Assignment 8

https://github.com/jchryssanthacopoulos/quantum_information/tree/main/assignment_8

Quantum Information and Computing
AA 2022–23

James Chryssanthacopoulos
9 January 2023

Università degli Studi di Padova

Real-Space Renormalization Group

- **Real-space renormalization group** algorithm can be used to compute ground-state energy E_0 of quantum Ising model with traverse field in one dimension
- Following steps are repeated until $|E_0^{(i+1)} - E_0^i| < \epsilon$, where i is iteration number:
  1. Starting with Hamiltonian of system with N sites, $\hat{H}_N$, construct Hamiltonian by replicating system:

     $$\hat{H}_{2N} = \hat{H}_N \otimes \mathbb{1}_N + \mathbb{1}_N \otimes \hat{H}_N + \hat{H}_{\text{int}}$$

     where $\hat{H}_{\text{int}} = \hat{A}_N \otimes \hat{B}_N$ is interaction between left and right bipartitions, initialized to $\hat{A}_N = \mathbb{1}_{N-1} \otimes \sigma^x$, $\hat{B}_N = \sigma^x \otimes \mathbb{1}_{N-1}$. $\hat{H}_N$ is initialized to Hamiltonian of Ising model with N sites
  2. Diagonalize $\hat{H}_{2N}$, obtaining ground-state energy E_0^i. Construct projector onto 2^N eigenvectors with lowest energy, $P = \sum_{j=i}^{2^N} |E_j\rangle \langle E_j|$
  3. Project operators into subspace spanned by chosen eigenvectors:

     $$\hat{H}_N = P^\dagger \hat{H}_{2N} P, \quad \hat{A}_N = P^\dagger (\hat{A}_N \otimes \mathbb{1}_N) P, \quad \hat{B}_N = P^\dagger (\mathbb{1}_N \otimes \hat{B}_N) P$$

Implementation

Program computes ground-state energy given number of sites N, interaction strength λ, termination threshold ϵ, and maximum number of iterations

```
! compute initial Hamiltonian
H = lambda * non_interacting_hamiltonian(N) + interacting_hamiltonian(N)

! compute operators in interaction Hamiltonian
sigma_x = get_sigma_x()
A = tensor_product(identity(N - 1), sigma_x)
B = tensor_product(sigma_x, identity(N - 1))

iter = 1
gs_energy = -1
prev_gs_energy = 0

do while ((iter .le. max_iter) .and. abs(gs_energy - prev_gs_energy) > thres)
    prev_gs_energy = gs_energy

    H = H * 0.5
    A = 1 / sqrt(2.) * A
    B = 1 / sqrt(2.) * B

    call iter_real_space_rg(N, H, A, B, eigenvalues, diag_method, debug)

    ! compute energy density
    gs_energy = eigenvalues(1) / dble(N)

    if (debug) then
        print "('Iteration = ', i3, ', ground state energy = ', f9.4)", iter, gs_energy
    end if

    iter = iter + 1
end do
```

$\hat{H}_N$, $\hat{A}_N$, $\hat{B}_N$ initialized

Operators scaled to account for increased system size

System replicated to produce $\hat{H}_{2N}$

```
H_2N = tensor_product(H_N, identity(N)) + tensor_product(identity(N), H_N) + tensor_product(A, B)
A_2N = tensor_product(A, identity(N))
B_2N = tensor_product(identity(N), B)
```

Eigenvalues and eigenvectors computed, P computed by truncating

```
! compute using zheev
call find_eigenvalues_using_zheev(H_2N, eigenvalues, full_P)

! truncate
do ii = 1, ndim
    full_P_trunc(:, ii) = full_P(:, ii)
end do
```

Operators projected into subspace

```
full_P_trunc_transpose = transpose(conjg(full_P_trunc))

H_N = matmul(matmul(full_P_trunc_transpose, H_2N), full_P_trunc)
A = matmul(matmul(full_P_trunc_transpose, A_2N), full_P_trunc)
B = matmul(matmul(full_P_trunc_transpose, B_2N), full_P_trunc)
```

Results

Università degli Studi di Padova

- Energy density, $e \equiv E_0/N$, computed using RSRG compared to mean-field result:

$$e_{\mathrm{MF}} = \begin{cases} -1 - \lambda^2/4, & |\lambda| \leq 2 \\ -|\lambda|, & |\lambda| > 2 \end{cases}$$

- Energy densities are very similar, but deviate the most when $0 < |\lambda| < 2$, when external field is present but not strong enough to coordinate all spins
- Experiments showed that energy density doesn't change with N, as expected
- Number of iterations increases with N, except when field is weak