Doping evolution of charge and spin excitations in two-leg Hubbard ladders: comparing DMRG and FLEX results

A. Nocera,[1,2] Y. Wang,[1,3] N. D. Patel,[1,2] G. Alvarez,[4] T. A. Maier,[4] E. Dagotto,[1,2] and S. Johnston[1,5]

[1]*Department of Physics and Astronomy, University of Tennessee, Knoxville, Tennessee 37996, USA*
[2]*Materials Science and Technology Division, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA*
[3]*Département de Physique and Institut Quantique, Université de Sherbrooke, Sherbrooke, Québec J1K 2R1, Canada.*
[4]*Computational Science and Engineering Division and Center for Nanophase Materials Sciences, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA*
[5]*Joint Institute for Advanced Materials at The University of Tennessee, Knoxville, Tennessee 37996, USA*

Reproducing the numerical DMRG 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 compute the ground state properties of the Hubbard ladder studied in the main text, consider the following input file `input.inp`:

```
TotalNumberOfSites=96
NumberOfTerms=1

DegreesOfFreedom=1
GeometryKind=ladder
GeometryOptions=ConstantValues
Connectors 1  -1
Connectors 1  -1
LadderLeg=2

Model=HubbardOneBand

hubbardU 96 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6

potentialV 192 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 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 0 0 0 0 0

InfiniteLoopKeptStates=128
FiniteLoops 7
47 1000 0
-94 1000 0 94 1000 0
-94 1000 0 94 1000 0
-94 1000 0 94 1000 1

TargetElectronsUp=42
TargetElectronsDown=42
Threads=24
SolverOptions=twositedmrg,useSvd,MatrixVectorKron
Version=version
TruncationTolerance=1e-7
LanczosEps=1e-7
OutputFile=dataGS.txt
Orbitals=1
```

Above, a maximum of $m = 1000$ states are kept in the simulation by keeping the DMRG truncation error below 10^{-7}. We have furthermore considered $U = 6.0$, and doping $n = 0.875$. We notice that computer cluster resources are needed for the production runs. For this reason we include below an input file, `input_L8x2.inp`, valid for a shorter system size $L = 8 \times 2$ with $N = 14$ electrons ($n = 0.875$), using only $m = 200$ DMRG states. `input_L8x2.inp` could be executed quickly on a standard laptop.

```
TotalNumberOfSites=16
NumberOfTerms=1

DegreesOfFreedom=1
GeometryKind=ladder
GeometryOptions=ConstantValues
Connectors 1  -1
Connectors 1  -1
LadderLeg=2

Model=HubbardOneBand

hubbardU 16
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6

potentialV 32
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

InfiniteLoopKeptStates=128
FiniteLoops 5
7 200 0
-14 200 0 14 200 0
-14 200 0 14 200 1

TargetElectronsUp=7
TargetElectronsDown=7
SolverOptions=twositedmrg,useSvd,MatrixVectorKron
Version=version
TruncationTolerance=1e-7
LanczosEps=1e-7
OutputFile=dataGS.txt
Orbitals=1
```

Once the ground state $|\Psi_0\rangle$ has been obtained, restart from ground state calculations. In order to calculate the dynamical spin structure factor in frequency space

$$S_{j,c}(\omega + i\eta) = -\frac{1}{\pi}\mathrm{Im}\left[\langle\Psi_0|S_j^z\frac{1}{\omega - \hat{H} + E_0 + i\eta}S_c^z|\Psi_0\rangle\right], \quad (1)$$

restart from the ground state calculation with input `input_275.inp` for each desired, say $\omega = 0.275$, value of the spectrum. For a short $L = 8 \times 2$ ladder we have

```
TotalNumberOfSites=16
NumberOfTerms=1

DegreesOfFreedom=1
GeometryKind=ladder
GeometryOptions=ConstantValues
Connectors 1  -1
Connectors 1  -1
LadderLeg=2

Model=HubbardOneBand

hubbardU 16
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6

potentialV 32
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

InfiniteLoopKeptStates=128
FiniteLoops 4
-14 200 2 14 200 2
-14 200 2 14 200 2

TargetElectronsUp=7
TargetElectronsDown=7

SolverOptions=CorrectionVectorTargetting,twositedmrg,restart,useSvd,MatrixVectorKron,minimizeDisk
CorrectionA=0
Version=version
Threads=4
RestartFilename=dataGS.txt
TruncationTolerance=1e-5
LanczosEps=1e-7

OutputFile=data275.txt

DynamicDmrgType=0
TSPSites 1 6
TSPLoops 1 1
TSPProductOrSum=sum
CorrectionVectorFreqType=Real

CorrectionVectorEta=0.08
CorrectionVectorAlgorithm=Krylov
TridiagonalEps=1e-7
TridiagSteps=1000
Orbitals=1

GsWeight=0.1
```

```
CorrectionVectorOmega=0.275

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
```

