

MOTIVATION

In many problems involving spin-1/2 systems with isotropic Heisenberg interaction one needs to simplify products of scalar and mixed products of Pauli matrices.

Typical Hamiltonian:

$$H = \sum_{\langle i,j \rangle} (\sigma_i \sigma_j),$$

where $\langle i, j \rangle$ denotes neighbouring sites

Typical problems:

- squaring parametrization of density matrices, ρ [1] (see also talk by N. Il'in and poster by E. Spagina)
- variational principle [2]
- Schrodinger equation in the form $H\rho = E\rho$ (see e.g. [3])

Important: account for all (possibly noncommuting) symmetries simultaneously!

STATIONARY SCHRODINGER EQUATION FOR A DENSITY MATRIX

ρ can be used instead of Ψ in a stationary Schrodinger equation (see e.g.[3]):

$$H\rho = E\rho.$$

If the Hamiltonian of N spins 1/2 is invariant w.r.t. rotations (i.e. $SU(2)$) and time reversal one can seek ρ in form

$$\rho = \frac{1}{2^N} (1 + a_{i,j}(\sigma_i \sigma_j) + b_{i,j,k,l}(\sigma_i \sigma_j)(\sigma_k \sigma_l) + \dots)$$

Example:

$$H = (\sigma_1, \sigma_2) + (\sigma_2, \sigma_3)$$

Additional symmetry of the Hamiltonian: $1 \leftrightarrow 3$

$$\rho = \frac{1}{8} (1 + a((\sigma_1, \sigma_2) + (\sigma_2, \sigma_3)) + b(\sigma_1, \sigma_3))$$

$$\begin{aligned} H\rho &= \frac{1}{8} ((\sigma_1, \sigma_2) + (\sigma_2, \sigma_3))(1 + a((\sigma_1, \sigma_2) + (\sigma_2, \sigma_3)) + b(\sigma_1, \sigma_3)) = \\ &= \frac{1}{8} (6a + (1 + b - 2a)(\sigma_1, \sigma_2) + 2a(\sigma_1, \sigma_3) + (1 + b - 2a)(\sigma_2, \sigma_3)) = \\ &= \frac{1}{8} (E + Ea(\sigma_1, \sigma_2) + Ea(\sigma_2, \sigma_3) + Eb(\sigma_1, \sigma_3)) = E\rho \end{aligned}$$

System of three quadratic equations:

Solution:

$$\begin{cases} 6a - E = 0 \\ 1 - 2a + b - aE = 0 \\ 2a - bE = 0 \end{cases} \quad \begin{cases} a = -\frac{1}{2} \\ b = \frac{1}{3} \\ E = -4 \end{cases} \quad \begin{cases} a = 0 \\ b = -1 \\ E = 0 \end{cases} \quad \begin{cases} a = \frac{1}{3} \\ b = \frac{1}{3} \\ E = 2 \end{cases}$$

VARIATIONAL PRINCIPLE

A variational principle of [2] allows to bound the ground state energy **from below**. For the Heisenberg model on the square lattice it reads

$$E_{gs}/N \geq \frac{2}{M} \inf_{\rho_c} \text{tr} H_c \rho_c,$$

where E_{gs}/N is the ground state energy per site, H_c is the Hamiltonian of an arbitrary cluster on the square lattice, M – number of couplings in this cluster, ρ_c – its density matrix respecting all symmetries of the *original model*.

Example: $H_c = (\sigma_1, \sigma_5) + (\sigma_2, \sigma_5) + (\sigma_3, \sigma_5) + (\sigma_4, \sigma_5)$

To ensure $\rho_c \geq 0$, $\text{tr} \rho_c = 1$, $\rho_c^\dagger = \rho_c$ we use **squaring parametrization** [3]:

$$\rho_c = \frac{\tau^2}{\text{tr} \tau^2}$$

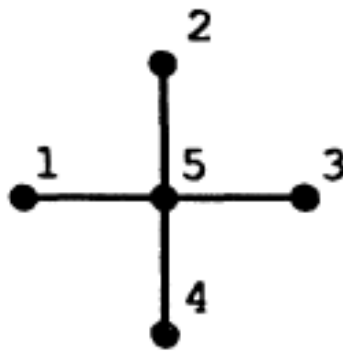
τ accounting for all symmetries:

$$\begin{aligned} \tau &= 1 + \\ &a_1((\sigma_1, \sigma_5) + (\sigma_2, \sigma_5) + (\sigma_3, \sigma_5) + (\sigma_4, \sigma_5)) + \\ &a_2((\sigma_1, \sigma_3) + (\sigma_2, \sigma_4)) + \\ &a_3((\sigma_1, \sigma_2) + (\sigma_2, \sigma_3) + (\sigma_3, \sigma_4) + (\sigma_4, \sigma_1)) + \\ &b_1((\sigma_1, \sigma_5)(\sigma_2, \sigma_3) + (\sigma_1, \sigma_5)(\sigma_4, \sigma_3) + (\sigma_2, \sigma_5)(\sigma_3, \sigma_5)(\sigma_1, \sigma_4) + \\ &+ (\sigma_3, \sigma_5)(\sigma_4, \sigma_1) + (\sigma_3, \sigma_5)(\sigma_2, \sigma_1) + (\sigma_4, \sigma_5)(\sigma_1, \sigma_2) + (\sigma_4, \sigma_5)(\sigma_3, \sigma_2)) + \\ &b_2((\sigma_1, \sigma_2)(\sigma_3, \sigma_4) + (\sigma_2, \sigma_3)(\sigma_4, \sigma_1)) + \\ &b_3(\sigma_1, \sigma_3)(\sigma_2, \sigma_4) + \\ &b_4((\sigma_1, \sigma_5)(\sigma_2, \sigma_4) + (\sigma_2, \sigma_5)(\sigma_1, \sigma_3) + (\sigma_3, \sigma_5)(\sigma_2, \sigma_4) + (\sigma_4, \sigma_5)(\sigma_1, \sigma_3)) \\ \text{tr} H_c \rho_c &= \frac{(24a_1 - 24a_1^2 + 24a_1a_2 + 48a_1a_3 + 48a_2b_1 + 192a_3b_1 - 192b_1^2 + 192b_1b_2 + 48b_1b_3 + 72a_2b_4 + 48a_3b_4 - 96b_1b_4 + 48b_2b_4 + 72b_3b_4 - 72b_4^2)}{1 + 12a_1^2 + 6a_2^2 + 12a_3^2 + 96b_1^2 + 24b_2^2 + 12b_2b_3 + 9b_3^2 + 48b_1b_4 + 36b_4^2} \\ N \text{Minimize} &\rightarrow \left\{ -6., \begin{cases} a_1 \rightarrow -0.5, & a_2 \rightarrow 0.333333, \\ a_3 \rightarrow 0.333333, \\ b_1 \rightarrow -0.1, & b_2 \rightarrow 0.0666667, \\ b_3 \rightarrow 0.0666667, & b_4 \rightarrow -0.1 \end{cases} \right\} \Rightarrow E_{gs}/N \geq -3 \end{aligned}$$

The same result was obtained by another method in [4].

We expect that our method outperforms the method of ref. [4] for larger clusters.

Work is underway.



ALGORITHM FOR SIMPLIFICATION OF PRODUCTS OF σ -MATRICES

The algorithm is used, in particular, for simplifying $H\rho$.

→ Input and output form: $(\sigma_i \sigma_j) = d(i, j)$ and $(\sigma_i \sigma_j \sigma_k) = t(i, j, k)$.

Algorithm:

- $d(i, j) \rightarrow \sigma(i, \alpha)\sigma(j, \alpha)$
 - $t(i, j, k) \rightarrow \varepsilon(\alpha, \beta, \gamma)\sigma(i, \alpha)\sigma(j, \beta)\sigma(k, \gamma)$

(we assume that different site indices are different: $i \neq j \neq k \neq i$ etc)
- σ -matrices with different site indices are commuting, so we can stable(!) sort it:
 - first σ in multiplication $\rightarrow \tilde{\sigma}$
 - while found $\tilde{\sigma}\sigma$
 - $\tilde{\sigma}\sigma \rightarrow \tilde{\sigma}[\sigma]$
 - while found $\tilde{\sigma}[\sigma]$
 - if(site indices of $\tilde{\sigma}$ and $[\sigma]$ are equal) push spin index of $[\sigma]$ to the back of $\tilde{\sigma}$ (for example $\tilde{\sigma}(i, \alpha, \beta)[\sigma](i, \gamma) \rightarrow \tilde{\sigma}(i, \alpha, \beta, \gamma)$)
 - else swap $\tilde{\sigma}$ and $[\sigma]$: $\tilde{\sigma}[\sigma] \rightarrow [\sigma]\tilde{\sigma}$
 - if $[\sigma]$ is in first position, $[\sigma] \rightarrow \tilde{\sigma}$
- $\tilde{\sigma}(i, \alpha, \beta, \gamma) = \sigma_i^\alpha \sigma_i^\beta \sigma_i^\gamma$

apply Pauli formula while $\tilde{\sigma}$ has more than one spin index

$$\sigma(i, \alpha, \beta, \gamma, \dots) \rightarrow \delta(\alpha, \beta)\sigma(i, \gamma, \dots) + i\varepsilon(\alpha, \beta, \mu)\sigma(i, \mu, \gamma, \dots)$$

- in each term σ -matrices commute

- simplify δ and ε symbols
- isolate scalar and mixed products $d(i, j)$ and $t(i, j, k)$

This algorithm was implemented in Wolfram Mathematica and Nikhef Form.

