Strategies for optimizing double-bracket quantum algorithms

Li Xiaoyue[1], **Matteo Robbiati**[2,3], **Andrea Pasquale**[3,4,5], **Edoardo Pedicillo**[3,4], **Andrew Wright**[6], **Stefano Carrazza**[2,3,4,5] **and Marek Gluza**[1]

[1]School of Physical and Mathematical Sciences, Nanyang Technological University, 21 Nanyang Link, 637371 Singapore, Republic of Singapore
[2]European Organization for Nuclear Research (CERN), Geneva 1211, Switzerland
[3]TIF Lab, Dipartimento di Fisica, Università degli Studi diMilano
[4]Quantum Research Centre, Technology Innovation Institute, Abu Dhabi, UAE
[5]INFN Sezione di Milano, Milan, Italy
[6]Institute of Physics, Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland.

E-mail: marekludwik.gluza@ntu.edu.sg

Abstract. Recently double-bracket quantum algorithms have been proposed as a way to compile circuits for approximating eigenstates. Physically, they consist of appropriately composing evolutions under an input Hamiltonian together with diagonal evolutions. Here, we present strategies to optimize the choice of the double-bracket evolutions to enhance the diagonalization efficiency. This can be done by finding optimal generators and durations of the evolutions. We present numerical results regarding the preparation of double-bracket iterations, both in ideal cases where the algorithm's setup provides analytical convergence guarantees and in more heuristic cases, where we use an adaptive and variational approach to optimize the generators of the evolutions. As an example, we discuss the efficacy of these optimization strategies when considering a spin-chain Hamiltonian as the target. To propose algorithms that can be executed starting today, fully aware of the limitations of the quantum technologies at our disposal, we finally present a selection of diagonal evolution parametrizations that can be directly compiled into CNOTs and single-qubit rotation gates. We discuss the advantages and limitations of this compilation and propose a way to take advantage of this approach when used in synergy with other existing methods.

1 Introduction

Preparing the eigenstates of a target physical system is crucial in fields like physics, chemistry, and combinatorial optimization. Various strategies exist within both classical [1, 2, 3, 4] and quantum computing domains. In the quantum realm, high-precision algorithms like quantum phase estimation [5, 6] (QPE) await technological advancements for practical use, while near-term routines such as variational quantum eigensolvers [7] (VQEs) are more feasible. This work focuses on a family of algorithms suitable for near-term quantum devices: double-bracket quantum algorithms (DBQAs).

These algorithms have been proposed to prepare eigenstates through a recursive double-bracket iteration (DBI) strategy [8] and they can be compiled into quantum circuits. In fact, originally, Ref. [8] reduced DBQA compilation from the abstract DBI equations to a sequence of queries to Hamiltonian simulation which can be compiled by known methods [9, 10, 11, 12], but left open quantitative questions

about the precise cost of compiling DBIs into circuits. In particular, the depth of DBQA circuits grows exponentially with the number of recursion steps.

Despite this, recent studies [13] show that when initialized properly, DBQAs can substantially improve the eigenstate fidelity reached by quantum algorithms such as VQEs in just one or two steps.

This trade-off between computational cost and gain is crucial: DBQAs must be used synergistically with other approximation techniques and double-bracket iterations must be optimized to provide maximum benefit in a few recursion steps.

Moreover, DBQAs can be optimized to maximize diagonalization gain in every step. Specifically, we adopt optimization methods to accelerate the preparation of a target eigenstate. In this proceedings, we aim to study in depth such optimization techniques. We present numerical results implemented in Qibo [14, 15, 16], as used in Ref [13].

2 Double-bracket quantum algorithms for diagonalization

The building blocks of the double-bracket quantum algorithms are double-bracket rotations (DBRs), which iteratively update a target Hamiltonian $\hat{H}_0$ following

$$\hat{H}_{k+1} = e^{s\hat{W}_k}\hat{H}_k e^{-s\hat{W}_k} \equiv \hat{R}_k^\dagger \hat{H}_k \hat{R}_k \ . \tag{1}$$

Here $\hat{W}_k = [\hat{D}_k, \hat{H}_k]$ is a commutator involving $\hat{D}_k$ which is a diagonal hermitian operator and $\hat{R}_k = e^{-s\hat{W}_k}$ is the unitary operator implementing the DBR. In order to see why DBRs have diagonalizing properties, we motivate the above ansatz $\hat{W}_k = [\hat{D}_k, \hat{H}_k]$, by letting $\hat{W}_k$ be any anti-hermitian operator. Let $\Delta(A)$ be the diagonal restriction of matrix A and $\sigma(A)$ the off-diagonal restriction, s.t. $A = \Delta(A) + \sigma(A)$. Performing a Taylor expansion of of $\sigma(\hat{H}_{k+1})$ in Eq. (1) around $s = 0$ and using that $\frac{\partial}{\partial s}||\sigma(e^{s\hat{W}_k}\hat{H}_k e^{-s\hat{W}_k})||^2_{\text{HS},s=0} = -2\langle\hat{W}_k, [\Delta(\hat{H}_k), \hat{H}_k]\rangle_{\text{HS}}$ we get

$$\sigma(\hat{H}_{k+1}) = \sigma(\hat{H}_k) - 2s\langle\hat{W}_k, [\Delta(\hat{H}_k), \hat{H}_k]\rangle_{\text{HS}} + \ldots \ , \tag{2}$$

which implies that the rotation reduces the off-diagonal norm of the Hamiltonian $\hat{H}_{k+1}$ compared to $\hat{H}_k$ if

$$\langle\hat{W}_k, [\Delta(\hat{H}_k), \hat{H}_k]\rangle_{\text{HS}} > 0 \ . \tag{3}$$

The choice, $\hat{W}_k = [\Delta(\hat{H}_k), \hat{H}_k]$, emerges naturally and guarantees an initial drop in the off-diagonal norm and is known as the canonical bracket. Additionally, if we choose $\hat{W}_k = [\hat{D}_k, \hat{H}_k]$, where $\hat{D}_k$ is a diagonal hermitian operator which may be easier to implement than $\Delta(\hat{H}_k)$, the overlap is likely to induce a σ-decrease in Eq. (2). Therefore, this ansatz is largely flexible in devising diagonalization DBIs outside of using the canonical bracket.

