PEPS Introduction, contraction methods

1.   Motivation & Definition of PEPS

Goal: generalize MPS ideas to 2 dimensions!

Most obvious idea: 2D-DMRG, using a 'snake-MPS':

[White1996] (2D Heisenberg, nn & nnn interactions)

[Stoudenmire2012] (brief review)

[He2016] (2D Kagome)

[Zheng2017] (recent high-end application: striped order in 2D Hubbard model]

g

2D-DMRG is one of the most powerful & accurate methods for studying 2D quantum

lattice models. Main limitation: not enough entanglement. Entanglement entropy

but according to area law, we need

Reason for insufficiency: entanglement between   is encoded in a single bond.

p

Natural generalization: add more bonds between rows. This leads to PEPS ansatz

[Verstraete2004]

g

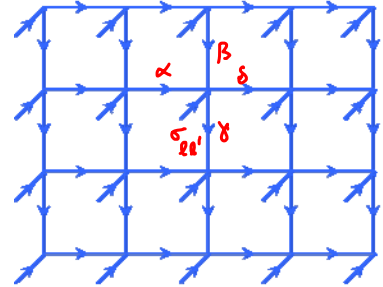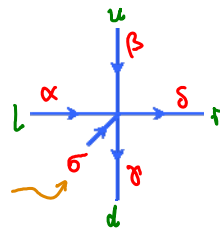Introduce 5-leg tensor for every site:

Sum over all virtual bonds linking neighboring sites:
p

       p

physical basis:

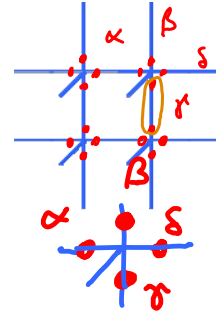   g

             g

contraction pattern:

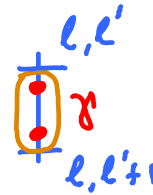Variationally minimize   .   # of variational parameters:

p

g

Why the name 'PEPS'? Verstraete & Cirac envisioned

generalization of AKLT construction:

Associate 4 'auxiliary particles' with each site:

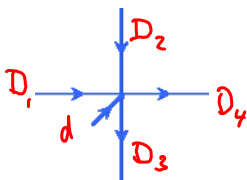Construct <u>e</u>ntangled <u>p</u>airs along bonds: g

g

Define projectors on each site:

Then

<u>General remarks</u>:   [Orus2014, Sec 5.2]

-PEPS are dense: any 2D state can be written as a PEPS, though possibly with

exponentially large D

-2D area law is satisfied:

-PEPS can handle polynomially-decaying correlations (in contrast to 1D MPS)

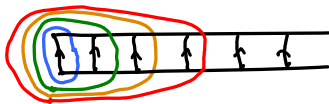-Exact contraction is #P hard -> contraction time

[#P hard = counting class for number of solutions to NP-complete problems

NP-complete = problems whose solution can be verified in polynomial time]

Why are exact contractions hard? Recall 1D situation:

Cheap contraction pattern: Expensive contraction

pattern: y

# of open indices   # of open indices

cost: cost:
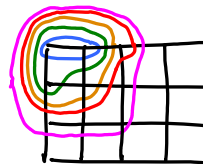
Moreover, if canonical form is used,

then contraction costs are very small:

(g   g   )

In 2D, growth of # of open indices is unavoidable:

open indices:

-Contraction cost would be manageable if a 'canonical form' were available!

Only started to be explored recently.

-'No exact canonical form exists' [Orus2014, Sec 5.2] (but this claim might be

outdated..)

-Restrictions to canonical forms are possible and possibly useful. [Zaletel2019,

Hagshenas2019]

2. Example: RVB state

Resonating valence bond (RVB) states are of continued

interest for constructing spin liquids. [Anderson1987],
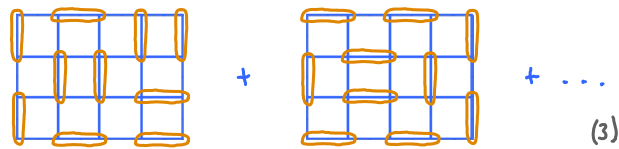
[Rokhsar1988] (high-Tc context)
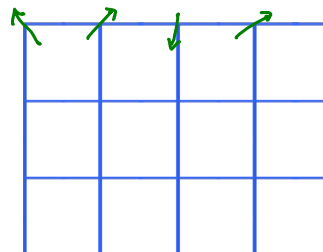
Canonical example: spin-1/2 on square lattice

