Title: Deep learning for quantum many-body physics or: Toolmaking beyond the papyrus complexity

Speakers: Giuseppe Carleo

Collection: Machine Learning for Quantum Design

Date: July 08, 2019 - 9:30 AM

URL: http://pirsa.org/19070003

Abstract: In this talk I will discuss some of the long-term challenges emerging with the effort of making deep learning a relevant tool for controlled scientific discovery in many-body quantum physics.   The current state of the art of deep neural quantum states and learning tools will be discussed in connection with open challenging problems in condensed matter physics, including frustrated magnetism and quantum dynamics.

Perimeter Institute

July 8 2019

Deep Learning For Quantum: The Good, The Bad, and The Ugly

Giuseppe Carleo

Center for Computational Quantum Physics
New York

FLATIRON INSTITUTE
SIMONS FOUNDATION

The Golden Age Of Deep Learning

The Golden Age Of Deep Learning

Image/Speech Recognition

2012
Beat All Other Image Classification Methods

2014
~Human in Face Classification

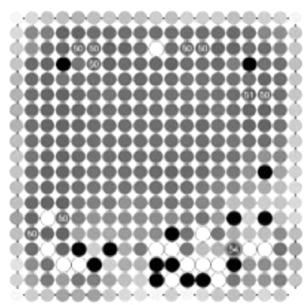
Language Translation

I can even speak Japanese.

私は日本語を話すことすらできます。

2016
~Human Translator on Isolated Sentences

Game Playing



2016
Superhuman in Go

2019
~Human in Strategy Game

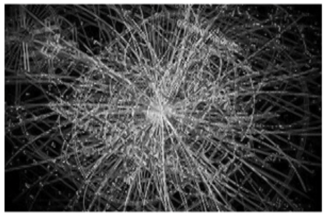
Marked By Many Achievements

Quantifiable Improvement
Over Existing Methods

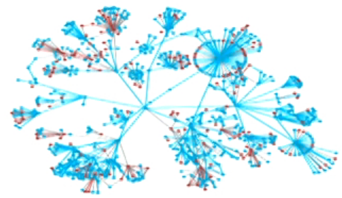
Enabled Genuinely
New Applications

Flexible Methods, Can
Work In Disparate Domains

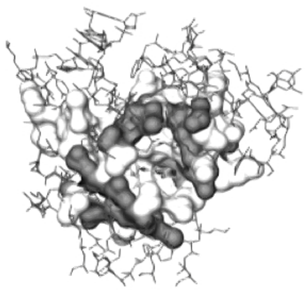
The (Iron?) Age of ML in Physics



Particle Physics



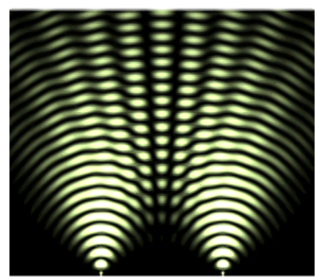
Statistical Physics



Chemistry/ Materials

Astrophysics

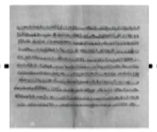
Carleo, Cirac, Cranmer, Daudet, Schuld, Tishby, Vogt, and Zdeborovà arXiv:1903.10563 (2019)



Quantum Physics

Many-Body Quantum Systems

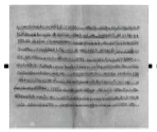
The Quantum Many-Body Problem

[3000 BCE]	[1455]	[1973]	[1993]	[2002]	[2019]
Papyrus	**Book**	**IBM 3340**	**IBM 3390**	**Earth Simulator**	**Summit**
					
10 Qubits	15 Qubits	23 Qubits	35 Qubits	46 Qubits	54 Qubits


The Quantum Many-Body Problem
[3000 BCE]
Papyrus
10 Qubits
[1455]
Book
15 Qubits
[1973]
IBM 3340
23 Qubits
[1993]
IBM 3390
35 Qubits
[2002]
Earth Simulator
46 Qubits
[2019]
Summit
54 Qubits
PERIMETER
PI
INSTITU