2.1 Types of DBIs

In this work, we will consider three types of DBIs: the first two are derived from double-bracket flow formulations (the continuous version of the one analyzed here), while the third is a variational approach to the problem of optimizing the generators $\hat{D}_k$ of the double-bracket evolutions.

The first DBI formulation is inspired by Brokett's work [17], which introduced a double-bracket flow involving a fixed diagonal operator $\hat{N}$. Afterwards, Moore, Mahony, and Helmke [18] have considered double-bracket iterations where each double-bracket rotation uses $\hat{N}$ as diagonal operator: $\hat{D}_k = \hat{N}$. They concluded that if $\hat{N}$ has a non-degenerate spectrum then BHMM converges to a diagonal fixed-point $\hat{H}_\infty$ and the only stable fixed-point is the one where $\hat{H}_\infty$ has the same sorting of eigenvalues as $\hat{N}$. Hence, we call this strategy the Brokett-Helmke-Moore-Mahony (BHMM) DBI.

The second double-bracket flow is based on works by Głazek, Wilson, and Wegner [19, 20] (GWW). The corresponding GWW DBI arises recursively by setting $\hat{D}_k = \Delta(\hat{H}_k)$, which makes $\hat{W}_k$ the canonical bracket. This choice results in an unconditional σ-decrease where Eq. (2) becomes

$$\frac{\partial}{\partial s}||\sigma(e^{s\hat{W}_k}\hat{H}_k e^{-s\hat{W}_k})||^2_{\text{HS},s=0} = -2||\hat{W}_k||^2_{\text{HS}} \ . \tag{4}$$

The third DBI paves the way for a variational approach. Given each $\hat{H}_k$, one can consider introducing a parametric form of $\hat{D}_k$ and optimizing its parameters to maximize the decrease of a cost function which quantifies the effectiveness of diagonalization. One possible choice for the cost function could be the off-diagonal norm of the target Hamiltonian. In Sec 2.2 we introduce some other possible cost functions. We will call adaptive such strategies where we optimize $\hat{D}_k$ for each $\hat{H}_k$. They have the advantage of possibly being more efficient but, due to implicit numerical optimization, their convergence analysis is precluded as opposed to GWW and BHMM DBIs.

2.2 Cost functions for diagonalization

When defining an optimization procedure to find a good candidate as a double-bracket rotation generator, it is crucial to select a proper cost function whose value has to be optimized. We next review possible cost functions and comment on their role in a prospective quantum computation.

A first possible choice (f_1 in Tab. 1) is the off-diagonal norm, i.e. the Hilbert-Schmidt norm of the restriction to the off-diagonal of an input, which is motivated by the monotonicity relation (2).

A second cost function (f_2 in Tab. 1) is inspired by the work of Brockett [17], who originally considered least-square functional optimization $\|\hat{A} - \hat{N}\|_{\mathrm{HS}}$ where $\hat{A}$ (in our case $\hat{A} = \hat{H}_0$ but the setting is more general) was being varied while $\hat{N}$ was a fixed target matrix which can be chosen to be diagonal without loss of generality. If the DBI considered is BHMM then we can set $\hat{D}_k = \hat{N}$ and indeed $f_2(\hat{H}_k) = \|\hat{H}_k - \hat{N}\|^2 - \|\hat{H}_0\|^2$, agreeing with Brockett's cost function because $\|\hat{H}_k\|$ is constant due to unitary invariance. As reported in f_2 formula from Tab. 1, the norm $\|\hat{D}_k\|_{\mathrm{HS}}^2$ is needed to fix normalization. For the GWW DBI we have $\hat{D}_k = \Delta(\hat{H}_k)$, the f_2 cost function reduces to f_1 because $\hat{H}_k = \Delta(\hat{H}_k) + \sigma(\hat{H}_k)$.

A third choice is to consider the energy expectation for a given state (f_3 in Tab. 1). Minimizing this value aims to find the ground energy and state of the target Hamiltonian, making it a good cost function to utilize in conjunction with the VQE [13].

Finally, we can use the energy fluctuation of a reference state $|\psi\rangle$ (f_4 in Tab. 1). It allows to detect whether $|\psi\rangle$ is converging to an eigenstate of $\hat{H}$. A few steps of DBI can usually produce a good eigenstate approximation for some states $|\psi\rangle$, making the energy fluctuation an appropriate measure of eigenstate approximation which is necessary for full diagonalization.

Function	Off-diagonal norm (f_1)	Least-squares (f_2)	Energy (f_3)	Energy fluctuation (f_4)
Formula	$\|\sigma(\hat{H}_k)\|_{\mathrm{HS}}$	$\frac{1}{2}\|\hat{D}_k\|_{\mathrm{HS}}^2 - \mathrm{Tr}(\hat{H}_k\hat{D}_k)$	$\langle\psi|\hat{H}|\psi\rangle$	$\sqrt{(\langle\psi|\hat{H}_k^2|\psi\rangle - \langle\psi|\hat{H}_k|\psi\rangle^2)}$
Utility	Diagonalization	Diagonalization	Ground state preparation	Eigenstate preparation

Table 1: Cost functions DBQAs optimization with possible applications.