'Dimer' or 'valence bond':

[sign conventions for bonds are needed and important]

RVB state: = (equal superposition of all possible dimer coverings of lattice)
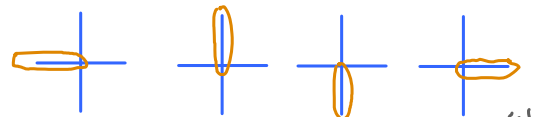
$$+ \; \ldots \tag{3}$$

VB fluctuations lower energy due to Hamiltonian matrix elements connecting different

configurations.

RVB state has a PEPS representation

[Verstraete2004, 2006]
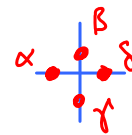
Defining properties of RVB state:

-each vertex has precisely one dimer attached

to it, so it can be involved in one of four possible states.

-Introduce four auxiliary sites per physical site,

each in one of the states

-define 'entangled pairs' using adjacent auxiliary sites from nearest neighbors of given

site:

-impose constraint: allow only one auxiliary spin-1/2 per physical site and identify it

with physical spin:

Projector on site   :

only nonzero elements of   -tensor:

PEPS form for RVB state:

Advantages of PEPS description of RVB state

-Dimer basis is hard to work with since individual

$\langle \sqsupset | \bowtie \rangle \neq 0$

components are not orthogonal:

Therefore explicit computations are easier in PEPS framework!

-PEPS description can be extended to larger classes of states, e.g. including longer-ranged bonds [Wang2013]

-'Parent Hamiltonian' (for which RVB state is exact ground state) can constructed systematically, but it is complicated: 19-site interaction [Schuch2012], 12-site interaction [Zhou2014].

PEPS II: Contraction Techniques

PEPS contractions, needed for computing expectation values, are #P hard when performed exactly -> impossible in practice. Various schemes for performing PEPS contractions approximately have been and continue to be proposed (this is a question at the cutting edge of current research). We describe two elementary schemes: the first uses purely MPS methodology, the second uses 2D transfer matrices.

2.   PEPS via finite-size MPS

Goal: reduce PEPS computations to familiar MPS computations ('reuse code') [Verstraete2004] (proof of principle application: spin 1/2 Heisenberg model on 4x4 lattice)
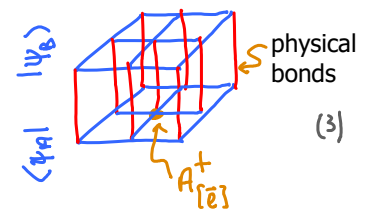
[Murg2007] 2D hard-core bosons = 2D XY model in uniform field

Strategy for PEPS ground state search: T
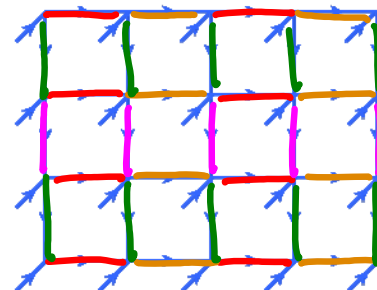
(i) Imaginary time evolution

(ii) truncate back to   , via variational compression (one tensor at a time) of

physical
bonds

(3)

T

( )   g   gy

Computation of   can be performed by repeated application of
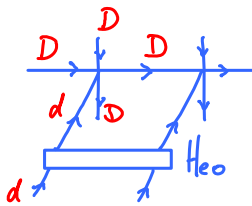
MPOs to MPS!

(i) <u>Trotter time evolution</u> [Murg2007, Sec IV]

Apply one such factor to all corresponding bonds:

Following bad habit of PEPS literature, we mostly drop arrows in our diagrams below…


3. time evolve

either   4. reshape

p   g   g

or   4'. truncate

p   g   g

Summary of steps:

1. Reshape 5-leg tensors into 3-leg MPS form.

2. (suggested by A. Weichselbaum) SVD to project   dim space onto

   (making step 3 cheaper) - known as bond projection.

