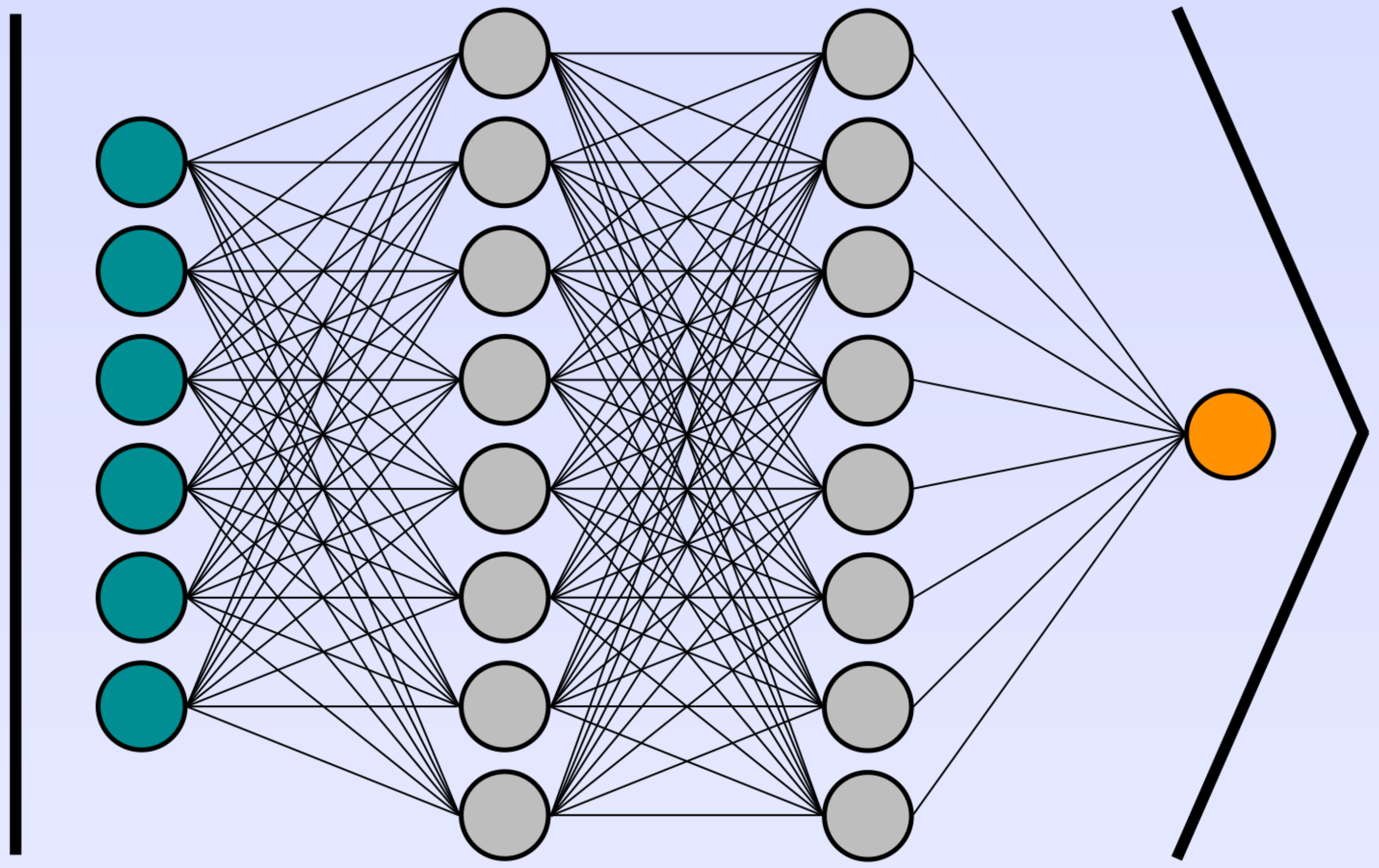


Deep Learning Pairing Correlations from Neural-Network Quantum States

Jane Kim
MSU/FRIB

ISNET-9
24 May 2023

Collaborators

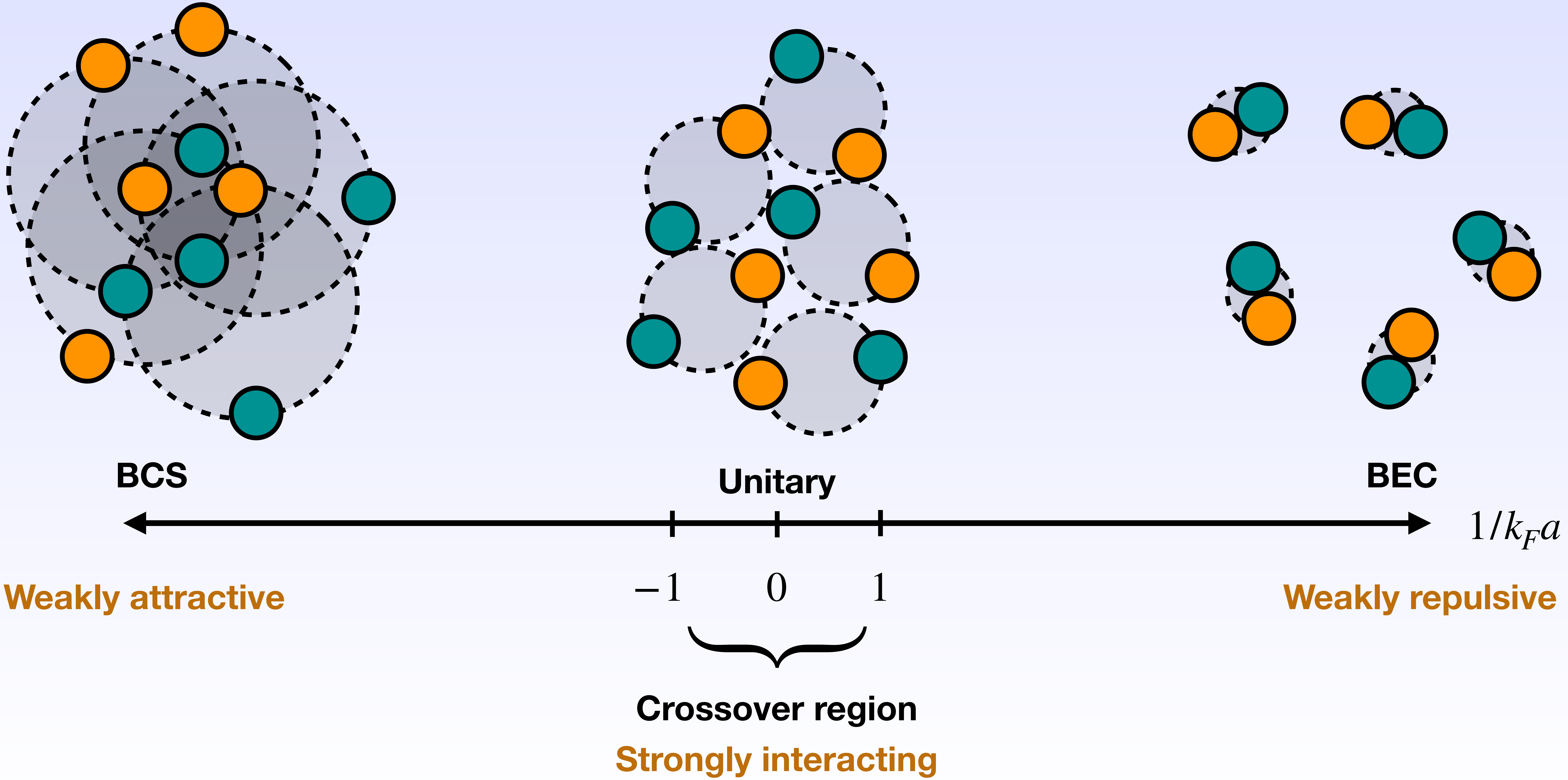
- Argonne National Laboratory: **Alessandro Lovato**, **Bryce Fore**
- École Polytechnique Fédérale de Lausanne: **Giuseppe Carleo**, **Gabriel Pescia**, **Jannes Nys**
- Los Alamos National Laboratory: **Stefano Gandolfi**
- Michigan State University / Facility for Rare Isotope Beams: **Morten Hjorth-Jensen**

- See our recent preprint: **"Neural-network quantum states for ultra-cold Fermi gases"** **arXiv:2305.08831**