As shown in Fig. 1, when we optimize DBRs based on the energy fluctuation of a state $|\psi\rangle$, while the energy fluctuation is indeed suppressed and the eigenstate is well approximated, the off-diagonal norm is not necessarily reduced.

2.3 Optimization tasks

Once a specific DBI is chosen and we decide to formalize the diagonalization by minimizing a specific cost function f_*, the optimization problem can be divided, in practice, into three tasks: the *scheduling*, the *generator selection* and the *compilation*.

The first task involves finding the DBR duration s_k which solves

$$\mathrm{argmin}_{s_k < s_{\max}}\left[f_*(e^{s[\hat{D}_k, \hat{H}_k]}\hat{H}_k e^{-s[\hat{D}_k, \hat{H}_k]})\right] \tag{5}$$

given a diagonalization operator $\hat{D}_k$. In Eq. (5) we limit the optimization to some $s_{\max}$ because s_k increases the gate cost of DBI, see Sec. 3.

The *generator selection* task seeks to optimize the diagonal operator $\hat{D}_k$ in the rotation generator $\hat{W}_k$. This entails parametrization of $\hat{D}_k$ appropriate for quantum computing and assumes oracle access

to a solver of the scheduling task. We quantify the output of the generator selection task by evaluating the diagonalization gain $f_*(e^{s[\hat{D}_k, \hat{H}_k]} \hat{H}_k e^{-s[\hat{D}_k, \hat{H}_k]})$ as detailed in Sec. 4.

Finally, DBR *compilation* means the DBR unitaries $\hat{R}_k$ must be approximated by a circuit made of primitive gates and is discussed in Sec. 5.

2.4 Models considered

As practical study cases, we consider two spin-chain Hamiltonians as targets. Firstly, we tackle the transverse field Ising model (TFIM) with periodic boundary conditions (PBC)

$$\hat{H}_{\text{TFIM}} = \sum_{j=1}^{L} \hat{X}_j \hat{X}_{j+1} + h\hat{Z}_j \ . \tag{6}$$

Sometimes TFIM is defined as $\sum_{j=1}^{L} \hat{Z}_j \hat{Z}_{j+1} + h\hat{X}_j$ but having diagonalization in mind we take the 2-qubit interaction to be off-diagonal. Secondly, we consider the Heisenberg XXZ model again with PBC

$$\hat{H}_{\text{XXZ}} = \sum_{j=1}^{L} (\hat{X}_j \hat{X}_{j+1} + \hat{Y}_j \hat{Y}_{j+1} + \delta \hat{Z}_j \hat{Z}_{j+1}) \ . \tag{7}$$

we will use TFIM and XXZ to demonstrate our observations about optimizing DBI in Qibo.

3 Scheduling

The simplest approach to find the optimal DBR duration s_k at each step consists of using a standard hyper-optimization algorithm. Grid search, random search [21], Tree of Parzen estimators [22] techniques are all viable for this task and they are available in Qibo. The scheduling cost functions from Sec. 2.2 have analytical properties which we can take into account by using Taylor expansion. For this we define

$$\Gamma_n = [\hat{W}, \Gamma_{n-1}] \tag{8}$$

with $\Gamma_0 = \hat{H}$ where $\hat{H}$ is any operator, e.g. $\hat{H}_0$ or $\hat{H}_k$. The Baker–Campbell–Hausdorff (BCH) formula gives $e^{s\hat{W}} \hat{H} e^{-s\hat{W}} = \Gamma_0 + s\Gamma_1 + \frac{s^2}{2}\Gamma_2 + O(s^3)$. Taking the off-diagonal norm as an example, Eq. (2) taken up to degree 2 yields a parabola of variable s:

$$\frac{\partial}{\partial s} \|\sigma(e^{s\hat{W}} \hat{H} e^{-s\hat{W}})\|_{\text{HS}}^2 = s^2 \operatorname{Tr}\left(3\sigma(\Gamma_1)\sigma(\Gamma_2) + \sigma(\Gamma_0)\sigma(\Gamma_3)\right) + 2s \operatorname{Tr}\left(\sigma(\Gamma_1)\sigma(\Gamma_1) + \sigma(\Gamma_0)\sigma(\Gamma_2)\right)$$
$$+ 2\operatorname{Tr}\left(\sigma(\Gamma_0)\sigma(\Gamma_1)\right) + O(s^3) \ . \tag{9}$$

Subsequently, we can integrate Eq. (9) and obtain the expansion for the off-diagonal norm $\|\sigma(e^{s\hat{W}} \hat{H} e^{-s\hat{W}})\|$ as a polynomial. Fig. 1 shows the results of scheduling based on finding minima of Taylor polynomials and compares the Taylor approximation to the exact σ-decrease curve. More specifically, we find the roots of the Taylor polynomial of Eq. (9) and output the minimizer s at the first extremum which by the monotonicity relation will be a minimum, provided that DBR is directed to diagonalization. This avoids evaluating multiple exponentials for cost function queries and the obtained approximation can be improved by taking higher-order Taylor polynomials. Moreover, while the first minimum may not be the global minimum, we have seen cases where taking the first local minimum outperforms taking the global minimum of the DBR after several iterations. This indicates that greedy strategies at every step are not necessarily optimal.