ALTERNATIVE BASIC ALGEBRAIC RELATIONS

We can use this formulas recursively to simplify products of scalar and mixed products of σ -matrices.

$$\begin{aligned} (\sigma_1 \sigma_2)^2 &= 3 - 2(\sigma_1 \sigma_2) \\ (\sigma_1 \sigma_2)(\sigma_2 \sigma_3) &= -i(\sigma_1 \sigma_2 \sigma_3) + (\sigma_1 \sigma_3) \\ (\sigma_1 \sigma_2)(\sigma_1 \sigma_2 \sigma_3) &= -(\sigma_1 \sigma_2 \sigma_3) - 2i(\sigma_1 \sigma_3) + 2i(\sigma_2 \sigma_3) \\ (\sigma_1 \sigma_2 \sigma_3)(\sigma_1 \sigma_2) &= -(\sigma_1 \sigma_2 \sigma_3) + 2i(\sigma_1 \sigma_3) - 2i(\sigma_2 \sigma_3) \\ (\sigma_1 \sigma_2)(\sigma_2 \sigma_3 \sigma_4) &= (\sigma_1 \sigma_3 \sigma_4) - i(\sigma_1 \sigma_3)(\sigma_2 \sigma_4) + i(\sigma_1 \sigma_4)(\sigma_2 \sigma_3) \\ (\sigma_2 \sigma_3 \sigma_4)(\sigma_1 \sigma_2) &= (\sigma_1 \sigma_3 \sigma_4) + i(\sigma_1 \sigma_3)(\sigma_2 \sigma_4) - i(\sigma_1 \sigma_4)(\sigma_2 \sigma_3) \\ (\sigma_1 \sigma_2 \sigma_3)^2 &= 6 - 2(\sigma_1 \sigma_2) - 2(\sigma_1 \sigma_3) - 2(\sigma_2 \sigma_3) \\ (\sigma_1 \sigma_2 \sigma_3)(\sigma_1 \sigma_2 \sigma_4) &= +i(\sigma_1 \sigma_3 \sigma_4) + i(\sigma_2 \sigma_3 \sigma_4) \\ &\quad - (\sigma_1 \sigma_3)(\sigma_2 \sigma_4) - (\sigma_1 \sigma_4)(\sigma_2 \sigma_3) + 2(\sigma_3 \sigma_4) \\ (\sigma_1 \sigma_2 \sigma_3)(\sigma_1 \sigma_4 \sigma_5) &= -i(\sigma_1 \sigma_2)(\sigma_3 \sigma_4 \sigma_5) + i(\sigma_1 \sigma_3)(\sigma_2 \sigma_4 \sigma_5) \\ &\quad + (\sigma_2 \sigma_4)(\sigma_3 \sigma_5) - (\sigma_2 \sigma_5)(\sigma_3 \sigma_4) \end{aligned}$$

ALGORITHM FOR GENERATION OF SYMMETRIC ρ

Accounts for symmetries w.r.t. to permutations etc.

→ We get input data as generators of permutation group which rest Hamiltonian.

- generate all elements of this group

- for each number pairs

For each Construction of Pairs (CP) we can get next CP.

We need split all CPs to "groups" of CPs which map into themselves by any permutation of Hamiltonian's group.

We do this by brute force.

- We parameterize each "group" of CPs by new parameter and convert each CP to expression and sum them