The Quantum Many-Body Problem

[3000 BCE] **Papyrus**



10 Qubits

[1455] **Book**

15 Qubits

[1973] **IBM 3340**

23 Qubits

[1993] **IBM 3390**

35 Qubits

[2002] **Earth Simulator**

46 Qubits

[2019] **Summit**

54 Qubits

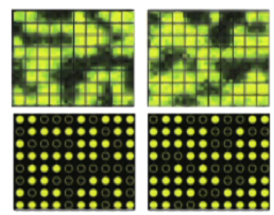
Exponentially hard problems

$$\mathcal{H}\,|\Psi\rangle = E\,|\Psi\rangle$$

$$\frac{d}{dt}\,|\Psi(t)\rangle = -i\mathcal{H}\,|\Psi(t)\rangle$$

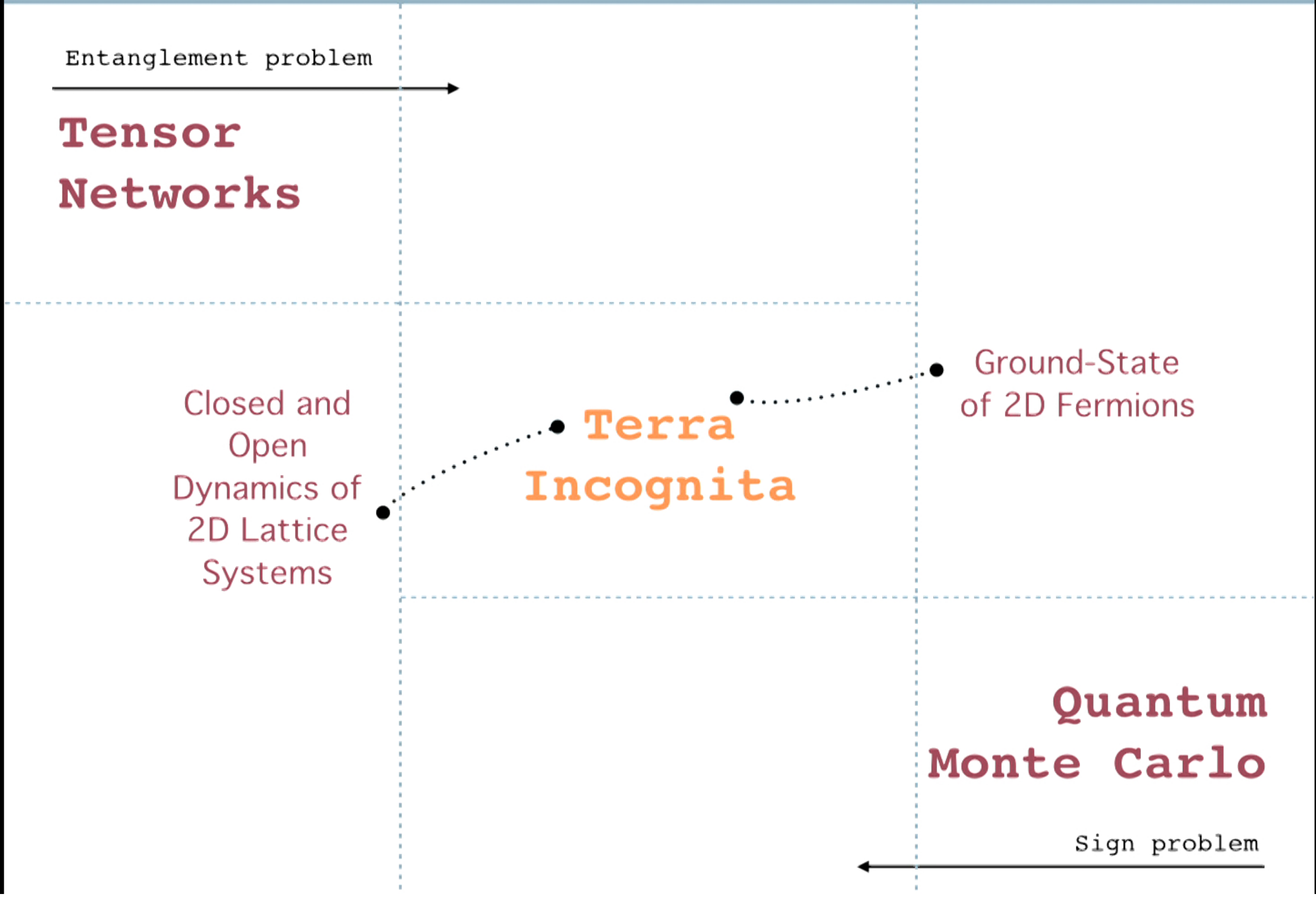
$$\frac{d}{dt}\rho(t) = L\rho(t)$$

Classical Simulation



Characterize Quantum Computers/Hardware

Challenges for state-of-the-art methods



Challenges for state-of-the-art methods

Entanglement problem

Tensor Networks

Simulation of Large Quantum Circuits

Exponential Scaling

Tensor Contractions

Closed and Open Dynamics of 2D Lattice Systems

Terra Incognita

Ground-State of 2D Fermions

Full Tomography

Exponential Scaling

Benchmark Large Quantum Hardware

Quantum Monte Carlo

Sign problem

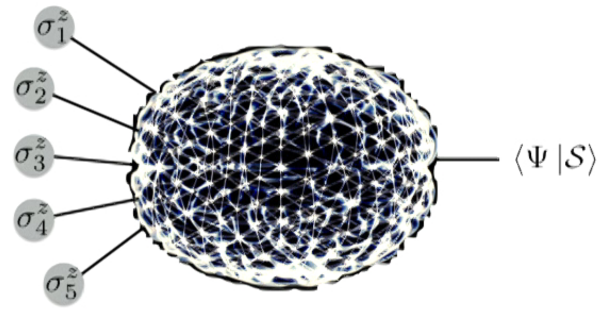
Is Machine Learning Useful?

Only Upon Showing That
ML Can Do Better/ Do More
Than Other Many-Body Approaches
We Can Enter a (Long-Lasting)
Golden Age
of ML for
Quantum Physics

The
State
Of
Things

Neural-Network Quantum States

Carleo, and Troyer
Science 355, 602 (2017)



Many-Body Amplitudes

$$\Psi(s_1 \ldots s_N) = g^{(L)} \circ W^{(L)} \ldots g^{(2)} \circ W^{(2)} g^{(1)} \circ W^{(1)} \mathbf{s}$$

Quantum Numbers

Nonlinear Activation Functions Applied Component-Wise

Variational Parameters

(complex-valued here)

Learning Framework

$$\mathcal{L}(\mathbf{W}) = \frac{\sum_{\mathbf{s}} P(\mathbf{s}, \mathbf{W}) L(\mathbf{s}, \mathbf{W})}{\sum_{\mathbf{s}} P(\mathbf{s}, \mathbf{W})}$$

Loss Function

Probability Density

Learning Framework

$$\mathcal{L}(\mathbf{W}) = \frac{\sum_{\mathbf{s}} P(\mathbf{s},\mathbf{W}) L(\mathbf{s},\mathbf{W})}{\sum_{\mathbf{s}} P(\mathbf{s},\mathbf{W})}$$

Loss Function

Probability Density

Simulation Driven

$$P(\mathbf{s},\mathbf{W}) = |\Psi(\mathbf{s},\mathbf{W})|^2$$

No External Data

Data Driven

$$P(\mathbf{s})$$

Stationary Probability
Defined by Data

Applications

Simulate Quantum Systems

Problem Driven
(Self learning)

Characterize Quantum Hardware

Data Driven
(Experimental Measurements)

Ground-State Search

Example Hard Problem #1

Variational Formulation

$$E(\mathbf{W}) = \frac{\langle \Psi(\mathbf{W}) | \mathcal{H} | \Psi(\mathbf{W}) \rangle}{\langle \Psi(\mathbf{W}) | \Psi(\mathbf{W}) \rangle} \geq E_0$$

Exact Ground-
State Energy

Variational Formulation

$$E(\mathbf{W}) = \frac{\langle \Psi(\mathbf{W}) | \mathcal{H} | \Psi(\mathbf{W}) \rangle}{\langle \Psi(\mathbf{W}) | \Psi(\mathbf{W}) \rangle} \geq E_0$$

Exact Ground-
State Energy

Expectation Minimization

$$\mathcal{L}(\mathbf{W}) = \frac{\sum_s |\Psi(s, W)|^2 E_{\text{loc}}(s, W)}{\sum_s |\Psi(s, W)|^2}$$

McMillan
Phys. Rev. 138,
A442 (1965)

Learning Framework

$$\mathcal{L}(\mathbf{W}) = \frac{\sum_{\mathbf{s}} P(\mathbf{s}, \mathbf{W}) L(\mathbf{s}, \mathbf{W})}{\sum_{\mathbf{s}} P(\mathbf{s}, \mathbf{W})}$$

Loss Function

Probability Density

Learning Framework

$$\mathcal{L}(\mathbf{W}) = \frac{\sum_{\mathbf{s}} P(\mathbf{s}, \mathbf{W}) L(\mathbf{s}, \mathbf{W})}{\sum_{\mathbf{s}} P(\mathbf{s}, \mathbf{W})}$$

