Applying an MPO onto an MPS

Author: Seung-Sup Lee

We demonstrate that the application of matrix product operator (MPO) onto a matrix product state (MPS) results in another MPS. In this tutorial, we consider an MPO representation of the Hamiltonian and its ground state(s) as an MPO and an MPS, respectively.

To start with, we consider the spin-1/2 Heisenberg model on a short chain,

$$\hat{H} = J \sum_{\ell=1}^{L-1} \hat{\vec{S}}_\ell \cdot \hat{\vec{S}}_{\ell+1},$$

where we choose $J = 1$ without loss of generality.

By treating a short chain of length $L = 6$, we can obtain the full Hamiltonian and the exact ground state by iteratively constructing the identiy tensors without truncating them. At all iterations except for the last iteration, we do not rotate the basis into the energy eigenbasis. It is necessary to compare the Hamiltonian constructed iteratively (represented by a matrix Hnow below) with the MPO form of the Hamiltonian (to be obtained in the next section).

```
clear

J = 1; % interaction strength
L = 6; % chain length
M = cell(1,L); % MPS

[S,I] = getLocalSpace('Spin',1/2);

Hprev = 0; % initialize Hamiltonian
Aprev = 1; % identity for the vacuum

for itN = (1:L)
    % rank-3 identity tensor for the current iteration
    Anow = getIdentity(Aprev,2,I,2,[1 3 2]);

    % contract the Hamiltonian up to the last iteration with
    % ket and bra tensors
    Hnow = updateLeft(Hprev,2,Anow,[],[],Anow);

    if itN > 1
        % spin-spin interactopin
        Hsp = updateLeft(Sprev,3,Anow, ...
            permute(conj(S),[2 1 3]),3,Anow);
        Hnow = Hnow + J*Hsp;
    end

    % spin operator at the current site; to be used for
    % generating the coupling term at the next iteration
```

```
    Sprev = updateLeft([],[],Anow,S,3,Anow);

    if itN == L % diagonalize Hamiltonian
        [V,D] = eig((Hnow+Hnow')/2);
        [E_G,minid] = min(diag(D));
        % select only the ground state
        M{itN} = contract(Anow,3,2,V(:,minid),2,1,[1 3 2]);
    else
        M{itN} = Anow;
    end

    Aprev = Anow;
    Hprev = Hnow;

    disptime(['#',sprintf('%02i/%02i',[itN,L]),' : ', ...
        'NK=',sprintf('%i/%i',[size(M{itN},2),size(Hnow,2)])]);
end
```

```
22-09-20 16:30:36 | #01/06 : NK=2/2
22-09-20 16:30:36 | #02/06 : NK=4/4
22-09-20 16:30:36 | #03/06 : NK=8/8
22-09-20 16:30:36 | #04/06 : NK=16/16
22-09-20 16:30:36 | #05/06 : NK=32/32
22-09-20 16:30:36 | #06/06 : NK=1/64
```

The ground-state energy is:

```
E_G
```

```
E_G = -2.4936
```

Currently, M is in a left-canonical form. By bringing it into a right-canonical form, the size of the constituent tensors can be decreased.

```
M
```

M = 1×6 cell

	1	2	3	4	5	6
1	1×2×2 double	2×4×2 double	4×8×2 double	8×16×2 double	16×32×2 double	32×1×2 double

```
M = canonForm(M,0,[],[]) % right-canonical form
```

M = 1×6 cell

	1	2	3	4	5	6
1	1×2×2 double	2×4×2 double	4×8×2 double	8×4×2 double	4×2×2 double	2×1×2 double

MPO representation of Heisenberg chain Hamiltonian

We construct the MPO representation of the chain Hamiltonian, following the recipe given in a lecture. The MPO consists of the same tensor, called bulk tensor, except for the first and the last sites. A bulk tensor is rank-4, and

its legs are sorted in the bottom-top-left-right order. The bottom (top) leg is to be contracted with the second leg of bra (ket) tensor.

Let's first generate a bulk tensor.

```
% bulk tensor for each chain site
Hloc = cell(5,5);
Hloc(:) = {zeros(size(I))};
Hloc{1,1} = I;
for ito = (1:size(S,3)) % different components of spin operators
    Hloc{ito+1,1} = S(:,:,ito);
    Hloc{end,ito+1} = J*S(:,:,ito)';
end
Hloc{end,end} = I;
Hloc = cell2mat(reshape(Hloc,[1 1 size(Hloc,1) size(Hloc,2)]));
```

For the tensors at the first and the last sites, we project the bulk tensor onto a specific index of its left and right legs, respectively. So the left and right legs, respectively, become dummy legs with singleton dimension.

```
% MPO for the full chain
Hs = cell(1,L);
Hs(:) = {Hloc};
Hs{1} = Hs{1}(:,:,end,:); % choose the last index of the left leg
Hs{end} = Hs{end}(:,:,:,1); % choose the first index of the right leg
```

To check whether the MPO construction is right, we compare the MPO and the Hamiltonian from iterative construction (represented by `Hnow`) We first contract the tensors of `Hs` to make a high-rank tensor acting onto all the chain sites. And we permute and reshape the high-rank tensor into a big matrix, which can be directly compared with `Hnow`.

```
Hs_tot = 1; % initialize
for itN = (1:L)
    Hs_tot = contract(Hs_tot,2*itN,2*itN,Hs{itN},4,3);
end
% permute the left- and rightmost legs to the end
Hs_tot = permute(Hs_tot,[(2:2*L+2) 1]);

% merge the incoming legs into a thick incoming leg;
% merge the outgoing legs into a thick outgoing leg
Hs_tot = permute(Hs_tot,[(1:2:2*L) (2:2:2*L)]);
Hs_tot = reshape(Hs_tot,(size(I,1)^L)*[1 1]);
```