4 Generator selection

In this section, we introduce some possible strategies to parametrize diagonal operators. We begin with ansatze that are motivated analytically and then discuss those suitable for quantum compiling. Finally, we present numerical results considering a 5-qubit Heisenberg model as the target and the off-diagonal norm as the cost function.
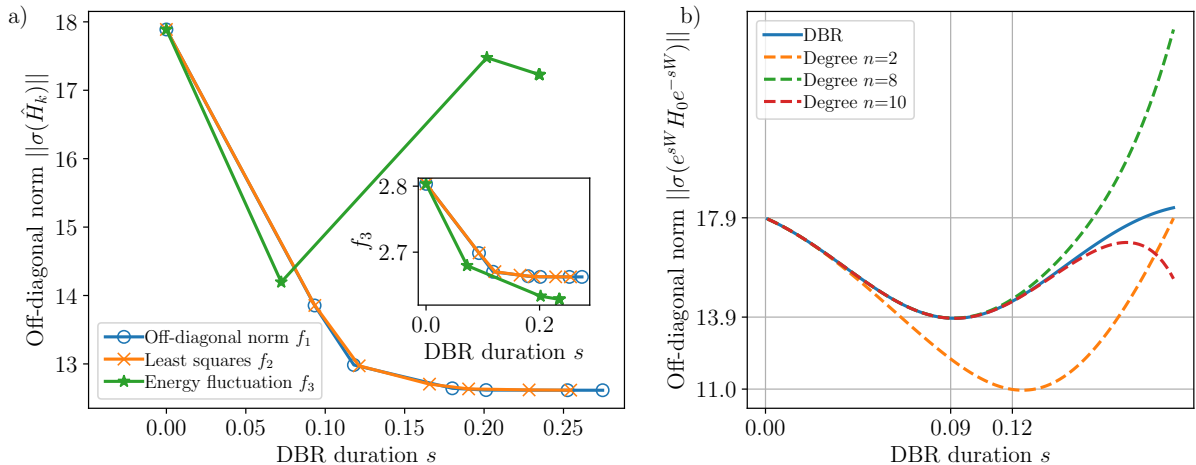
Figure 1: TFIM with $L = 5$, transverse field $h = 3$ and $\hat{D}_0 = \Delta(\hat{H}_0)$. a) We show f_1 for DBI with steps optimized using f_1, f_2 and f_3. The inset shows that optimization with f_3 is sensitive to the reference state. b) We show the validity of the Taylor expansion for various orders n. For all n the DBR is well approximated for short DBR durations $s < 0.04$. In order to match the first local minimum of the σ-decrease curve, we find that a relatively high polynomial degree is needed.

4.1 Min-max inspired parametrizations

Moore, Mahony, and Helmke proved that if D^* is non-degenerate then the BHMM DBI converges [18]. A simple example of such an operator is the *min-max parametrization* where the values on the diagonal are arranged from the smallest to the largest element of the diagonal entries of $\hat{H}$ and are equidistant. In other words, we set $\delta = \frac{d_n - d_1}{n}$ so that

$$\hat{D}^*_{\min-\max} = \sum_{i=1}^{2^L} (\min(\Delta(\hat{H})) + i\delta) \, |i\rangle \langle i| \ . \tag{10}$$

This ansatz can be difficult to implement on a quantum computer using local gates but it is an instructive toy model regarding convergence properties of DBIs, e.g. it is sufficient if $\hat{D}^*$ is non-degenerate.

Since the BHMM DBIs have sorting properties (the fixed point is sorted according to $\hat{D}^*$), we also investigated the inverted *max-min parametrization* with entries like min-max but in the reversed order, i.e. we use max instead of min in Eq. (10). Remarkably, depending on the input basis of $\hat{H}$ one of these two orderings might be closer to the initialization.

To explore the impact of the ordering, we apply a random permutation to the spectrum of the min-max parametrization. We refer to this strategy as *shuffled min-max parametrization*. There are many possible orderings of a min-max parametrization and a random sample of an ordering can reveal insights that structured min-max and max-min parametrization does not exhibit.

To further investigate how degeneracy affects diagonalization efficiency, we consider a diagonal operator whose diagonal entries are sampled with repetition from the spectrum $\mathrm{spec}(\hat{D}^*_{\min-\max})$. We sort them in an ascending order to facilitate comparisons with the min-max parametrization. We refer to this parametrization as *sampled min-max parametrization*.

If $\hat{H}$ has a non-degenerate spectrum then instead of the min-max parametrization we can choose $\hat{D}^*_{\mathrm{eigen}} = \mathrm{diag}(\lambda_1, \lambda_2, ..., \lambda_n)$, where $\lambda_1, \lambda_2, ..., \lambda_n$ are eigenvalues of $\hat{H}$ and we take them in ascending order $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$. Checking this operator should reveal whether the linear spectrum in min-max should instead have a non-linearity.

4.2 Adaptive parametrizations

We now explore some *adaptive strategies*, in the sense that the parameters of the diagonal operators are from now on considered to be optimized at each recursion step.

The first strategy is the *dephasing channel parametrization* and is based on the fact the GWW DBI involves the dephasing channel which restricts matrices to their diagonal. Thus, for each k, we set $\hat{D}_k = \Delta(\hat{H}_k)$. This strategy can also be used to initialize the BHMM DBI and inform it about the initial diagonal structure of the input matrix. For example, $\Delta(\hat{H}_{\text{TFIM}}) = \sum_{j=1}^{L} \hat{Z}_j$, which coincides with a constant magnetic field. The XXZ model has $\Delta(\hat{H}_{\text{XXZ}}) = \delta \sum_{j=1}^{L} \hat{Z}_j \hat{Z}_{j+1}$, which means that the first step of the GWW BHMM DBI agrees with a DBI parametrized by the classical nearest-neighbor (NN) Ising model.