Loss Function

Probability Density

Simulation Driven

$$P(\mathbf{s}, \mathbf{W}) = |\Psi(\mathbf{s}, \mathbf{W})|^2$$

No External Data

Data Driven

$$P(\mathbf{s})$$

Stationary Probability
Defined by Data

Key Issues

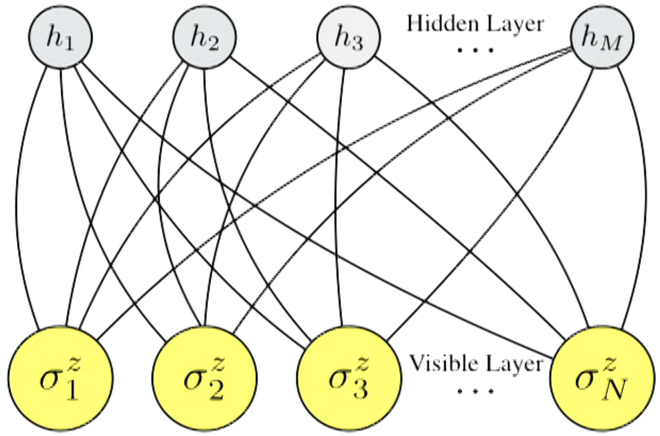
Variational Learning Is Significantly Harder Than Data-Driven Applications

Specific Requirements In Terms of High Accuracy/ Complex Networks, etc

“Legacy” Solution

Restricted Boltzmann Machines

“Spherical Chicken in Vacuum” [L. Wang]



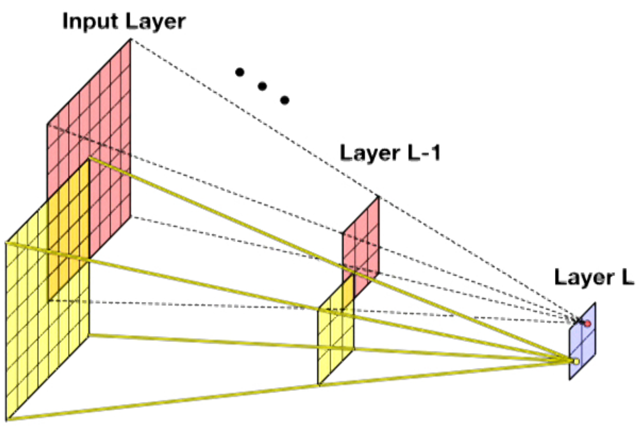
Carleo, and Troyer
Science 355, 602 (2017)

Melko, Carleo, Carrasquilla, and Cirac
Nat. Phys., Perspective (2019)

$$g^{(1)}(x) = \log\cosh(x)$$

$$g^{(2)}(x) = \exp(x)$$

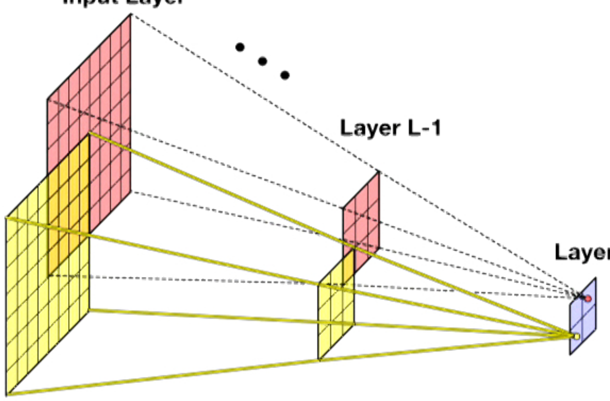
But Good Arguments For "Animals In The Wild"



Deep Convolutional Networks Are Polynomially More Efficient Than Shallow Ones

Levine, Sharir, Cohen, and Shashua
Physical Review Letters *122, 065301 (2019)*

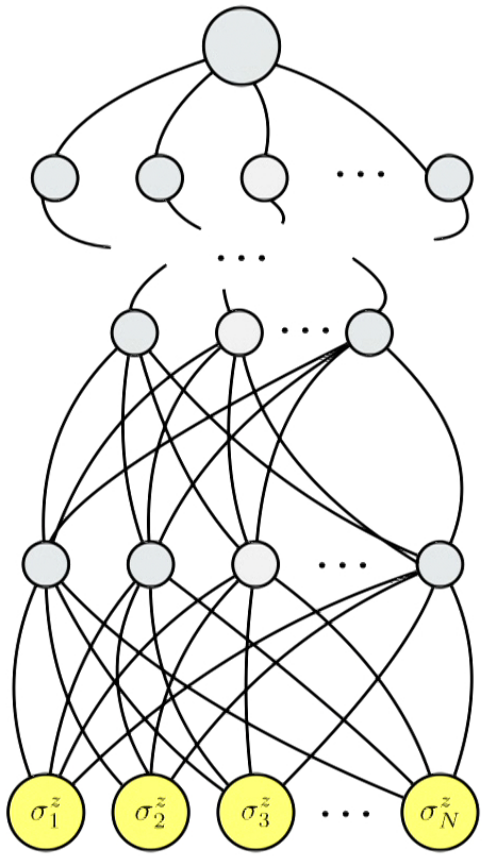
But Good Arguments For "Animals In The Wild"



Deep Convolutional Networks Are Polynomially More Efficient Than Shallow Ones

Levine, Sharir, Cohen, and Shashua
Physical Review Letters *122, 065301 (2019)*

Deeper NQS



Saito, and Kato
J. Phys. Soc. Jpn. 87, 014001 (2018)

Choo, Carleo, Regnault, and Neupert
Phys. Rev. Lett. 121, 167204 (2018)

Liang, et al.
Phys. Rev. B 98, 104426 (2018)

Kochkov, and Clark
arXiv:1811.12423 (2018)

Choo, Neupert, and Carleo
arXiv:1903.06713 (2019)

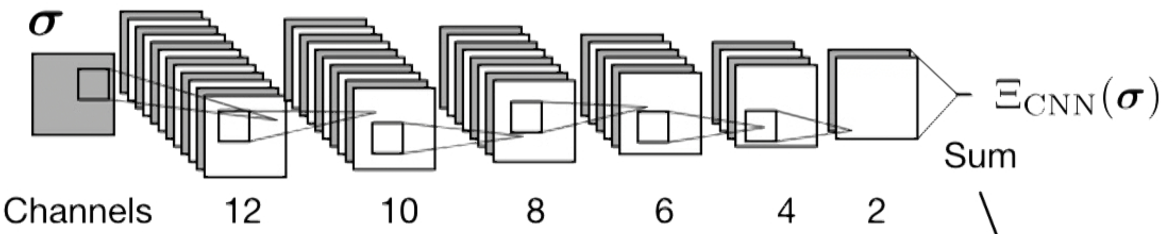
Sharir, et al.
arXiv:1902.04057 (2019)

...