The two forms of the Hamiltonian are equivalent.

```
max(abs(Hs_tot(:)-Hnow(:)))
```

```
ans = 0
```

Application of MPO onto MPS

We apply the MPO representation of Hamiltonian $\hat{H}$ (represented by a cell array `Hs`) onto the ground state $|\psi\rangle$ (by a cell array `M`). The tensors at each site, `Hs{n}` and `M{n}`, are contracted, and their horizontal legs are merged by using isometries. The result is a rank-3 tensor `HM{n}`, which constitutes a MPS form of $\hat{H}|\psi\rangle$.

```
HM = cell(1,L);
for itN = (1:L)
    HM{itN} = contract(Hs{itN},4,2,M{itN},3,3);
    % leg order: Hbottom-Hleft-Hright-Mleft-Mright

    % isometry to merge left legs
    if itN == 1
        Aleft = 1; % there are only dummy legs
    else
        % use Aright from the previous iteration,
        % to be a valid insertion of identity
        Aleft = conj(Aright);
    end

    % isometry to merge right legs
    Aright = getIdentity(HM{itN},3,HM{itN},5);

    % contract isometries
    HM{itN} = contract(Aleft,3,[1 2],HM{itN},5,[2 4]);
    % leg order: Aleft-Hbottom-Hright-Mright
    HM{itN} = contract(HM{itN},4,[3 4],Aright,3,[1 2],[1 3 2]);
end
```

As shown in the above tensor diagram, the bond dimensions have increased.

```
HM
```

HM = 1×6 cell

	1	2	3	4	5	6
1	1×10×2 double	10×20×2 double	20×40×2 double	40×20×2 double	20×10×2 double	10×1×2 double

The bond dimensions in the MPS form of $\hat{H}|\psi\rangle$ can be compressed by performing a "round trip" of bringing into canonical forms; first into left-canonical, then into right-canonical.

```
[HM,HMnorm] = canonForm(HM,L,[],[]) % left-canonical
```

HM = 1×6 cell

	1	2	3	4	5	6
1	1×2×2 double	2×4×2 double	4×8×2 double	8×16×2 double	16×8×2 double	8×1×2 double

HMnorm = 2.4936

```
[HM,HMnorm2] = canonForm(HM,0,[],[]) % here the second output should be 1
```

HM = 1×6 cell

	1	2	3	4	5	6
1	1×2×2 double	2×4×2 double	4×8×2 double	8×4×2 double	4×2×2 double	2×1×2 double

```
HMnorm2 = 1.0000
```

Now, the tensors have the same sizes as those in `M`.

```
M
```

`M` = 1×6 cell

	1	2	3	4	5	6
1	1×2×2 double	2×4×2 double	4×8×2 double	8×4×2 double	4×2×2 double	2×1×2 double

The value of the norm `HMnorm` (which means $\|\widehat{H}|\psi\rangle\|$) is equal to the absolute value of the ground-state energy (represented by `E_G`).

```
HMnorm - abs(E_G) % zero up to numerical noise
```

```
ans = 4.4409e-16
```

For later purpose, we let the first tensor `HM{1}` absorb `HMnorm`.

```
HM{1} = HM{1}*HMnorm;
```

We can also explicitly compute $\langle\psi|\widehat{H}|\psi\rangle$ by contracting `HM` and `M`.

```
MHM = 1;
for itN = (1:L)
    MHM = updateLeft(MHM,2,M{itN},[],[],HM{itN});
end
MHM - E_G % zero up to numerical noise
```

```
ans = -1.7764e-15
```

Here we know that `M` represents the ground state, since the Hamiltonian can be exactly diagonalized, thank to small system size. For general systems, however, there is no straightforward way to verify whether a given state $|\psi\rangle$ is the true ground state of the system, not being a local mimima.

Exercise (a): MPO representation of the AKLT Hamiltonian

We have studied the AKLT states in the previous tutorials. The AKLT states are the ground states of the AKLT model, which is a chain of spin-1's that interact via nearest-neighbor interactions,

$$\widehat{H} = \sum_{\ell=1}^{N-1} \left[(\widehat{\vec{S}}_\ell \cdot \widehat{\vec{S}}_{\ell+1}) + \frac{1}{3} (\widehat{\vec{S}}_\ell \cdot \widehat{\vec{S}}_{\ell+1})^2 \right].$$

The first term on the right-hand side is the Heisenberg interaction. The second term is a biquadratic term that has a form of the squared Heisenberg interaction. We consider an open boundary condition.

(i) Construct a bulk tensor of an MPO representation of the AKLT model Hamiltonian. The tensors of the MPO at the first and the last sites can be obtained in the same way as above, by projecting onto specific indices.

(ii) Check whether your construction of the MPO is correct, by comparing with the Hamiltonian from iterative diagonalization of a short chain, e.g., $L = 4$.

Exercise (b): Confirm whether the AKLT states are the eigenstates of the AKLT Hamiltonian

In a previous tutorial, we have learned that there are four AKLT states $|\psi(\alpha, \beta)\rangle$ $(\alpha, \beta = 1, 2)$ in case of the open boundary condition. For all four AKLT states $|\psi(\alpha, \beta)\rangle$, (i) compute the expectation values $\langle\psi(\alpha, \beta)|\widehat{H}|\psi(\alpha, \beta)\rangle$ and (ii) confirm $\langle\psi|\widehat{H}|\psi\rangle^2 - \langle\psi|\widehat{H}^2|\psi\rangle = 0$. For this exercise, consider a long chain of length $L = 50$.