Motivated by the above, we introduce the *magnetic field parametrization*, which considers a local magnetic field in the z-direction

$$\hat{D}_k(B^{(k)}) = \sum_{j=1}^{L} \alpha_j^{(k)} \hat{Z}_j \, , \tag{11}$$

where $\alpha_j^{(k)}$ are the values of an artificial magnetic field of qubit j in step k. While $\alpha_j^{(k)}$ can be arbitrary, e.g. arising from gradient descent optimization, for BHMM DBI we show the cases where $\alpha_j = 1$ (constant), $\alpha_j = j$ (linear), and $\alpha_j = j^2$ (quadratic). For adaptive DBI we optimize these using gradient descent in Qibo. These choices are motivated by the above min-max parametrization in the sense that by changing the α_j values we may try to lift degeneracies.

Next, we consider an ansatz which can have an entangling action on the qubits and such extension of the non-interacting magnetic field can be useful for quantum many-body Hamiltonians. More specifically we consider the classical Ising model with nearest-neighbor (NN) interactions

$$\hat{D}_k(B^{(k)}, J^{(k)}) = \sum_{j=1}^{L} (\alpha_j^{(k)} \hat{Z}_j + \beta_j^{(k)} \hat{Z}_{j+1} \hat{Z}_j), \tag{12}$$

which gives also the name to the ansatz. For BHMM DBI we will choose uniform $\beta_i = 1$ and for adaptive DBI again use gradient descent.

Eigenstates of quantum many-body systems in the middle of the spectrum are very strongly entangled so we also consider the case where there are local, i.e. 2-qubit, interactions as above but the range of interaction is arbitrary

$$\hat{D}_k(B^{(k)}, J^{(k)}) = \sum_{j,j'=1}^{L} \left(\alpha_j^{(k)} \hat{Z}_j + \beta_{j,j'}^{(k)} \hat{Z}_j \hat{Z}_{j'} \right) \, . \tag{13}$$

We call this strategy *all-to-all classical Ising model* and its all-to-all interactions are a hallmark advantage of trapped-ion platforms. To be specific we will set $\beta_{i,j} = 1$ to be the uniform all-to-all parametrization. Note that selecting $\beta_{i,j} = \delta_{i,j-1}$ reduces this model to the NN model.

Finally, we can explore what changes if we take more qubits to participate in an interaction. Here we shall choose a binary string $\mu_k \in \{0,1\}^L$ and consider

$$\hat{D}_k = (Z_1)^{\mu_1} (Z_2)^{\mu_2} \ldots (Z_L)^{\mu_L}. \tag{14}$$

This allows any subset of qubits to interact. The optimization in this case varies over the Hamming cube aided by solving the scheduling task. We refer to this final strategy as *Pauli-Z product parametrization*.

All above models can be viewed as parametrizations for optimization over the parameters, e.g. $\alpha \in R^L$ and $\beta \in R^{L^2}$ for the all-to-all parametrization. The extreme case is to allow all matrix elements to vary

$$\hat{D}_k = \sum_{i=1}^{2^L} d_i^{(k)} |i\rangle\langle i| \tag{15}$$

The disadvantage of this ansatz lies in its scalability as the number of parameters grows exponentially with the system size $\mathcal{O}(2^L)$.
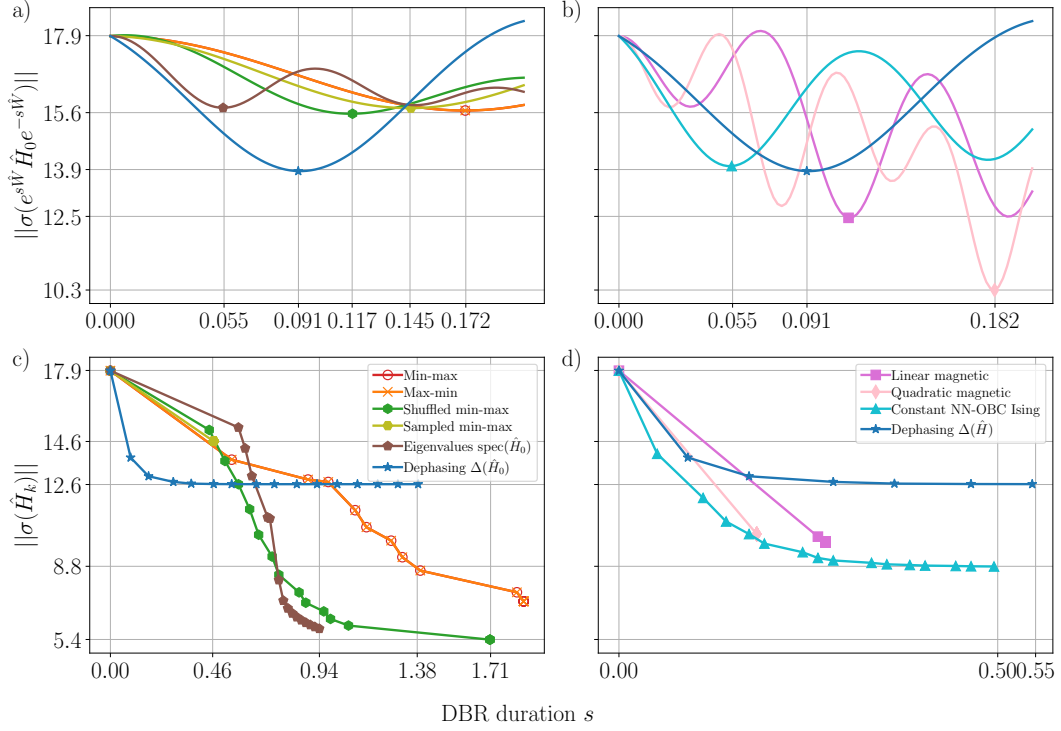
Figure 2: DBRs (top) and DBIs (bottom) for the XXZ model with $L = 5$ qubits and $\Delta = 0.5$. In panel a) we show DBRs for the analytically motivated ansatze. The GWW parametrization $\Delta(\hat{H}_0)$ yields the largest σ-decrease. Bullets indicate the first local minima extracted numerically. In panel b) we show DBRs for parametrizations which can be easily compiled using CNOT and single-qubit rotation gates. Here the global minimum is not necessarily the first. The GWW curve is repeated because it coincides with the NN Ising strategy. In panels c) and d) we show the respective BHMM iteration. We stop, for each adopted strategy, the DBI steps when they would have provided only marginal gain at the expense of long evolutions. Due to the symmetries of XXZ the GWW and magnetic field strategies yield block-diagonalization as opposed to full diagonalization. In panel d) for short s the local $\hat{D}$ parametrization strategies perform comparably well with those in panel c). These results set the benchmark when moving from analytically prescribed BHMM DBIs to adaptive DBIs.