Ultra-cold Fermi gases

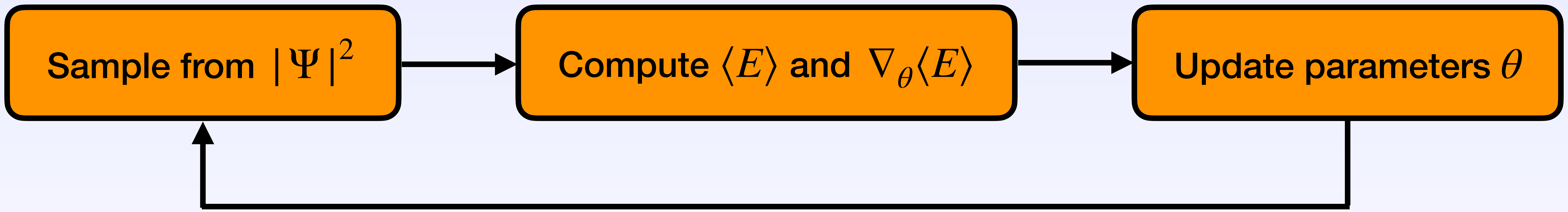
- Highly non-perturbative, short-range, attractive interaction
- Two component fermions (spin-up & spin-down, heavy & light, etc.)
- Dilute $\rightarrow$ mostly s-wave
  $\rightarrow$ behavior mainly governed by s-wave scattering length a and effective range r_e
- Can be created in the laboratory with variable scattering length a
  - $a < 0$ BCS regime of long-range Cooper pairs
  - $a > 0$ BEC regime of tightly-bound dimers
  - $|a| \rightarrow \infty$ Unitary limit (universal)
- Relevant for understanding superfluidity in fermionic systems, dilute neutron star matter, and development of many-body methods

The BCS-BEC Crossover



Our Approach

- Simulate unpolarized gas with N fermions in a periodic box of side length L
- Design a neural-network quantum state (NQS) that efficiently captures pairing and backflow correlations, while enforcing symmetries and boundary conditions
- Train NQS using variational Monte Carlo (VMC) method, i.e. minimize the energy $\langle E \rangle$



- Compare to state-of-the-art Diffusion Monte Carlo (DMC) calculations
- Compare to similar NQS based on Slater determinants, with and without backflow

Fermionic Wave Functions

- Wave function must be antisymmetric w.r.t. particle exchange

$$\Psi(X) = e^{J(X)}\Phi(X)$$

antisymmetric — symmetric — antisymmetric

$$X = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$$

$$\mathbf{x}_i = (\mathbf{r}_i, s_i^z)$$

- Typically use a Slater Determinant of single-particle pairing orbitals $\Phi(X) = \det\left[\phi_\alpha(\mathbf{x}_i)\right]$

- For ultra-cold Fermi gases, state-of-the-art calculations use a Slater determinant of spin-singlet pairing orbitals (aka geminal or BCS wave function) $\Phi(X) = \det\left[\phi(\mathbf{x}_{i\uparrow}, \mathbf{x}_{j\downarrow})\right]$

- However, BCS wave function relies on fixing the spins (not applicable for nuclear systems)

Pfaffian Wave Function

- Most general way to construct an antisymmetric wave function from a pairing orbital

$$\Phi(X) = \mathrm{pf}\left[\phi(\mathbf{x}_i, \mathbf{x}_j)\right]$$

  where ϕ must be antisymmetric.