Convolutional Quantum States

Choo, Neupert, and Carleo
arXiv:1903.06713 (2019)

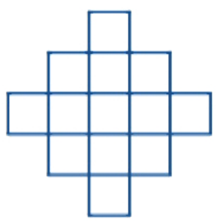
6 layer deep fully convolutional network:



$$\Psi_{\text{CNN}}(\boldsymbol{\sigma}) = \exp(\Xi_{\text{CNN}}(\boldsymbol{\sigma}))$$

The sum output layer makes the ansatz translationally invariant

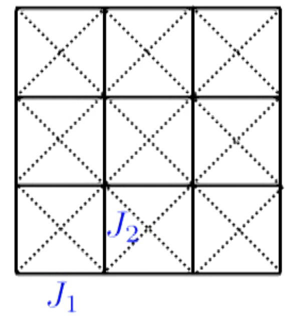
Filter Shape:



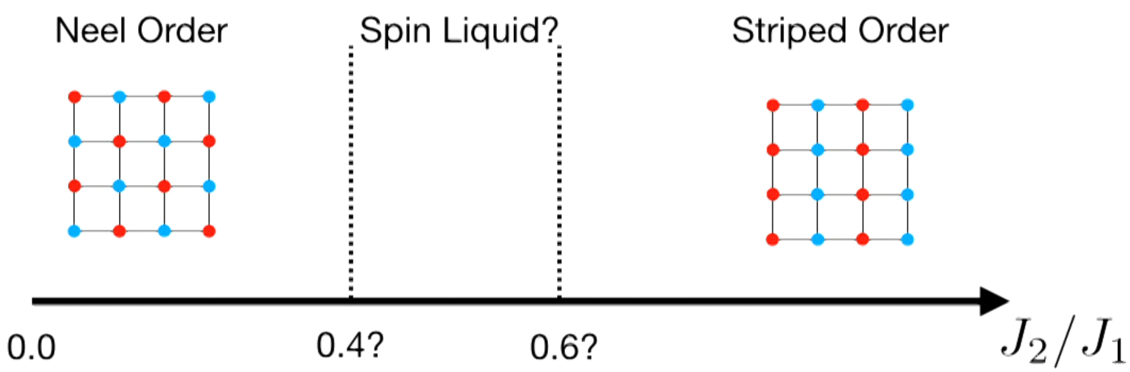
Frustrated 2D Spins

J1-J2 Model

$$\hat{H} = J_1 \sum_{\langle ij \rangle} \hat{\boldsymbol{S}}_i \cdot \hat{\boldsymbol{S}}_j + J_2 \sum_{\langle\langle ij \rangle\rangle} \hat{\boldsymbol{S}}_i \cdot \hat{\boldsymbol{S}}_j$$

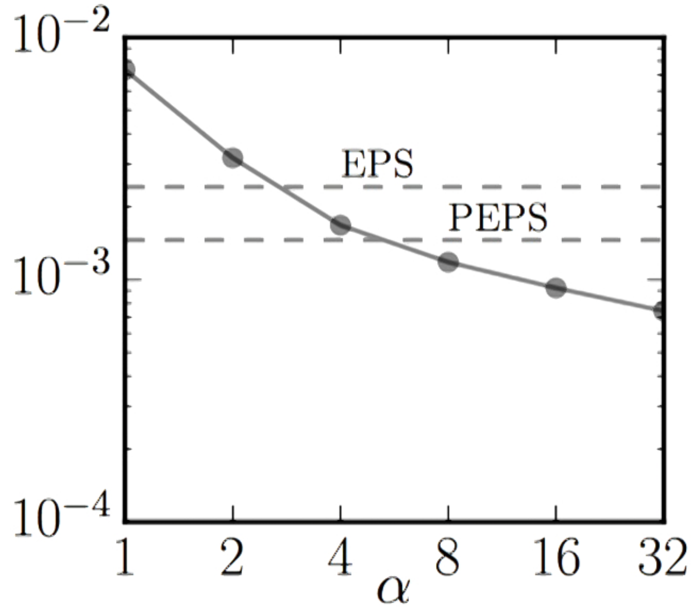


Phase Diagram



Heisenberg Limit J2=0

Carleo, and Troyer
Science 355, 602 (2017)

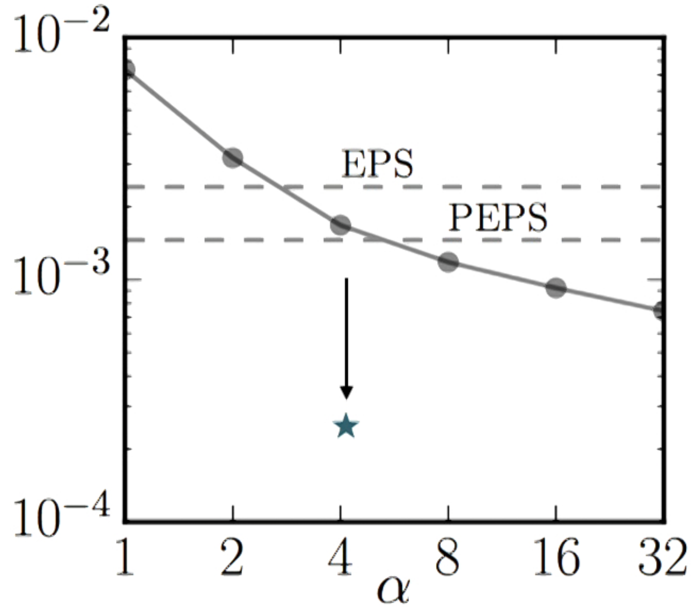


10 by 10 cluster

Early (2016) Results With
Shallow (RBM) Network

Heisenberg Limit J2=0

Choo, Neupert, and Carleo
arXiv:1903.06713 (2019)

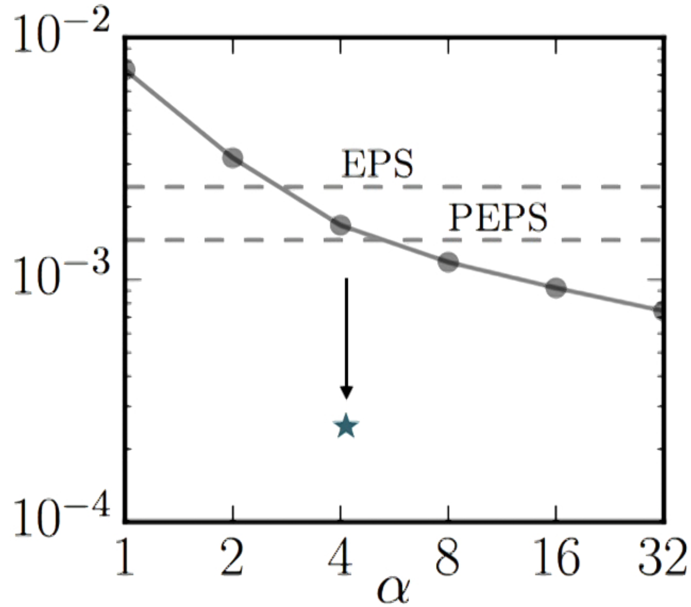


10 by 10 cluster

(Mildly) deep network further improves

Heisenberg Limit J2=0

Sharir, Levine, Wies, Carleo, and Shashua
arXiv:1902.04057 (2019)