3. Time-evolve. This causes increase in dimension of central bond from

4. Reshape 3-leg MPS tensors back into 5-leg PEPS form. These define the time-

   evolved PEPS. Its central bond dimension must still be decreased, using variational

   compression in step (ii) below. [Comment: when reshaping a big   space is

   split into the direct product of 3 smaller   spaces. This involves much freedom

   which should be used to minimize entanglement. That is partially achieved as a

   byproduct of the variational compression of (ii), but more direct strategies involving

   'disentanglers' (to be discussed in a subsequent lecture), might lead to further

   improvements]

4.' To get a good starting point for (ii), one may use SVD to truncate central bond from

(Note: performing only this truncation, without subsequently performing (ii), would yield

a suboptimal result, since this truncation includes no information about environment of

open legs. By contrast, such information is included during variational compression (ii),

hence the latter is essential.

(ii) Variational compression [Meng2007, Sec II]

To compute the overlaps in   , we need objects like
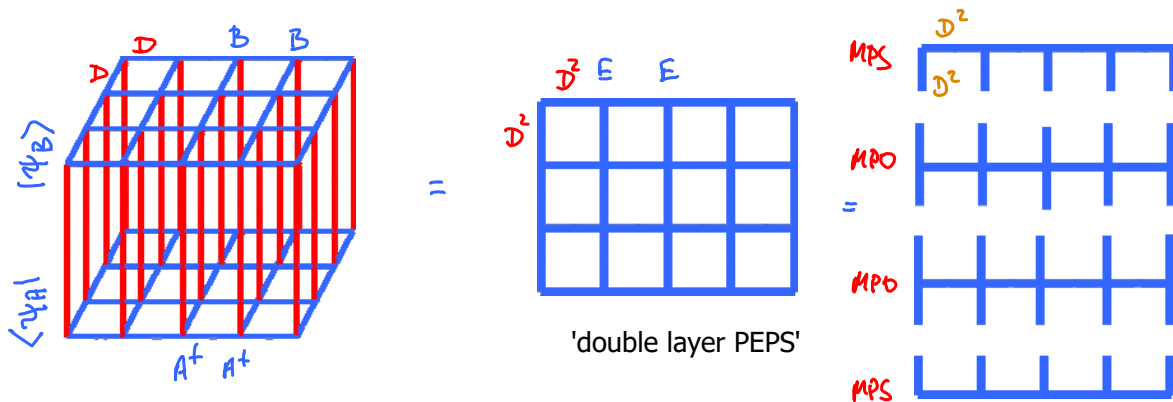
'double layer PEPS'

where   is a 4-leg tensor with bond dimension

View first and last rows of double-layer PEPS as MPS, and rows in-between as MPOs.

Each application of an MPO increases bond dimension:

variational
compression

Iteratively apply MPOs. In last step, apply   to obtain

which looks like a contracted MPS.

For   one can proceed similarly but with one   missing from lower
(bra) layer, and correspondingly some open legs.

In this manner, compute   and   , then solve

Sweep until convergence, the converged result yields   . Then return to (i), with

next Trotter gate.

Results for ground state energy [Murg2007]
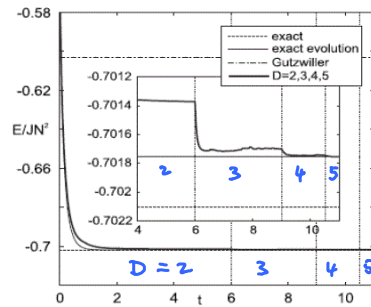
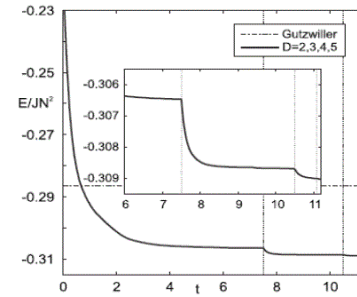Bond dimension was increased at three distinct times, each yielding further decrease in ground state energy.

FIG. 3. Energy as a function of time for the imaginary time evolution of the system of hard-core bosons on a 4×4 lattice. The evolutions are performed sequentially with PEPS of virtual dimension $D=2$, $D=3$, $D=4$, and $D=5$. The times at which D is increased are indicated by vertical lines. For comparison, the exact ground-state energy, the exact imaginary time evolution, and the energy of the optimal Gutzwiller ansatz are included.