- For our NQS, we define

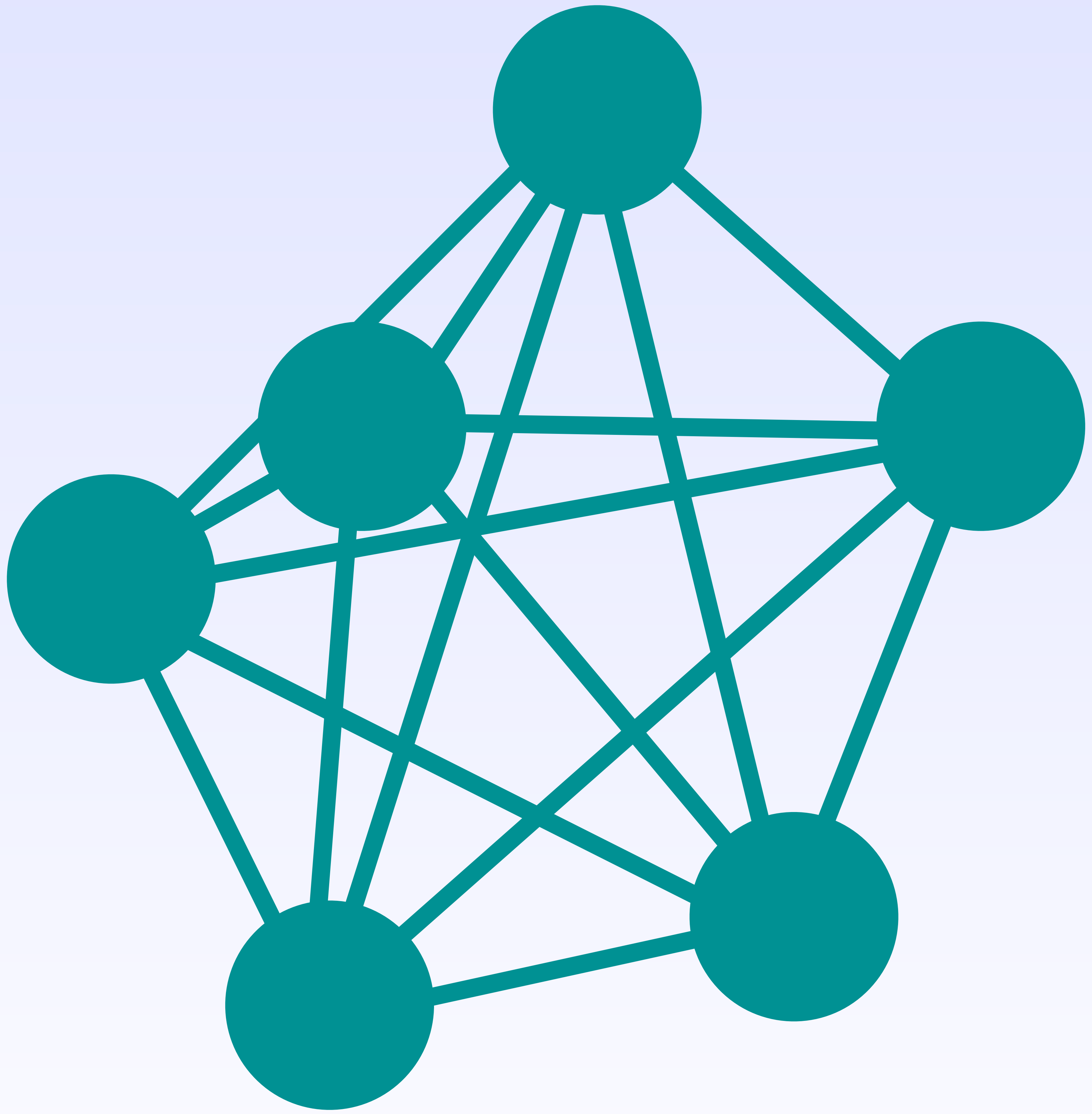
$$\phi(\mathbf{x}_i, \mathbf{x}_j) \equiv \nu(\mathbf{x}_i, \mathbf{x}_j) - \nu(\mathbf{x}_j, \mathbf{x}_i)$$

  where ν is a neural network.

- Systematically improvable (universal approximation theorem)