4.3 Benefits of normalized adaptive parametrizations

Given our structuring of optimization strategies, varying the norm of the diagonal operator $\hat{D}_k$ will not facilitate cost function gains. To see this, let us consider a rescaling factor r and we find that if $\hat{R} = e^{-s_k[\hat{D}_k, \hat{H}_k]}$ was the optimal DBR for $\hat{D}_k$ then $s'_k = s_k/r$ is the optimal duration for $\hat{D}'_k = r\hat{D}_k$. This is because $\hat{R}'_k = e^{-s'_k \ [\hat{D}'_k, \hat{H}_k]} = \hat{R}_k$ is optimal whenever $\hat{R}_k$ is optimal.

We can use this fact together with the fact that adding a multiple of the identity zI to the diagonal operator leaves the corresponding DBR invariant, namely because $[\hat{D}_k + zI, \hat{H}_k] = [\hat{D}_k, \hat{H}_k]$. If we optimize the entries of $\hat{D}$ over any interval $[a, b]$, then the optimal diagonal operator can be found. Indeed, if the true minimizer lies in the interval $[a', b']$ then the rescaling and shifting invariance of DBRs allow to find r and z such that $[ra + z, rb + z] \subset [a', b']$.

Finally, we point out that one may use projected gradient approaches to optimize over unit-norm operators. This can be implemented by taking a gradient descent step and dividing by the norm after an update. Focusing on unit-norm parametrizations of diagonal operators is particularly useful for the least-squares loss function f_2. Here ordinary gradient descent will tend to focus on reducing the norm of $\hat{D}$ instead of the $\text{Tr}(\hat{H}_k \hat{D}_k)$ which is more important for diagonalization. In fact, when using projected gradient descent one could choose $f_2(\hat{H}_k) = \text{Tr}(\hat{H}_k \hat{D}_k)$.

4.4 Numerical results for the BHMM DBI

In Fig. 2 we show the cost function landscape for the first DBI step which showcases how different parametrizations yield different σ-decrease. We then show the performance of the BHMM DBI with the numerically obtained scheduling. In Fig. 2 a) and c) we consider $\hat{D}$'s which are analytically motivated. We find, that even though they possess analytical convergence guarantees it is slow for XXZ. Additionally, we find that for the first few steps, degeneracies do not inhibit σ-decrease. The DBIs in panel c) cannot be efficiently implemented on a quantum computer e.g. eigenvalues have to be computed; due to their large number, their compilation could be difficult even if they were known. In Fig. 2 b) and d) we consider $\hat{D}$'s relevant for near-term quantum computers. For the first few steps, these parametrizations perform similarly to those Fig. 2 a) and c), again despite having spectral degeneracies.

For all parametrizations, we find that BHMM DBIs lose efficiency and would need many steps with very short duration. This motivates us to optimize $\hat{D}_k$ in every step and consider adaptive DBIs.

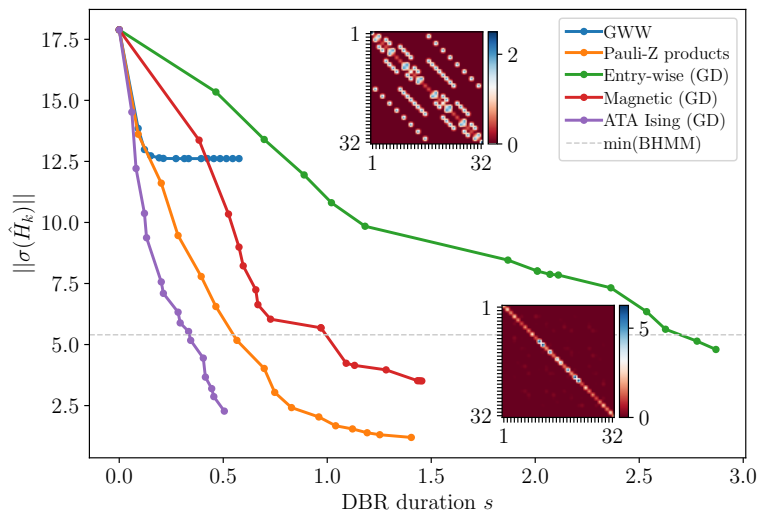### 4.5 Numerical results for the adaptive DBIs