FIG. 4. Energy as a function of time for the imaginary time evolution of the system of hard-core bosons on a 11×11 lattice. The evolutions are performed sequentially with PEPS of virtual dimension $D=2$, $D=3$, $D=4$, and $D=5$. The times at which D is increased are indicated by vertical lines. For comparison, the energy of the optimal Gutzwiller ansatz is included.
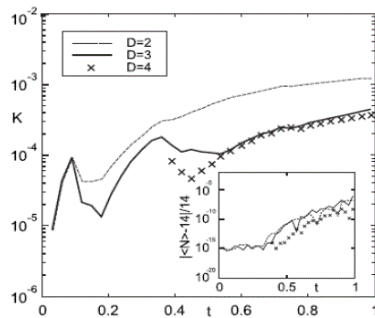
Accuracy and numerical efficiency

FIG. 9. Distance K between the time-evolved state and the state with reduced virtual dimension. The virtual dimensions $D=2$, $D=3$, and $D=4$ are included. The distance is plotted for the evolution of a Mott distribution with $N=14$, as explained in Fig. 8. From the inset, the deviation of the particle number from the value 14 can be gathered.

Main bottleneck: # of multiplications   , memory

-> 11x11 lattice with D=5 took 55 hours for 1 time step, required 2 GB of memory

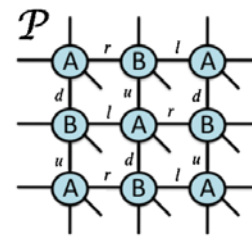[computations were performed on a workstation with 3.0GHZ Intel Xeon processor]

'Going beyond D=5 is difficult at the moment' (2007)

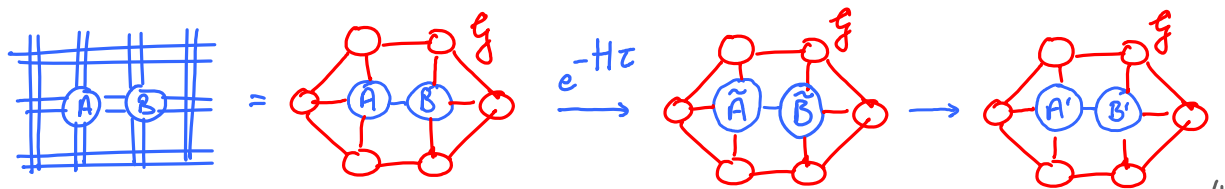By now,   is possible when exploiting non-Abelian symmetries.

2. Infinite-size PEPS (iPEPS)

[Jordan2008] 2D quantum Ising model

Goal: exploit translational invariance by

borrowing tools from iTEBD.

Strategy: represent entire lattice in terms of just two tensors,

double-layer PEPS   system-environment   time-evolved version   variational compression

Steps:

(i) Choose specific bond between two tensors,   forming unit cell. Compute

environment,   built from   too, using transfer matrix tools and

iTEBD.

(ii) Perform imaginary time evolution on specified bond between active sites,

(iii) Use variational compression, involving   to compress

Then move on to next bond and iterate...

Overall structure

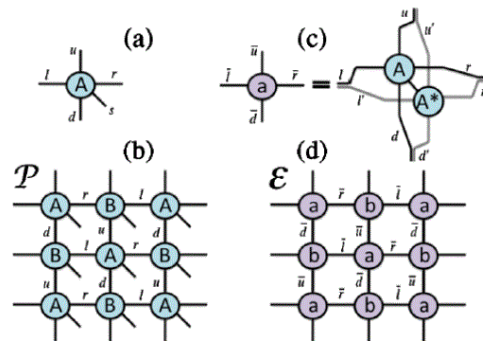Exploit translational invariance:

Use two-site active unit cell,

for bipartite covering of lattice.

FIG. 1 (color online).   Diagrammatic representations of (a) a
PEPS tensor A_{sudlr} with one physical index s and four inner
indices u, d, l and r; (b) local detail of the tensor network $\mathcal{P}$ for
an iPEPS. Copies of tensors A and B are connected through four
types of links; (c) reduced tensor a of Eq. (2); and (d) local detail
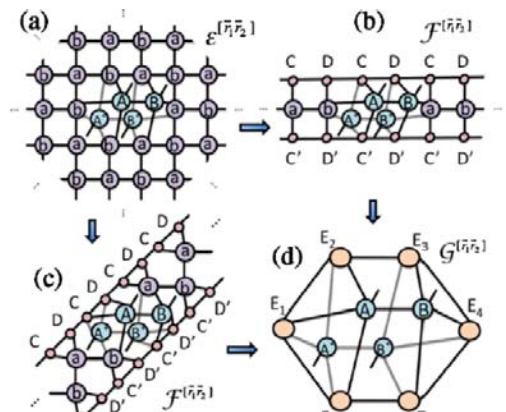of the tensor network $\mathcal{E}$.