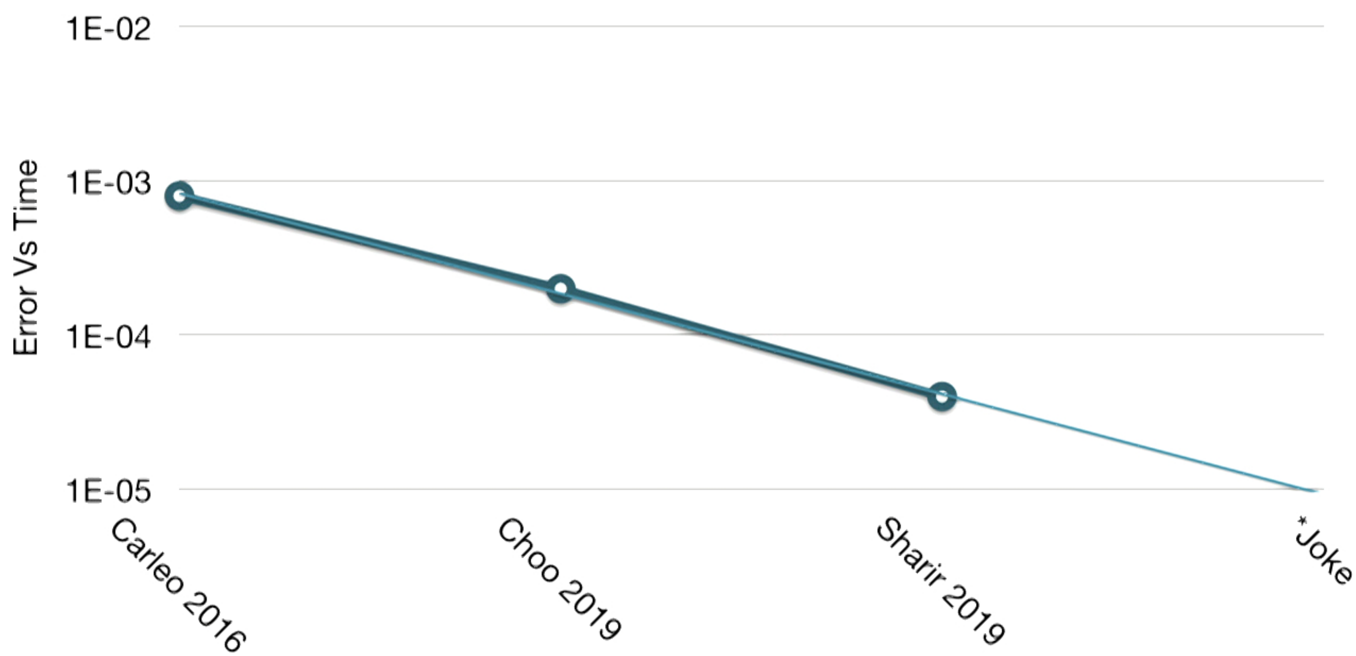
10 by 10 cluster

(Actual) deep network further improves

~4x10^-5

“Carleo’s Law” For ML Supremacy*

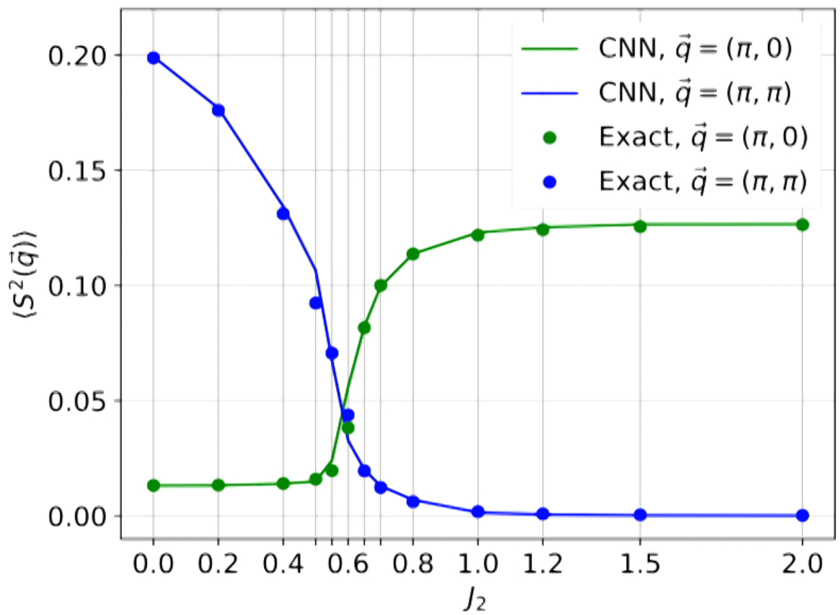
Neural-Network Quantum States Are Improving At A (Doubly!?!) Exponential Rate



Comparison with ED

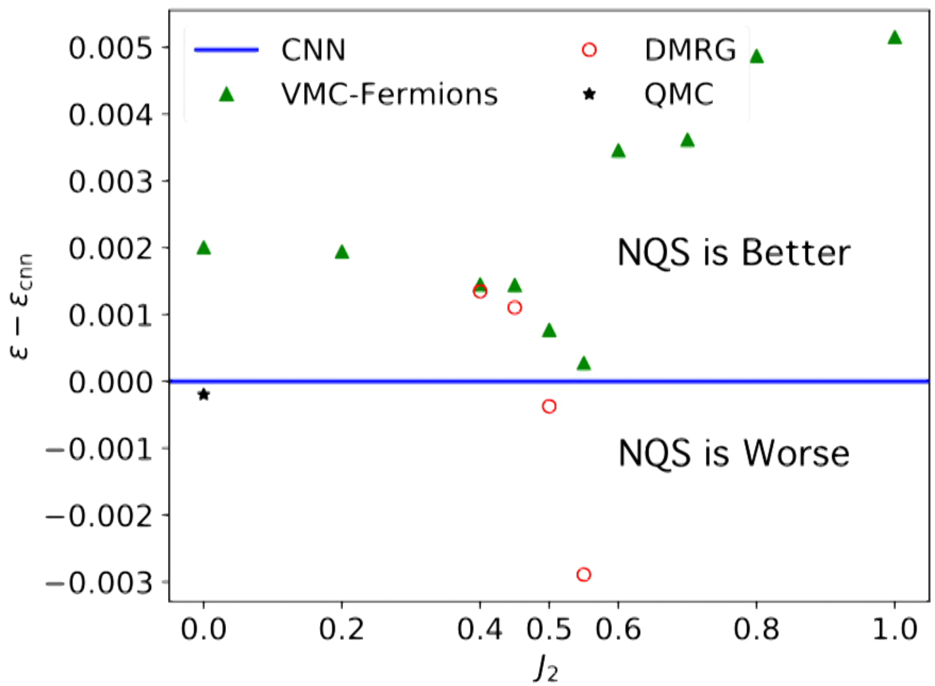
Spin-spin correlation structure factor

$$S^2(\vec{q}) = \frac{1}{N(N+2)} \sum_{i,j} \langle \hat{\boldsymbol{S}}_i \cdot \hat{\boldsymbol{S}}_j \rangle e^{i\vec{q}\cdot(\vec{r}_i - \vec{r}_j)}$$



H.J. Schulz et al, J. Phys. I France 6, 675-703 (1996)

Comparison with Other Methods



10 by 10 cluster

Choo, Neupert, and Carleo
arXiv:1903.06713 (2019)

Tangible Improvements on Open Challenging Models
(~20 Years of Numerical Studies)

Bridging The Gap With Deep Learning Applications?