Figure 3: DBI applied to XXZ model with $L = 5$ qubits, $\delta = 0.5$ using variational strategies. Variational strategies, except for the GWW strategy (blue) surpass the best diagonalization achieved with BHMM (grey dotted line). See Fig.2. Pauli-Z strategy achieved the most diagonalization but is difficult to implement physically. We also see that despite being more computationally exhaustive, gradient descent with computational ansatz (green) is not necessarily better than gradient descent with magnetic field (red) and Ising model ansatz (purple).

Fig. 3 shows results for adaptive DBIs again for the XXZ model on $L = 5$ sites. GWW DBI is both analytically motivated and adaptive but saturates due to XXZ symmetries. Instead, we use standard gradient descent with an optimized learning rate for the magnetic field parametrization B and NN Ising parametrization B and J with open boundary conditions. Gradient descent targets local minima that depend on initialization so choosing suitable parametrizations as above is important.

We find that within the same number of DBI steps the final cost function is significantly lower than for BHMM. Thus if rounds of optimization are possible, which is true for future quantum hardware, adaptive DBIs should be considered.

5 Group commutator

A DBI is a sequence of DBRs. However, DBR rotations cannot be directly implemented on a quantum computer. Instead, quantum compiling in primitive gates should be used. We compile approximations to DBR unitaries by means of group commutator formulas. These consist of unitaries which then are simple exponentials of Hamiltonians and so Hamiltonian simulations can be used which combine primitive gates into evolution operators $e^{-it\hat{A}}$, where $\hat{A}$ is a local Hamiltonian.
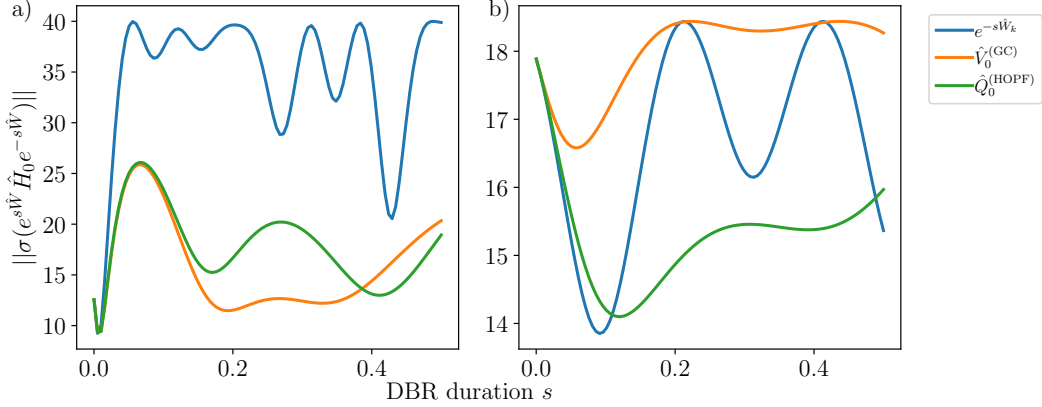
Figure 4: Group commutator approximation to DBR. In panel a) we apply DBR to the TFIM model with $L = 5$ qubits, $h = 3$ and see that both GC and HOPF are good approximations to the desired evolution up to $s < 0.02$ and found the global minimum. Panel b) shows that on the XXZ model with $L = 5$ qubits, $\delta = 0.5$, HOPF performs considerably better than GC, which is far from the true minimum.

Specifically the DBR unitary $\hat{R}_0 = e^{-s_0[\hat{D}_0, \hat{J}_0]}$ can be approximated by the group commutator unitary

$$\hat{V}_0^{(\text{GC})} = e^{i\sqrt{s_0}\hat{J}_0}e^{-i\sqrt{s_0}\hat{D}_0}e^{-i\sqrt{s_0}\hat{J}_0}e^{i\sqrt{s_0}\hat{D}_0} = \hat{R}_0 + O(s_0^{3/2}) \ . \tag{16}$$

Following Ref. [23] we use the higher-order product formula (HOPF) with $\phi = \frac{1}{2}(\sqrt{5} - 1)$

$$\hat{Q}_0^{(\text{HOPF})} = e^{i\phi\sqrt{s_0}\hat{J}_0}e^{-i\phi\sqrt{s_0}\hat{D}_0}e^{-i\sqrt{s_0}\hat{J}_0}e^{i(\phi+1)\sqrt{s_0}\hat{D}_0}e^{i(1-\phi)\sqrt{s_0}\hat{J}_0}e^{-i\sqrt{s_0}\hat{D}_0} = \hat{R}_0 + O(s_0^2) \ . \tag{17}$$

One may note that when using the operators, the normalization results are no longer usable and so the optimization domain may correspond to the entire space.

In Fig. 4 we show the comparison of these approximations to the DBR unitary. We found that consistently HOPF accurately matches the performance of DBR, even if the scheduling must be adjusted. GC uses one less query to Hamiltonian simulation for $\hat{J}_0$ but it yields much less σ-decrease.

6   Conclusions

In this work, we presented strategies for optimizing double-bracket quantum algorithms. The overall goal is to prepare eigenstate on a quantum computer and we discussed different diagonalization cost functions which allow us to quantify the progress for this task. We then described the formulation of DBQAs as being rooted in selecting the duration of the recursive steps and selecting the DBR generators.