Then iPEPS depends on

coefficients

Single-layer iPEPS:

Double-layer iPEPS:

(i) <u>Contract out environment for specified bond</u>

Approximate the double-layer iPEPS,   ,

by an infinite strip,

(either horizontally, vertically, or diagonally)

surrounding the active sites.

Approximate strip g by 'environment',

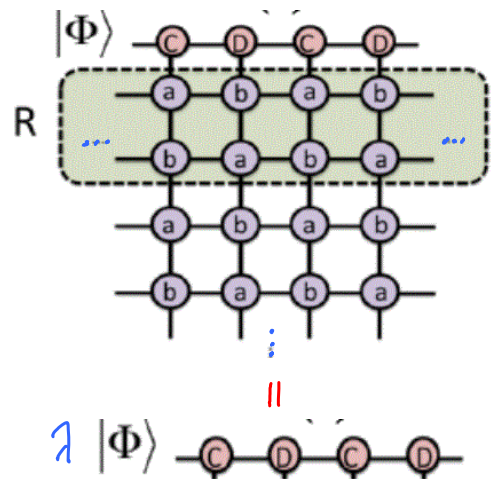encoded in six E-tensors.

Computation of strip g using transfer matrix methods

Define transfer matrix, R, built from two infinite rows of

a, b:
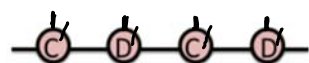
Compute its dominant right eigenvector,

then

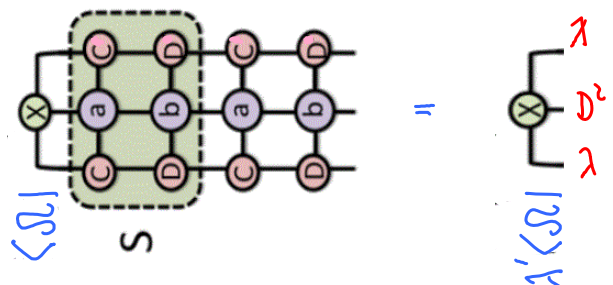represents entire 'upper half plane' (above active sites).

In practice:

is computed using iTEBD methods, with　　　instead of　　　as evolution operator.

Similarly, compute dominant left eigenvector,

then

represents lower

half-plane.

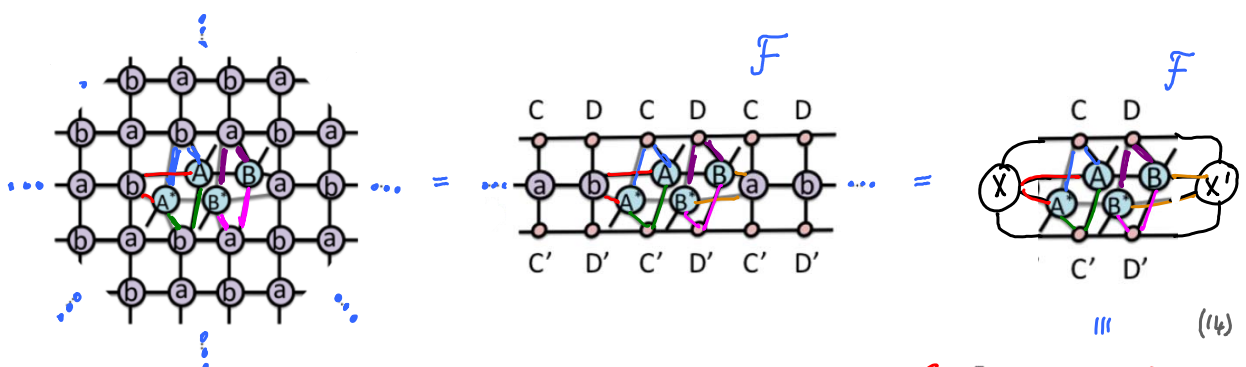Next, define transfer matrix S from the six tensors a, b, C, D, C', D':

Compute its dominant left eigenvector,

then   represents left part of strip.

Similarly, compute its dominant right eigenvector,

then   represents right part of strip.