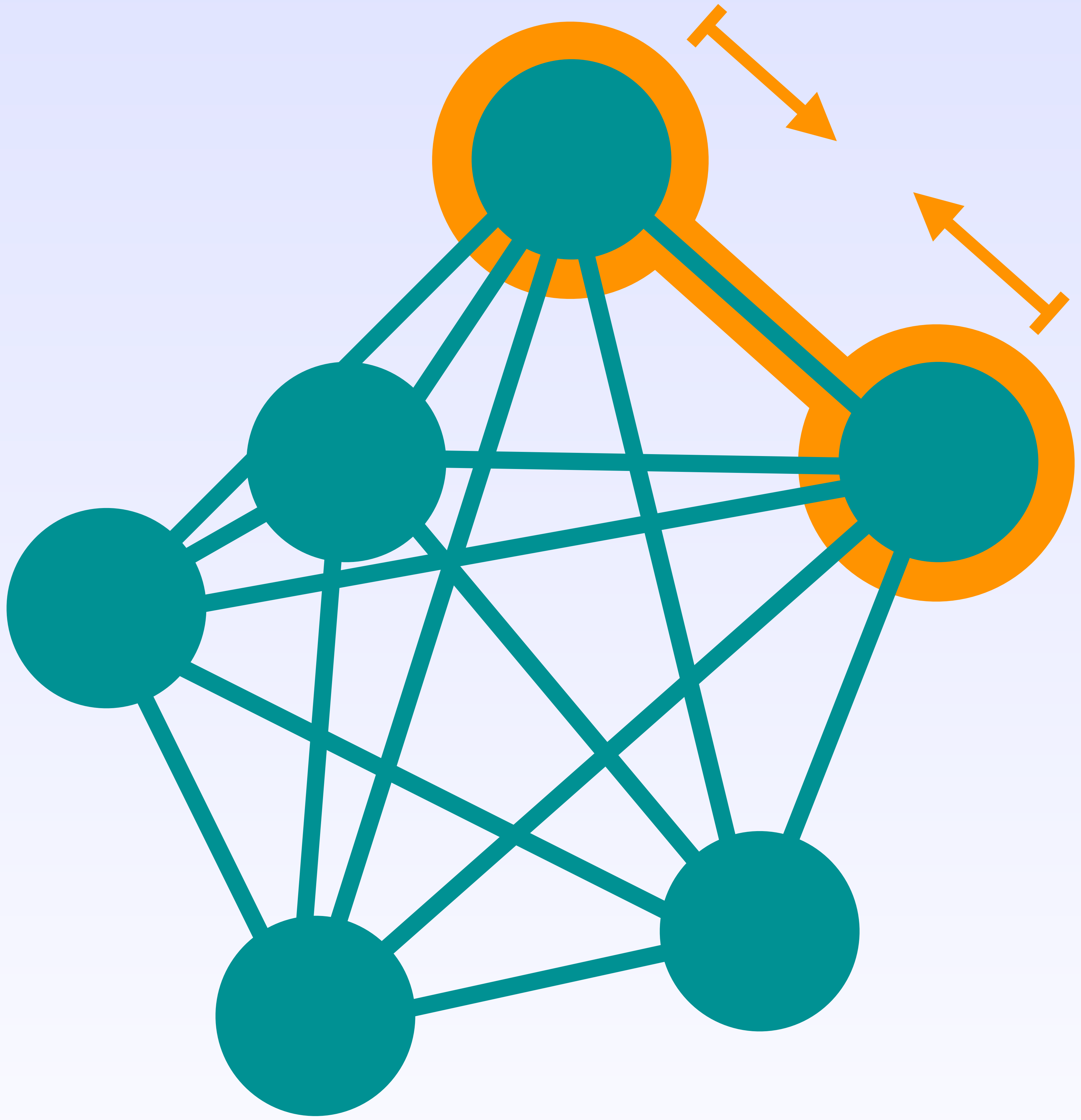
- Naturally encodes singlet *and* triplet pairing because ν takes spins as input

Message-Passing Neural Network



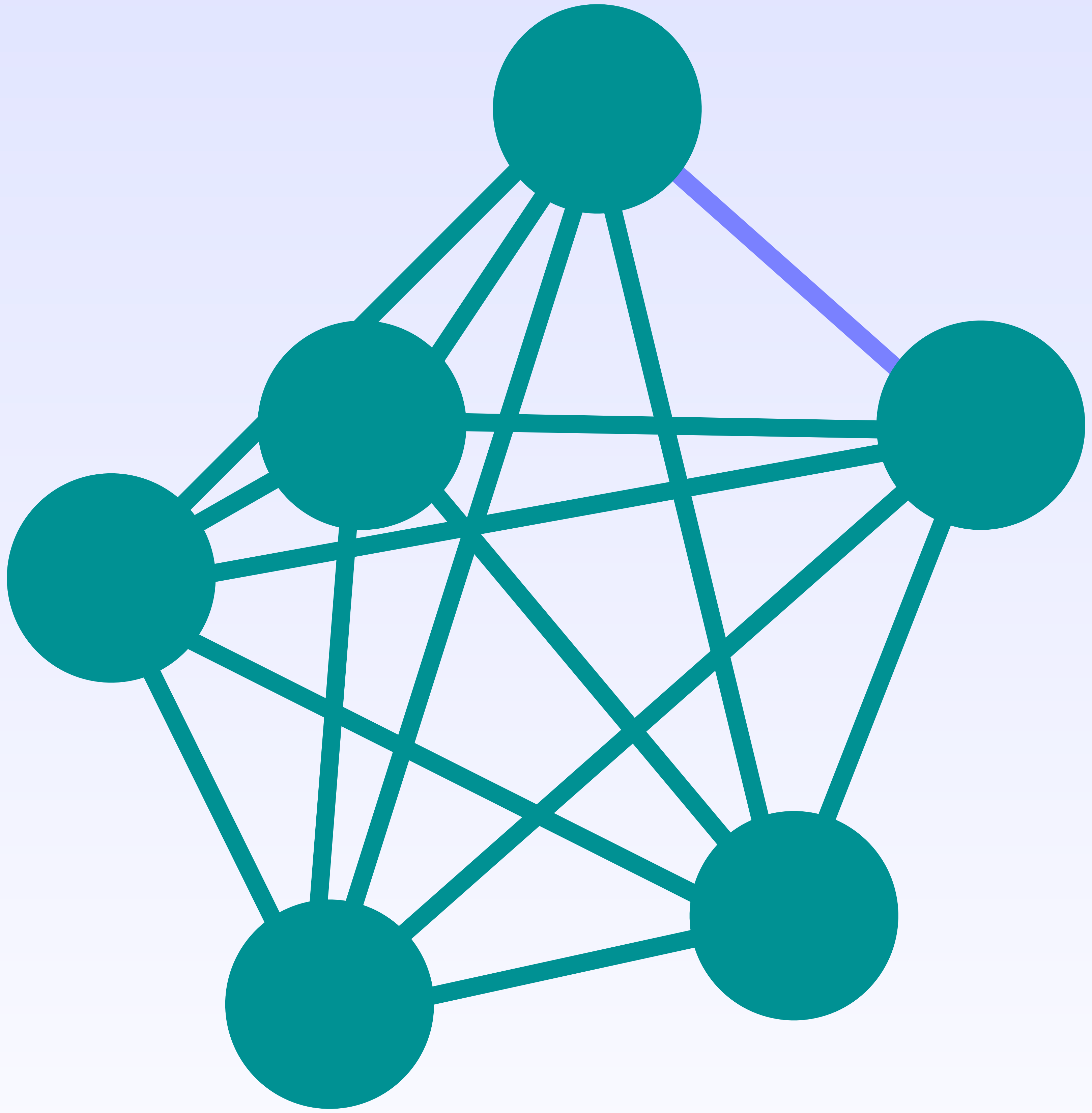
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