This algorithm was implemented in Wolfram Mathematica:

```
rho8 = rhoGen[8, {swapToPerm[{1 -> 2}, 8], swapToPerm[{2 -> 3}, 8], swapToPerm[{1 -> 6, 2 -> 7, 3 -> 8, 4 -> 5}, 8]}]

{d[1, 2] + d[1, 3] + d[2, 3] + d[6, 7] + d[6, 8] + d[7, 8]} a1 + (d[1, 4] + d[2, 4] + d[3, 4] + d[5, 6] + d[5, 7] + d[5, 8]) a2 + ... +
(d[1, 4] d[2, 3] d[5, 8] d[6, 7] + d[1, 3] d[2, 4] d[5, 8] d[6, 7] + d[1, 2] d[3, 4] d[5, 8] d[6, 7] + d[1, 4] d[2, 3] d[5, 7] d[6, 8] +
d[1, 2] d[3, 4] d[5, 7] d[6, 8] + d[1, 4] d[2, 3] d[5, 6] d[7, 8] + d[1, 3] d[2, 4] d[5, 6] d[7, 8] + d[1, 2] d[3, 4] d[5, 6] d[7, 8]) d9,
{a1, a2, a3, a4, b1, b2, b3, b4, b5, b6, b7, b8, b9, b10, b11, b12, b13, b14, b15, b16, b17, b18, b19, b20, b21, b22, b23, b24, b25, b26, b27, b28, b29, b30, b31, b32, b33, b34, b35, b36, b37, b38, b39, b40, b41, b42, b43, b44, b45, b46, b47, b48, b49, b50, b51, b52, b53, b54, b55, b56, b57, b58, b59, b60, b61, b62, b63, b64, b65, b66, b67, b68, b69, b70, b71, b72, b73, b74, b75, b76, b77, b78, b79, b80, b81, b82, b83, b84, b85, b86, b87, b88, b89, b90, b91, b92, b93, b94, b95, b96, b97, b98, b99, b100} d1,
large output show less show more show all set size limit...

solveShredinger[d[1, 4] + d[2, 4] + d[3, 4] + d[4, 5] + d[5, 6] + d[5, 7] + d[5, 8],
, 1 + rho8[[1]], rho8[[2]]] // TableForm

Solve::vars: Equations may not give solutions for all 'solve' variables. >=
1) d2 -> 1/15 (1 + 27 c6) d3 -> 1/15 (-1 + 27 c6) d4 -> 1/15 (1 + 27 c6) d5 -> 1/45 (-1 + 18 c6) d6 -> 1/45 (1 - 63 c6) d7 -> 1/45 (-1 + 18 c6) energ -> -1
d1 -> 0 d2 -> 0 d3 -> 0 d4 -> 1/15 d5 -> 1/15 d6 -> -1/27 d7 -> -1/27 energ -> -3
d1 -> 0 d2 -> 0 d3 -> 0 d4 -> 0 d5 -> 1/15 d6 -> 1/15 d7 -> 1/15 energ -> 3
d1 -> 0 d2 -> 0 d3 -> 0 d4 -> -1/600 d5 -> -1/15 d6 -> -1/15 d7 -> -1/15 energ -> 5
d1 -> 1/540 d2 -> 1/540 d3 -> 1/540 d4 -> 1/540 d5 -> 1/540 d6 -> 1/540 d7 -> 1/540 energ -> 7
d1 -> 0 d2 -> 0 d3 -> 0 d4 -> 0 d5 -> 1/27 sqrt(2) d6 -> 1/27 sqrt(2) d7 -> -1/27 sqrt(2) energ -> -1 + 2 sqrt(2)
d1 -> 0 d2 -> 0 d3 -> 0 d4 -> 0 d5 -> -1/27 sqrt(2) d6 -> -1/27 sqrt(2) d7 -> 1/27 sqrt(2) energ -> -1 + 2 sqrt(2)
d1 -> 0 d2 -> 0 d3 -> 0 d4 -> 0 d5 -> 1/27 (1 - sqrt(3)) d6 -> 1/27 (1 - sqrt(3)) d7 -> 1/27 (1 + sqrt(3)) energ -> -3 + 2 sqrt(3)
d1 -> 0 d2 -> 0 d3 -> 0 d4 -> 0 d5 -> 1/27 (1 + sqrt(3)) d6 -> 1/27 (1 + sqrt(3)) d7 -> 1/27 (1 - sqrt(3)) energ -> -3 + 2 sqrt(3)
d1 -> 1/45 (3 - sqrt(15)) d2 -> 1/45 (3 - sqrt(15)) d3 -> 1/45 (3 + sqrt(15)) d4 -> -2/45 d5 -> 1/15 (-1 + sqrt(15)) d6 -> -1/15 (-1 + sqrt(15)) d7 -> 1/15 (-1 + sqrt(15)) energ -> -3 + 2 sqrt(15)
d1 -> 1/45 (3 + sqrt(15)) d2 -> 1/45 (3 + sqrt(15)) d3 -> 1/45 (3 - sqrt(15)) d4 -> -2/45 d5 -> 1/15 (-1 - sqrt(15)) d6 -> -1/15 (-1 - sqrt(15)) d7 -> 1/15 (-1 - sqrt(15)) energ -> -3 + 2 sqrt(15)
```

REFERENCES

[1] N. Il'in, E. Shpagina, F. Uskov, O. Lychkovskiy Squaring parametrization of constrained and unconstrained sets of quantum states. *arXiv:1704.03861*.

[2] Lychkovskiy, Oleg and Gamayun, Oleksandr and Cheianov, Vadim Time Scale for Adiabaticity Breakdown in Driven Many-Body Systems and Orthogonality Catastrophe *Phys. Rev. Lett.* 119,200401 (2017).

[3] David A. Mazziotti Advances in Chemical Physics, Reduced-Density-Matrix Mechanics: With Application to Many-Electron Atoms and Molecules *Volume 134. Wiley-Interscience, 1 edition., 2007.*

[4] R. Tarrah and R. Valenti Exact lower bounds to the ground-state energy of spin systems: The two-dimensional $S = \frac{1}{2}$ antiferromagnetic Heisenberg model *Physical review B*, 1990.