Spectral Functions with DMRG Revisited: Correction-vector with the Krylov-space Approach

A. Nocera and G. Alvarez
Computer Science & Mathematics Division and Center for Nanophase Materials Sciences, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA

PACS numbers: 78.70.Nx, 79.60.Bm, 02.70.Hm, 71.27.+a, 71.10.Fd, 74.20.-z, 75.10.Jm

Supplemental: Reproducing the numerical results.

The DMRG++ code can be obtained with:

```
git clone https://github.com/g1257/dmrgpp.git
```

and PsimagLite with:

```
git clone https://github.com/g1257/PsimagLite.git
```

To compile:

```
cd PsimagLite/lib
perl configure.pl
(you may now optionally edit Config.make)
make
cd ../../dmrgpp/src
perl configure.pl
(you may now optionally edit Config.make)
make
```

To obtain the contour plot of Figure 1, which corresponds to the dynamical spin structure factor of a Heisenberg chain with $L = 64$ sites, first use the input file `input_L64_Heisenberg.inp` to calculate the ground state of the system:

```
TotalNumberOfSites=64
NumberOfTerms=2

DegreesOfFreedom=1
GeometryKind=chain
GeometryOptions=ConstantValues
Connectors 1 1

DegreesOfFreedom=1
GeometryKind=chain
GeometryOptions=ConstantValues
Connectors 1 1

Model=Heisenberg
HeisenbergTwiceS=1

TargetElectronsTotal=64
TargetSzPlusConst=32

SolverOptions=twositedmrg
Version=version
OutputFile=DataGS.txt
InfiniteLoopKeptStates=64

FiniteLoops 5
```

```
31 1000 0 -62 1000 0
62 1000 0 -62 1000 0
62 1000 0

TruncationTolerance=1e-8
```

Above, a maximum of $m = 1000$ states are kept in the simulation by keeping the DMRG truncation error below 10^{-8}. One can run the code executing the command

`./dmrg -f input_L64_Heisenberg.inp.`

Once the ground state has been obtained, restart from ground state calculations. In order to calculate the expression in Eq. (13) of the main text, *restart* from the ground state calculation with input `input_L64_Heisenberg_x.inp` for each desired $\omega = x$ value of the spectrum:

```
TotalNumberOfSites=64
NumberOfTerms=2

DegreesOfFreedom=1
GeometryKind=chain
GeometryOptions=ConstantValues
Connectors 1 1

DegreesOfFreedom=1
GeometryKind=chain
GeometryOptions=ConstantValues
Connectors 1 1

Model=Heisenberg
HeisenbergTwiceS=1

TargetElectronsTotal=64
TargetSzPlusConst=32

InfiniteLoopKeptStates=64
FiniteLoops 3
-62 1000 2 62 1000 2 -62 1000 2

SolverOptions=CorrectionVectorTargetting,restart,twositedmrg
CorrectionA=0
Version=version

TruncationTolerance=1e-8

OutputFile=datax.txt

DynamicDmrgType=0
TSPSites 1 31
TSPLoops 1 1
TSPProductOrSum=sum
CorrectionVectorFreqType=Real

CorrectionVectorEta=0.075
CorrectionVectorAlgorithm=Krylov
TridiagEps=1e-7
TridiagSteps=1000

RestartFilename=DataGS.txt

GsWeight=0.1
```

```
CorrectionVectorOmega=x

TSPOperator=raw
RAW_MATRIX
2 2
-0.5 0
0 0.5
FERMIONSIGN=1
JMVALUES 0 0
AngularFactor=1
```