Autoregressive Quantum States

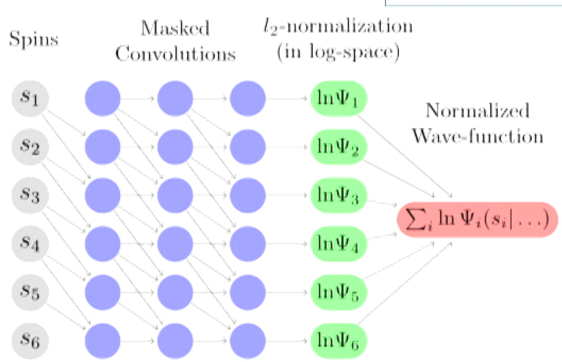
Sharir, Levine, Wies, Carleo, and Shashua
arXiv:1902.04057 (2019)

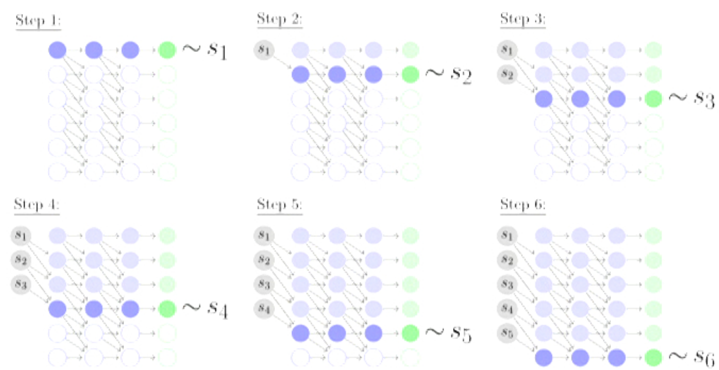
$$\Psi(s_1, \ldots, s_N) = \prod_{i=1}^{N} \psi_i(s_i | s_{i-1}, \ldots, s_1)$$

$$\sum_{s'} |\psi_i(s' | s_{i-1}, \ldots, s_1)|^2 = 1$$

Normalized "Conditionals"

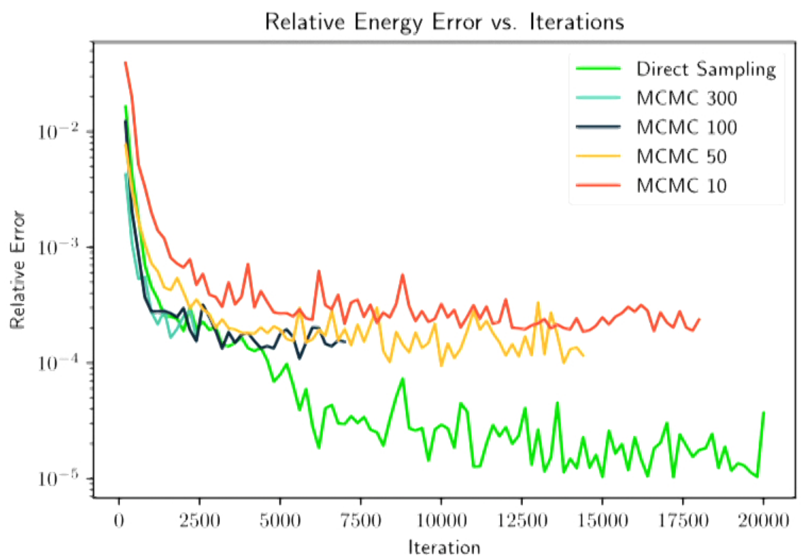
Exact Sampling: no Metropolis, no Markov Chains, no autocorrelation times…





Removing The Sampling Bottleneck Pays Off

Sharir, Levine, Wies, Carleo, and Shashua
arXiv:1902.04057 (2019)



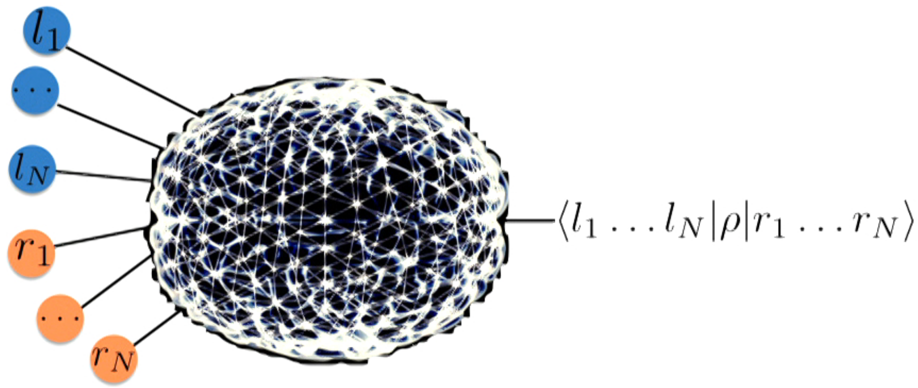
21x21 Transverse-Field Ising in 2d

Depth 20
About 1 Million Parameters

Open
Many-Body
Quantum
Systems

Example
Hard Problem
#2

Neural-Network Density Matrix



Density Matrix

$$\rho(l_1 \dots l_N, r_1 \dots r_N) = g^{(L)} \circ W^{(L)} \dots g^{(2)} \circ W^{(2)} g^{(1)} \circ W^{(1)} \mathbf{s}$$

Can be made hermitian
No guarantee to be pos. def.

Purification

$$\rho(\vec{l},\vec{r}) = \sum_{h_1 \ldots h_{\tilde{M}}} \Psi(\vec{l},\vec{h})\Psi(\vec{r},\vec{h})^\star$$

Ancillary variables

Hermitian
Positive semi-def.