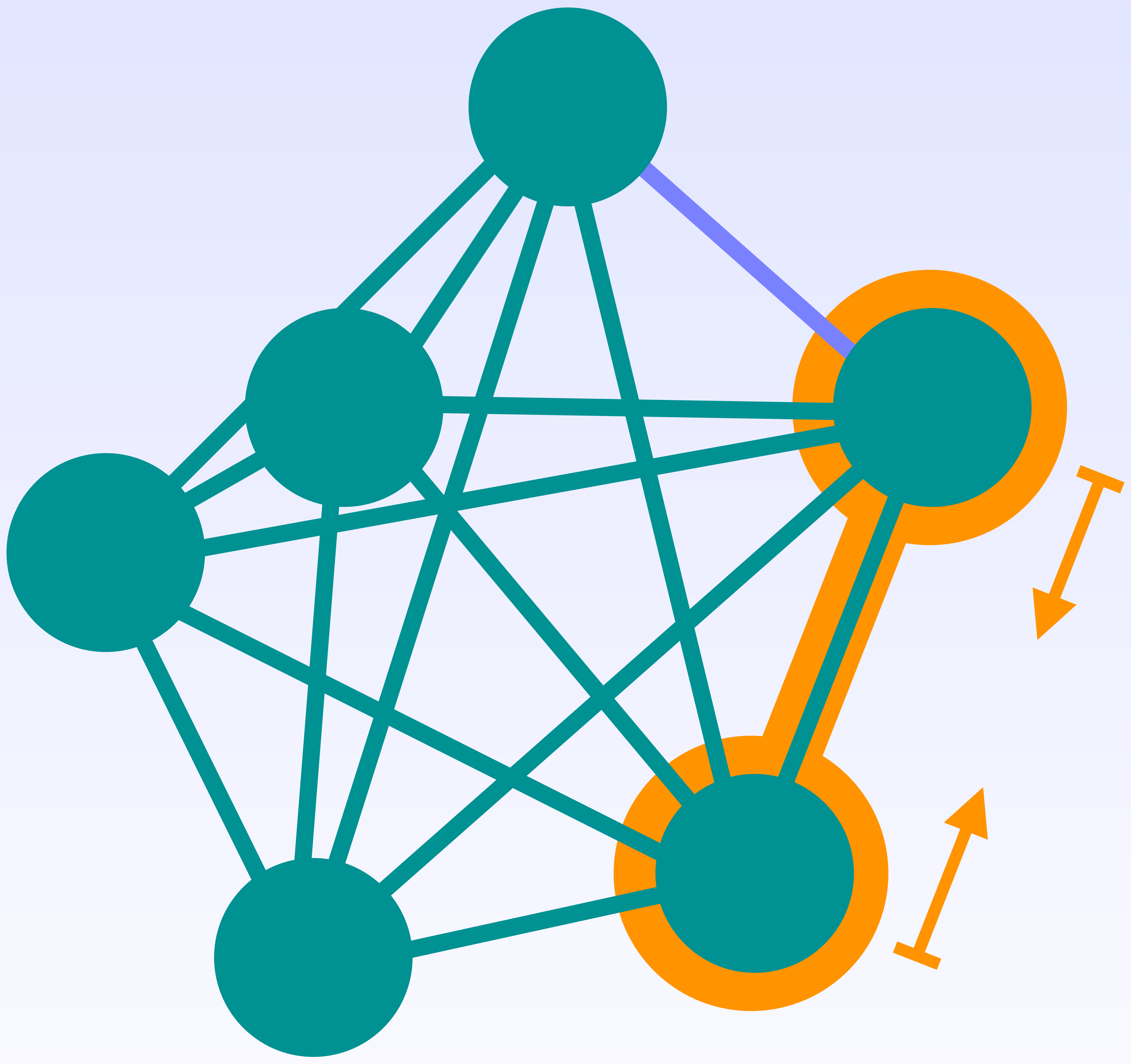
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