Notice the `restart` and `CorrectionVectorTargetting` flags in the `SolverOptions` of the code. User must substitute x with the value of desired value ω. This input restarts the DMRG simulations applying the operator S^z_c, represented explicitly at the bottom of the input as

```
TSPOperator=raw
RAW_MATRIX
2 2
-0.5 0
0 0.5
FERMIONSIGN=1
JMVALUES 0 0
AngularFactor=1
```

at the center of the chain $c = L/2 - 1 = 31$ using the flag `TSPSites 1 31`. The code calculates then the correction vector for frequency $\omega = x$ and $\eta = 0.075$. The expression in Eq.(13) of the main text is calculated as

```
./dmrg -f input_L64_Heisenberg_x.inp 'z',
```

where the option `'z'` means that we are measuring *in situ* the spin operator S^z_j on all the sites of the chain. In the output file `runForinputx.cout`, they are going to appear lines such as

```
j re 0 <PSI|z|P2> <PSI|P2>
..
j im 0 <PSI|z|P3> <PSI|P3>
```

where `|P2>` and `|P3>` represent the real and imaginary part of the correction vector $|x(\omega + i\eta)\rangle$, `|PSI>` the ground state, while `re` and `im` the real and imaginary part of the correlator $\langle\Psi_0|S^z_j|x(\omega + i\eta)\rangle$. Collecting the data for all sites j allows us to calculate the $S(k, \omega)$ spectrum according to equation Eq. (14) of the main text.

Above, the correction vector is calculated with the Krylov method presented in the main text, notice the lines

```
CorrectionVectorAlgorithm=Krylov
TridiagEps=1e-7
TridiagSteps=1000
```

where a maximum of 1000 iteration steps is set keeping the Lanczos error below 10^{-7}. If conjugate-gradient method is to be used, use the following lines instead

```
CorrectionVectorAlgorithm=ConjugateGradient
ConjugateGradientSteps=1000
ConjugateGradientEps=1e-7.
```

We finally provide the inputs to obtain Fig. 3 panel (d) of the main text. First, obtain the ground state of the Hubbard chain executing the command

```
./dmrg -f input_L48_Hubbard.inp
```

using the input `input_L48_Hubbard.inp` given by

```
TotalNumberOfSites=48
NumberOfTerms=1

DegreesOfFreedom=1
```

```
GeometryKind=chain
GeometryOptions=ConstantValues
Connectors 1  -1.0

Model=HubbardOneBand
hubbardU 48    4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
               4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
potentialV 96  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

SolverOptions=twositedmrg
Version=version
OutputFile=DataGS.txt
InfiniteLoopKeptStates=64
FiniteLoops 5
 23 1000 0 -46 1000 0
 46 1000 0 -46 1000 0
 46 1000 0

TargetElectronsUp=24
TargetElectronsDown=24
TruncationTolerance=1e-8
```

As done above for the Heisenberg case, *restart* from the ground state calculation with the input `input_L48_Hubbard_x.inp` for each desired $\omega = x$ value of the spectrum:

```
TotalNumberOfSites=48
NumberOfTerms=1

DegreesOfFreedom=1
GeometryKind=chain
GeometryOptions=ConstantValues
Connectors 1  -1.0

Model=HubbardOneBand
hubbardU 48    4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
               4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
potentialV 96  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
               0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

TargetElectronsUp=24
TargetElectronsDown=24

InfiniteLoopKeptStates=64
FiniteLoops 3
-46 1000 2 46 1000 2 -46 1000 2

SolverOptions=CorrectionVectorTargetting,restart,twositedmrg
CorrectionA=0
Version=version

TruncationTolerance=1e-8
OutputFile=datax.txt

DynamicDmrgType=0
```

```
TSPSites 1 23
TSPLoops 1 1
TSPProductOrSum=sum
CorrectionVectorFreqType=Real

CorrectionVectorEta=0.1

CorrectionVectorAlgorithm=ConjugateGradient
ConjugateGradientSteps=1000
ConjugateGradientEps=1e-7

RestartFilename=DataGS.txt

GsWeight=0.1
CorrectionVectorOmega=x

TSPOperator=raw
RAW_MATRIX
4 4
0 0 0 0
0 0.5 0 0
0 0 -0.5 0
0 0 0 0
FERMIONSIGN=1
JMVALUES 0 0
AngularFactor=1
```

The expression in Eq. (13) of the main text is calculated as

```
./dmrg -f input_L48_Hubbard_x.inp 'z'.
```

Collecting the data for the correlator $\langle \Psi_0 | S^z_j | x(\omega + i\eta) \rangle$ allows us to calculate the $S(k, \omega)$ spectrum according to equation Eq. (14) of the main text.