Construction of E-tensors representing environment (e.g. for r-bond)
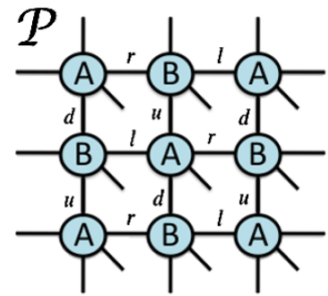

Note: 'open up' those double-legs from double-layer tensors

a, b (and C, D, C', D' ,X, X') which are connected with single-layer tensors A, B, A', B'.
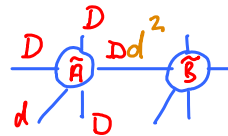
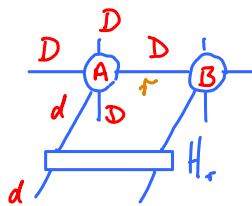Thus, environment (computed from   )

is represented by
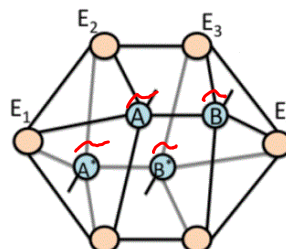
(ii) Imaginary time evolution of active basis sites

g   g   g   g

Time-evolve, e.g. for r-bond

g   g   g   g

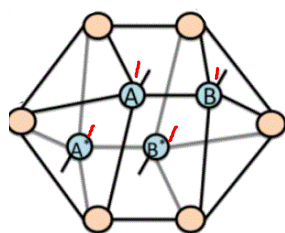g   g   g   g

(iii) Compress   via variational compression

Find new tensors   (with bond dimension   ), which maximize overlap with

time-evolved state.

variationally maximize overlap

The updated A', B' define the new iPEPS. Move to next bond, iterate steps (i) to (iii) until convergence.

How to deal with the environment at each time step?

-'Full update': recompute environment after each time step (most accurate, most expensive)

-'Fast full update': use same environment for several time steps (less accurate, faster)

-'Simple update': don't compute full environment, but approximate it using bond tensors (least accurate, but often still reasonable, by far the fastest)

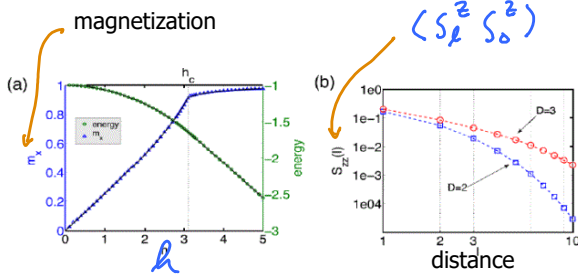Results for transverse field Ising model [Jordan2008]

FIG. 4 (color online). (a) Transverse magnetization m_x and energy per site e as a function of the transverse magnetic field h. The continuous line shows series expansion results (to 26th and 16th order in perturbation theory) for h smaller and larger than $h_c \approx 3.044$ [13]. Increasing D leads to a lower energy per site e. For instance, at $h = 3.1$, $e(D = 2) \approx -1.6417$ and $e(D = 3) \approx -1.6423$. (b) Two-point correlator $S_{zz}(l)$ near the critical point, $\lambda = 3.05$. For nearest neighbors, the correlator quickly converges as a function of D, whereas for long distances we expect to see convergence for larger values of D.
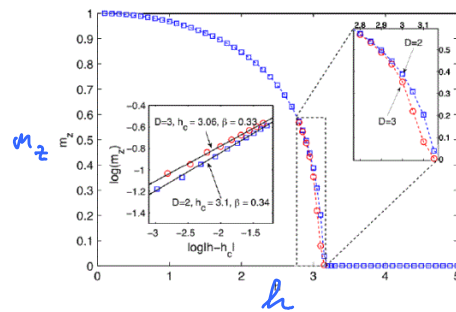
FIG. 5 (color online). Magnetization $m_z(\lambda)$ as a function of the transverse magnetic field λ. Dashed lines are a guide to the eye. We have used the diagonal scheme for $(D, \chi) = (2, 20)$, $(3,25)$ and $(4,35)$ [15] (the vertical or horizontal scheme leads to comparable results with slightly smaller χ.) The inset shows a log plot of m_z versus $|\lambda - \lambda_c|$, including our estimate of λ_c and β. The continuous line shows the linear fit.