Multiple States in the Density Matrix Renormalization Group with The Singular Value Decomposition

E. F. D'Azevedo,[1] W. R. Elwasif,[1] N. D. Patel,[2] and G. Alvarez[3]

[1] *Computer Science and Mathematics Division, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA*
[2] *Department of Physics, The Ohio State University, Columbus, Ohio 43210, USA*
[3] *Computational Sciences and Engineering Division and Center for Nanophase Materials Sciences, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA*

(Dated: December 4, 2018)

I. KRONECKER PRODUCTS

This appendix lists a number of special properties of Kronecker Product (KPs). Details for the derivation can be found in [1].

Let matrix A be $r_A \times c_A$ and matrix B be $r_B \times c_B$, then $G = A \otimes B$ has shape $(r_B\, r_A) \times (c_B\, c_A)$ given as

$$A \otimes B = \begin{pmatrix} a_{11}B & a_{12}B & \dots & a_{1,c_A}B \\ a_{21}B & a_{22}B & \dots & a_{2,c_A}B \\ \vdots & \vdots & \vdots & \vdots \\ a_{r_A,1}B & a_{r_A,2}B & \dots & a_{r_A,c_A}B \end{pmatrix}.$$

The transpose of a KP is the KP of the transpose of its parts: $G^t = A^t \otimes B^t$. The KP is associative: $A \otimes (B \otimes C) = (A \otimes B) \otimes C$. Moreover, the matrix product of KPs can be evaluated as the KP of matrix products

$$(A \otimes B)\,(C \otimes D) = (A\,C) \otimes (B\,D).$$

KPs admit an efficient evaluation of matrix-vector multiplication that is of importance to DMRG:

$$\text{vector}(Y) = (A \otimes B)\,\text{vector}(X) = B\,X\,A^t,$$

where X can be reshaped as a $cB \times cA$ matrix and Y as $rB \times rA$ matrix. Let e_A denote the number of non-zero elements of A, and similarly for B. The computation of $Y = B\,X\,A^t$ can be evaluated as (i) $Z = B\,X$, then $Y = Z\,A^t$ using $2(e_B c_A + r_B e_A)$ or $2 r_B c_A (r_A + c_B)$ operations; (ii) $Z = X\,A^t$, then $Y = B\,Z$ using $2(c_B e_A + e_B r_A)$ or $2 r_A c_B (c_A + r_B)$ operations; (iii) in the special case where matrices A and B are very sparse, then explicitly evaluating $\text{vector}(Y) = (A \otimes B)\,\text{vector}(X)$ will require $2(e_A e_B)$ operations. Depending on the matrix shapes and sparsity, the most efficient method should be selected.

II. REPRODUCING THE NUMERICAL RESULTS.

The DMRG++ computer program 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
```

The documentation can be found at `https://g1257.github.io/dmrgPlusPlus/manual.html` or can be obtained by doing `cd dmrgpp/doc; make manual.pdf`.

The data needed for all the figures is in `RawData.tar.gz`.

To reproduce figure 2, a ground state run needs to be made with `./dmrg -f gs32x2.inp` where all inputs are in `RawData.tar.gz`. After the ground state run has finished, the batches and inputs for all frequency runs can be generated with

```
cp /path/to/dmrgpp/scripts/OmegaUtils.pm .
cp /path/to/dmrgpp/scripts/manyOmegas.pl .
perl manyOmegas.pl InputDollarized32x2.inp batchDollarized.pbs test
```

that can be launched by replacing `test` with `submit`, and by selecting the correct run for either A^+ or A^- in `batchDollarized`. When all frequency runs have finished, the post-processing to obtain the `.pgfplots` files is

```
cp /path/to/dmrgpp/scripts/procOmegas.pl .
perl procOmegas.pl -f InputDollarized32x2.inp -p -z
```

For A^-, runs need to be repeated with input `InputDollarized32x2upper.inp` and results added to those of A^+.

This supplemental will also be available at `https://g1257.github.io/papers/84/`.

[1] C. F. Van Loan, Journal of Computational and Applied Mathematics **123**, 85 (2000).