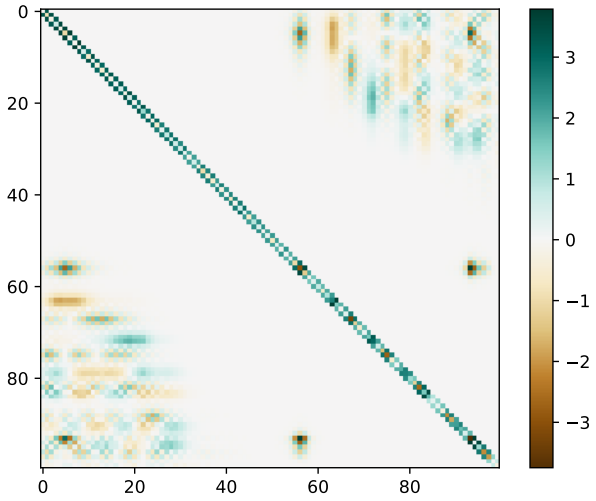

Figure A.2: INSERT CAPTION.

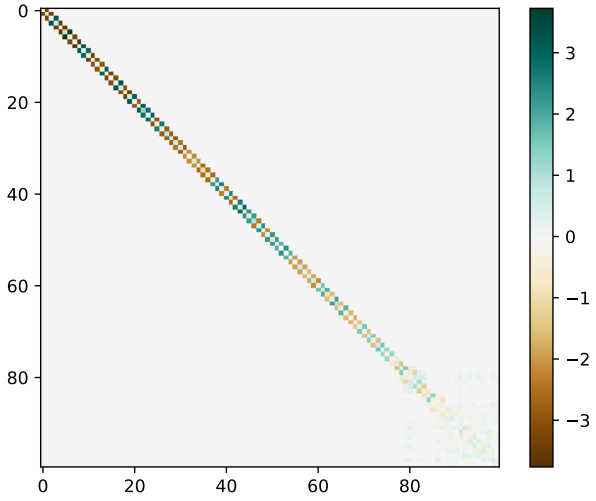

Figure A.3: INSERT CAPTION.

Bibliography

[1] H. Barghathi, C. M. Herdman, and A. Del Maestro. "Rényi generalization of the accessible entanglement entropy." Phys. Rev. Lett. **121**, 150501 (2018). https://link.aps.org/doi/10.1103/PhysRevLett.121.150501

[2] P. Zanardi. "Virtual Quantum Subsystems." Phys. Rev. Lett. **87**, 077901 (2001).

[3] Y. Shi. "Quantum entanglement of identical particles." Phys. Rev. A **67**, 024301 (2003).

[4] Y. Shi. "Quantum entanglement in second-quantized condensed matter systems." J. Phys. A: Math. Gen **37**, 6807 (2004).

[5] G. Ghirardi and L. Marinatto. "General criterion for the entanglement of two indistinguishable particles." Phys. Rev. A **70**, 012109 (2004).

[6] H. Barnum, E. Knill, G. Ortiz, R. Somma, and L. Viola. "A Subsystem-Independent Generalization of Entanglement." Phys. Rev. Lett. **92**, 107902 (2004).

[7] J. Dunningham, A. Rau, and K. Burnett. "From Pedigree Cats to Fluffy-Bunnies." Science **307**, 872 (2005).

[8] H. M. Wiseman, S. D. Bartlett, and J. A. Vaccaro. "Ferreting out the fluffy bunnies: entanglement constrained by generalized superselection rules." In "Proceedings of the XVI International Conference on Laser Spectroscopy," pages 307–314 (World Scientific, Queensland, Australia, 2011). https://www.worldscientific.com/doi/abs/10.1142/9789812703002_0047

[9] H. M. Wiseman and J. A. Vaccaro. "Entanglement of Indistinguishable Particles Shared between Two Parties." Phys. Rev. Lett. **91**, 097902 (2003).

[10] F. Benatti, R. Floreanini, and U. Marzolino. "Entanglement robustness and geometry in systems of identical particles." Phys. Rev. A **85**, 042329 (2012).

[11] A. P. Balachandran, T. R. Govindarajan, A. R. de Queiroz, and A. F. Reyes-Lega. "Entanglement and Particle Identity: A Unifying Approach." Phys. Rev. Lett. **110**, 080503 (2013).

[12] B. J. Dalton, J. Goold, B. M. Garraway, and M. D. Reid. "Quantum entanglement for systems of identical bosons: I. general features." Phys. Scrip. **92**, 023004 (2017).