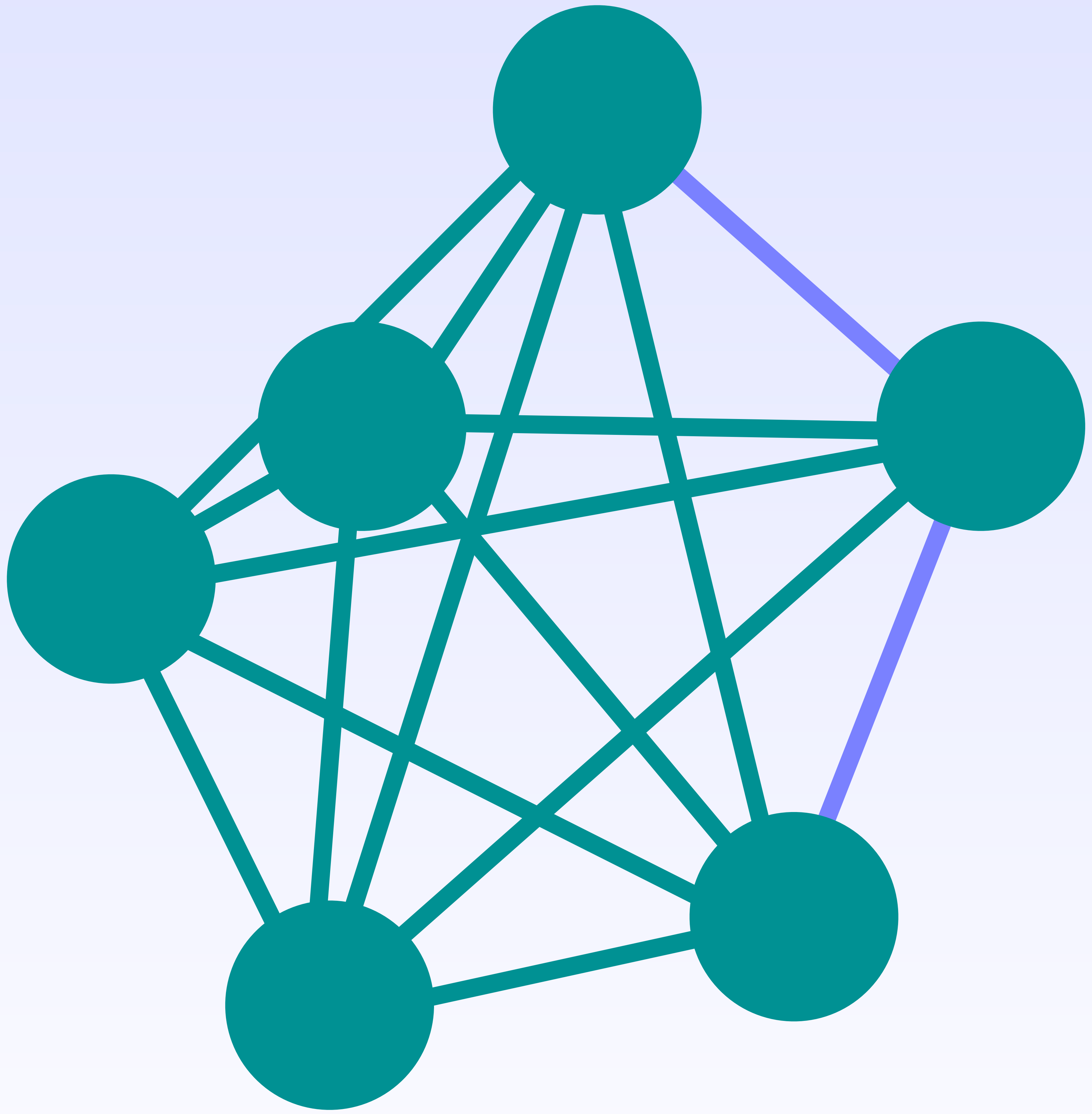
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