Explicit Form For Shallow Networks

Torlai and Melko
Phys. Rev. Lett. 120, 240503 (2018)

$$\rho_{\vec{l},\vec{r}} = \exp\left[\sum_{j=1}^{N}(a_j l_j + a_j^* r_j)\right] \times \prod_{k=1}^{M} \mathcal{X}_k \times \prod_{p=1}^{\tilde{M}} \mathcal{Y}_p$$

$$\mathcal{X}_k = \cosh\left(b_k + \sum_{j=1}^{N} W_{k,j} l_j\right) \cosh\left(b_k^* + \sum_{j=1}^{N} W_{k,j}^* r_j\right)$$

$$\mathcal{Y}_p = \cosh\left(c_p + c_p^* + \sum_{j=1}^{N}(U_{p,j} l_j + U_{p,j}^* r_j)\right)$$

Hartmann and Carleo
Phys. Rev. Lett. 122, 250502 (2019)

(III) Lindblad Dynamics

Variational Formulation

$$\dot{\rho} = -i[H,\rho] + \sum_j \frac{\gamma_j}{2}\left(2c_j\rho c_j^\dagger - c_j^\dagger c_j \rho - \rho c_j^\dagger c_j\right)$$

Variational Time Evolution

$$\partial_t \vec{\rho} = \sum_k \dot{\alpha}_k O_k \vec{\rho}$$

$$[O_k]_{\vec{l},\vec{r};\vec{l},\vec{r}} = \partial \ln(\rho_{\vec{l},\vec{r}})/(\partial \alpha_k)$$

Variational Operators

Exact Time Evolution

$$\partial_t \vec{\rho} = \mathcal{L}\vec{\rho}$$

Variational Formulation

$$\delta = \left\| \sum_k \dot{\alpha}_k O_k \vec{\rho} - \mathcal{L}\vec{\rho} \right\|_2^2$$

Minimize L2 Distance

$$\sum_p S_{k,p}\, \dot{\alpha}_p = f_k$$

Explicit Equations of Motion

Hartmann and Carleo
Phys. Rev. Lett.
122, 250502 (2019)

Yoshioka and Hamazaki
Phys. Rev. B 99,
214306 (2019)

Nagy and Savona
Phys. Rev. Lett.
122, 250501 (2019)

Vicentini, et al.
Phys. Rev. Lett. 122,
250503 (2019)

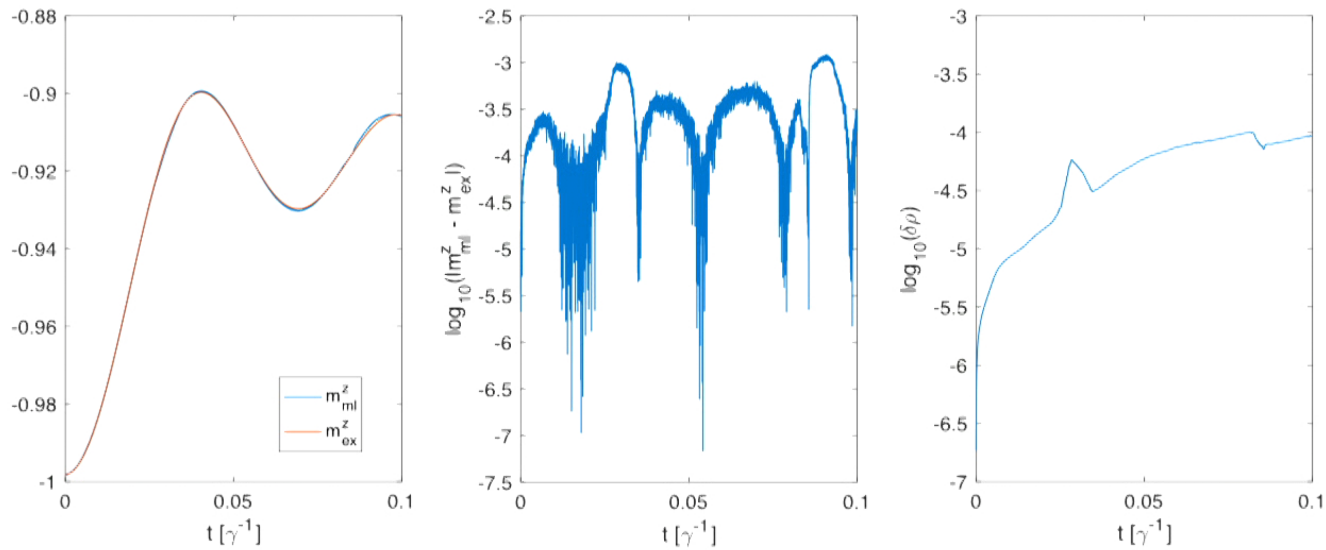
Example Application

Dissipative Anisotropic Heisenberg

$$H = \sum_{j=1}^{N} B\sigma_j^z + \sum_{<j,l>} \sum_{a=x,y,z} J_a \sigma_j^a \sigma_l^a$$

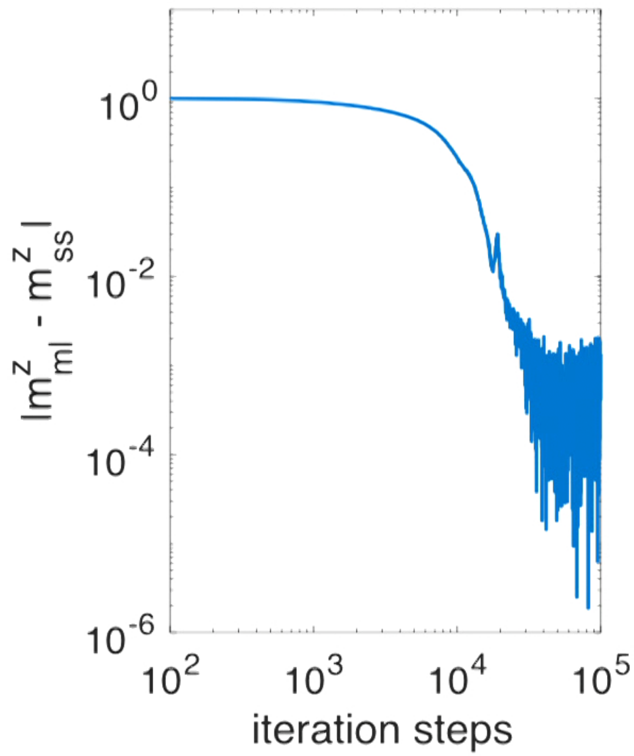
$$\mathcal{D}[\rho] = \frac{\gamma}{2} \sum_{j=1}^{N} \left(2\sigma_j^- \rho \sigma_j^+ - \sigma_j^+ \sigma_j^- \rho - \rho \sigma_j^+ \sigma_j^-\right)$$