Notice the `restart` and `CorrectionVectorTargetting` flags in the `SolverOptions` of the code. In eq. (1), $\hat{H}$ and E_0 represent the Hamiltonian and the ground state energy of the Hubbard model. This input restarts the DMRG simulations applying to the ground state $|\Psi_0\rangle$ the operator S^z_c, represented explicitly at the bottom of the input as

```
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
```

at the center of the ladder $c = L/2 - 2 = 6$ using the flag `TSPSites 1 6`. The code calculates then the correction vector

$$|x(\omega + i\eta)\rangle \equiv \frac{1}{\omega - \hat{H} + E_0 + i\eta} S^z_c |\Psi_0\rangle \tag{2}$$

for the frequency $\omega = 0.275$ and $\eta = 0.08$. The expression in eq. (1) is calculated as

$$S_{j,c}(\omega + i\eta) = -\frac{1}{\pi}\mathrm{Im}\left[\langle\Psi_0|S^z_j|x(\omega + i\eta)\rangle\right] \tag{3}$$

with

```
./dmrg -f input_275.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 ladder. In the output file `runForinput_275.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 imaginary and real part of the correction vector $|x(\omega + i\eta)\rangle$, `|PSI>` the ground state, while `im` and `re` the numerical values of imaginary and real 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 $S(q, \omega)$ in frequency momentum space.

In order to calculate the dynamical charge structure factor in frequency space

$$N_{j,c}(\omega + i\eta) = -\frac{1}{\pi}\mathrm{Im}\left[\langle\Psi_0|(n_j - \langle n_j\rangle)\frac{1}{\omega - \hat{H} + E_0 + i\eta}(n_c - \langle n_c\rangle)|\Psi_0\rangle\right], \tag{4}$$

one needs to first compute the local densities $\langle n_j\rangle$ in the ground state as

```
./observe -f input_L8x2.inp '<gs|n|gs>'.
```

Then, one needs to produce files `nx.txt` for each site `x` of the ladder, including the fluctuation operators matrices $(n_j - \langle n_j \rangle)$. The local occupation operator is given by

```
TSPOperator=raw
RAW_MATRIX
4 4
0 0 0 0
0 1 0 0
0 0 1 0
0 0 0 2
FERMIONSIGN=1
JMVALUES 0 0
AngularFactor=1
```

With the above input parameters, for the center site of the ladder we have $\langle n_c \rangle = 0.809363$. Therefore, the fluctuation operator matrix is

```
TSPOperator=raw
RAW_MATRIX
4 4
-0.809363 0 0 0
0 0.190637 0 0
0 0 0.190637 0
0 0 0 1.190637
FERMIONSIGN=1
JMVALUES 0 0
AngularFactor=1
```

and needs to be stored in a file `n6.txt`. Finally, the input file `inputN_275.inp` to compute the dynamical charge correlation function at frequency $\omega = 0.275$ is

```
TotalNumberOfSites=16
NumberOfTerms=1

DegreesOfFreedom=1
GeometryKind=ladder
GeometryOptions=ConstantValues
Connectors 1  -1
Connectors 1  -1
LadderLeg=2

Model=HubbardOneBand

hubbardU 16
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6

potentialV 32
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

InfiniteLoopKeptStates=128
FiniteLoops 4
-14 200 2 14 200 2
-14 200 2 14 200 2

TargetElectronsUp=7
TargetElectronsDown=7

SolverOptions=CorrectionVectorTargetting,twositedmrg,restart,useSvd,MatrixVectorKron,minimizeDisk
CorrectionA=0
```

```
Version=version
Threads=4
RestartFilename=dataGS.txt
TruncationTolerance=1e-5
LanczosEps=1e-7

OutputFile=data275.txt

DynamicDmrgType=0
TSPSites 1 6
TSPLoops 1 1
TSPProductOrSum=sum
CorrectionVectorFreqType=Real

CorrectionVectorEta=0.08
CorrectionVectorAlgorithm=Krylov
TridiagonalEps=1e-7
TridiagSteps=1000
Orbitals=1

GsWeight=0.1
CorrectionVectorOmega=0.275

TSPOperator=raw
RAW_MATRIX
4 4
-0.809363 0 0 0
0 0.190637 0 0
0 0 0.190637 0
0 0 0 1.190637
FERMIONSIGN=1
JMVALUES 0 0
AngularFactor=1
```

The code above calculates then the correction vector

$$|w(\omega+i\eta)\rangle \equiv \frac{1}{\omega-\hat{H}+E_0+i\eta}(n_c-\langle n_c\rangle)|\Psi_0\rangle. \tag{5}$$

The expression in eq. (4) is then calculated as

$$N_{j,c}(\omega+i\eta) = -\frac{1}{\pi}\mathrm{Im}\left[\langle\Psi_0|(n_j-\langle n_j\rangle)|w(\omega+i\eta)\rangle\right] \tag{6}$$

with

```
./dmrg -f inputN_275.inp 'n$.txt',
```

where the option `'n$.txt'` means that we are measuring *in situ* the charge operators $n_j-\langle n_j\rangle$ on all the sites of the ladder. In the output file `runForinput_275.cout`, they are going to appear lines such as

```
j re 0 <PSI|n$.txt|P2> <PSI|P2>
..
j im 0 <PSI|n$.txt|P3> <PSI|P3>
```

where `|P2>` and `|P3>` represent the imaginary and real part of the correction vector $|w(\omega+i\eta)\rangle$, `|PSI>` the ground state, while `im` and `re` the numerical values of imaginary and real part of the correlator $\langle\Psi_0|(n_j-\langle n_j\rangle)|w(\omega+i\eta)\rangle$. Collecting the data for all sites j allows us to calculate the $N(q,\omega)$ in frequency momentum space.

The Krylov method for the computation of the correction vector has been discussed in ref. 1 and computational details are provided in its supplemental material. In both inputs `input_275.inp` and `inputN_275.inp`, 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}.

---

[1] A. Nocera and G. Alvarez, Phys. Rev. E **94**, 053308 (2016), URL https://link.aps.org/doi/10.1103/PhysRevE.94.053308.