Fig. 1 reports that the utilities featured in Qibo allow to reliably find schedulings that yield effective diagonalization iterations. We next discussed parametrizations of DBR generators aimed at obtaining effective DBIs through optimization. We also showcased parametrizations suitable for BHMM DBIs which have analytical convergence guarantees. Fig. 2 explored the role played by spectral degeneracies of the DBR generators. For practical applications, we did not find it to be particularly key, especially when only a few recursion steps are available. Additionally, we showed that BHMM DBIs can get stuck for exactly solvable models such as TFIM and XXZ which motivates using adaptive DBIs.

Fig. 3 shows that adaptive DBIs achieve more diagonalization in fewer steps than the BHMM DBIs that we considered. We further showed that numerical optimizations of the DBR generators within prescribed parameter spaces allow us to extend the diagonalization convergence beyond what BHMM DBIs. Finally, we discuss the strategies for approximating DBIs in a way that is viable for quantum compiling. This is done by means of product formulas, e.g. the group commutator and Fig. 4 shows that much better results are obtained by a higher-order product formula approximation to each DBR in a DBI.

In summary, we conceptually broke down the task of optimizing DBQAs into three distinct stages. Our implementation in Qibo allowed us to showcase the performance of numerical optimization in each of these steps. By this, double-bracket quantum algorithms are provided by a more systematic optimization methodology which we hope will aid further progress in quantum compiling tasks such as eigenstate preparation of quantum many-body models involving large numbers of qubits.

7 Acknowledgements

References

[1] Steven R. White. Density matrix formulation for quantum renormalization groups. *Phys. Rev. Lett.*, 69:2863–2866, Nov 1992.

[2] Giuseppe Carleo and Matthias Troyer. Solving the quantum many-body problem with artificial neural networks. *Science*, 355(6325):602–606, February 2017.

[3] Julien Toulouse, Roland Assaraf, and C. J. Umrigar. Introduction to the variational and diffusion Monte Carlo methods, 2015.

[4] Jacob Biamonte and Ville Bergholm. Tensor networks in a nutshell, 2017.

[5] Yulong Dong, Lin Lin, and Yu Tong. Ground-state preparation and energy estimation on early fault-tolerant quantum computers via quantum eigenvalue transformation of unitary matrices. *PRX Quantum*, 3:040305, Oct 2022.

[6] Yimin Ge, Jordi Tura, and J. Ignacio Cirac. Faster ground state preparation and high-precision ground energy estimation with fewer qubits, 2018.

[7] Alberto Peruzzo et al. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1), July 2014.

[8] Marek Gluza. Double-bracket quantum algorithms for diagonalization. *Quantum*, 8:1316, April 2024.

[9] Guang Hao Low and Isaac L. Chuang. Hamiltonian simulation by qubitization. *Quantum*, 3:163, July 2019.

[10] Earl Campbell. Random compiler for fast Hamiltonian simulation. *Phys. Rev. Lett.*, 123:070503, Aug 2019.

[11] Dominic W. Berry et al. Simulating Hamiltonian dynamics with a truncated Taylor series. *Phys. Rev. Lett.*, 114:090502, Mar 2015.

[12] Andrew M. Childs and Yuan Su. Nearly optimal lattice simulation by product formulas. *Phys. Rev. Lett.*, 123:050503, Aug 2019.

[13] Matteo Robbiati, Edoardo Pedicillo, Andrea Pasquale, Xiaoyue Li, Andrew Wright, Renato M. S. Farias, Khanh Uyen Giang, Jeongrak Son, Johannes Knörzer, Siong Thye Goh, Jun Yong Khoo, Nelly H. Y. Ng, Zoë Holmes, Stefano Carrazza, and Marek Gluza. Double-bracket quantum algorithms for high-fidelity ground state preparation, 2024.

[14] Stavros Efthymiou et al. Qibo: a framework for quantum simulation with hardware acceleration. *Quantum Science and Technology*, 7(1):015018, December 2021.

[15] S. Carrazza, S. Efthymiou, M. Lazzarin, and A. Pasquale. An open-source modular framework for quantum computing. *Journal of Physics: Conference Series*, 2438(1):012148, February 2023.

[16] Stavros Efthymiou, Marco Lazzarin, Andrea Pasquale, and Stefano Carrazza. Quantum simulation with just-in-time compilation. *Quantum*, 6:814, September 2022.

[17] R.W. Brockett. Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems. *Linear Algebra and its Applications*, 146:79–91, 1991.

[18] J. B. Moore, R. E. Mahony, and U. Helmke. Numerical gradient algorithms for eigenvalue and singular value calculations. *SIAM J. Matrix Anal. Appl.*, 15(3):881–902, jul 1994.

[19] Stanisław D. Głazek and Kenneth G. Wilson. Renormalization of Hamiltonians. *Phys. Rev. D*, 48:5863–5872, Dec 1993.

[20] Franz Wegner. Flow-equations for Hamiltonians. *Annalen der Physik*, 506(2):77–91, 1994.

[21] Petro Liashchynskyi and Pavlo Liashchynskyi. Grid search, random search, genetic algorithm: A big comparison for nas, 2019.

[22] Shuhei Watanabe. Tree-structured parzen estimator: Understanding its algorithm components and their roles for better empirical performance, 2023.

[23] Yu-An Chen et al. Efficient product formulas for commutators and applications to quantum simulation. *Phys. Rev. Res.*, 4:013191, Mar 2022.