Small Chain With PBC



$$B = 10\gamma,\ J_x = 20\gamma,\ J_y = 0 \text{ and } J_z = 10\gamma$$

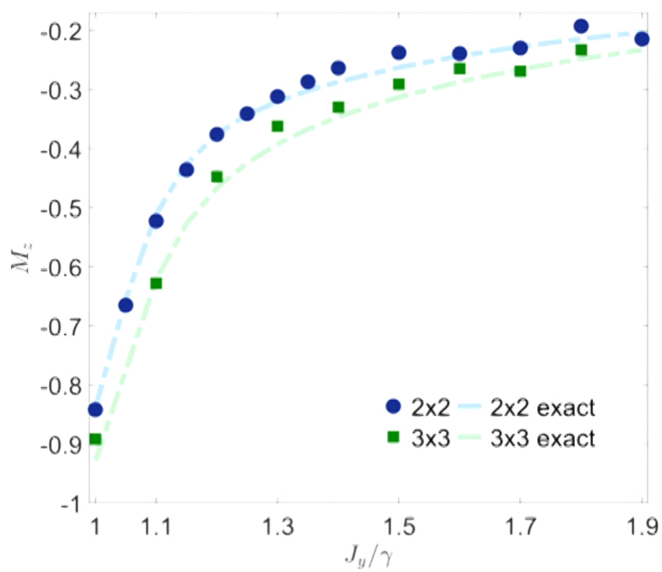
Larger Chain

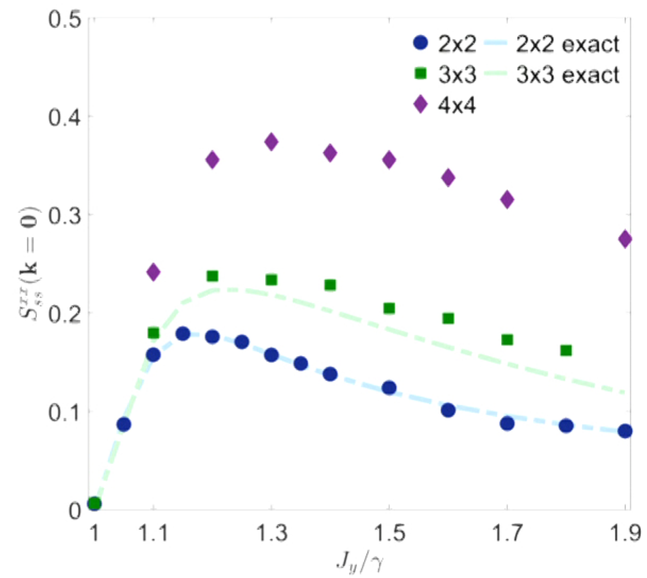


$$B = 10\gamma,\ J_x = \gamma,\ J_y = 0 \text{ and } J_z = 0$$

Two-Dimensional Examples

Nagy and Savona
arXiv:1902.09483 (2019)





$$B = 0,\ J_x/\gamma = 0.9,\ J_z/\gamma = 1.0$$

Software

The NetKet Project

www.netket.org

NetKet: A Machine Learning Toolkit for Many-Body Quantum Systems

Giuseppe Carleo,[1] Kenny Choo,[2] Damian Hofmann,[3] James E. T. Smith,[4] Tom Westerhout,[5] Fabien Alet,[6] Emily J. Davis,[7] Stavros Efthymiou,[8] Ivan Glasser,[8] Sheng-Hsuan Lin,[9] Marta Mauri,[1,10] Guglielmo Mazzola,[11] Christian B. Mendl,[12] Evert van Nieuwenburg,[13] Ossian O'Reilly,[14] Hugo Théveniaut,[6] Giacomo Torlai,[1] and Alexander Wietek[1]

[1] *Center for Computational Quantum Physics, Flatiron Institute, 162 5th Avenue, NY 10010, New York, USA*
[2] *Department of Physics, University of Zurich, Winterthurerstrasse 190, 8057 Zürich, Switzerland*
[3] *Max Planck Institute for the Structure and Dynamics of Matter, Luruper Chaussee 149, 22761 Hamburg, Germany*
[4] *Department of Chemistry, University of Colorado Boulder, Boulder, Colorado 80302, USA*
[5] *Institute for Molecules and Materials, Radboud University, NL-6525 AJ Nijmegen, The Netherlands*
[6] *Laboratoire de Physique Théorique, IRSAMC, Université de Toulouse, CNRS, UPS, 31062 Toulouse, France*
[7] *Department of Physics, Stanford University, Stanford, California 94305, USA*
[8] *Max-Planck-Institut für Quantenoptik, Hans-Kopfermann-Straße 1, 85748 Garching bei München, Germany*
[9] *Department of Physics, T42, Technische Universität München, James-Franck-Straße 1, 85748 Garching bei München, Germany*
[10] *Dipartimento di Fisica, Università degli Studi di Milano, via Celoria 16, I-20133 Milano, Italy*
[11] *Theoretische Physik, ETH Zürich, 8093 Zürich, Switzerland*
[12] *Technische Universität Dresden, Institute of Scientific Computing, Zellescher Weg 12-14, 01069 Dresden, Germany*
[13] *Institute for Quantum Information and Matter, California Institute of Technology, Pasadena, CA 91125, USA*
[14] *Southern California Earthquake Center, University of Southern California, 3651 Trousdale Pkwy, Los Angeles, CA 90089, USA*

arXiv: 1904:00031 (2019)

```python
import netket as nk

# 1D Lattice
g = nk.graph.Hypercube(length=20, n_dim=1, pbc=True)

# Hilbert space of spins on the graph
hi = nk.hilbert.Spin(s=0.5, graph=g)

# Ising spin hamiltonian
ha = nk.operator.Ising(h=1.0, hilbert=hi)

# RBM Spin Machine
ma = nk.machine.RbmSpin(alpha=1, hilbert=hi)
ma.init_random_parameters(seed=1234, sigma=0.01)

# Metropolis Local Sampling
sa = nk.sampler.MetropolisLocal(machine=ma)

# Optimizer
op = nk.optimizer.Sgd(learning_rate=0.1)

# Stochastic reconfiguration
gs = nk.variational.Vmc(
    hamiltonian=ha,
    sampler=sa,
    optimizer=op,
    n_samples=1000,
    diag_shift=0.1,
    method='Sr')

gs.run(output_prefix='test', n_iter=300)
```

The Ugly

Still missing a meaningful measure of classical simulation complexity beyond entanglement entropy

Only A Few Works Have Focused on Fermions

The Point at J2/J1=0.5


PERIMETER
PI
INSTITUTE

The Bad

Only Some Symmetries Can Be Efficiently Imposed At The Moment

For open systems, scale to simulation of larger systems in two dimensions

The Good

Neural Network Quantum States Can Already Outperform Existing Techniques In Selected Challenging Applications

Fast Progress Seen When Closing The Gap With State-of-the-art Deep Learning

Much closer to a standardization of tools and software