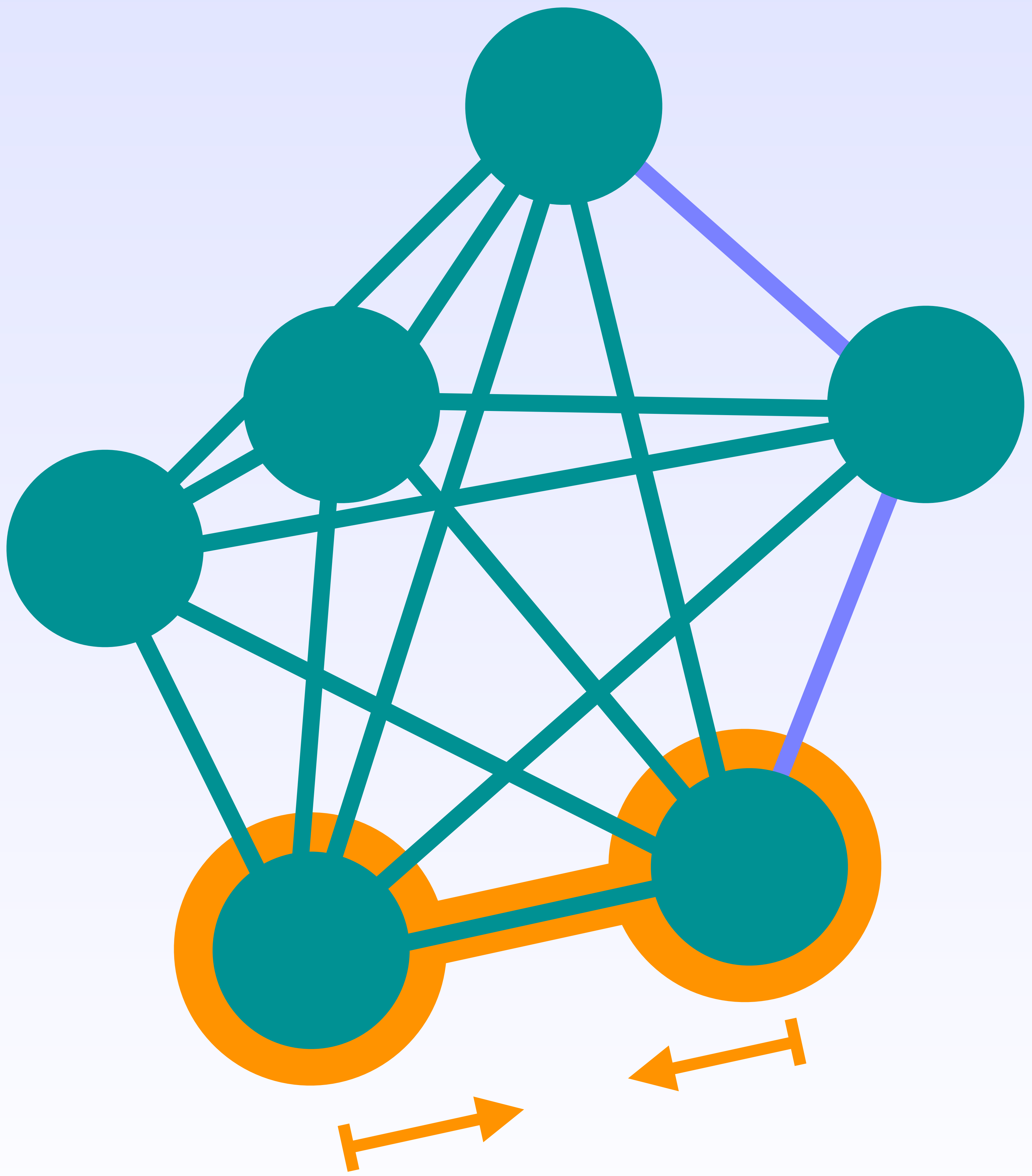
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



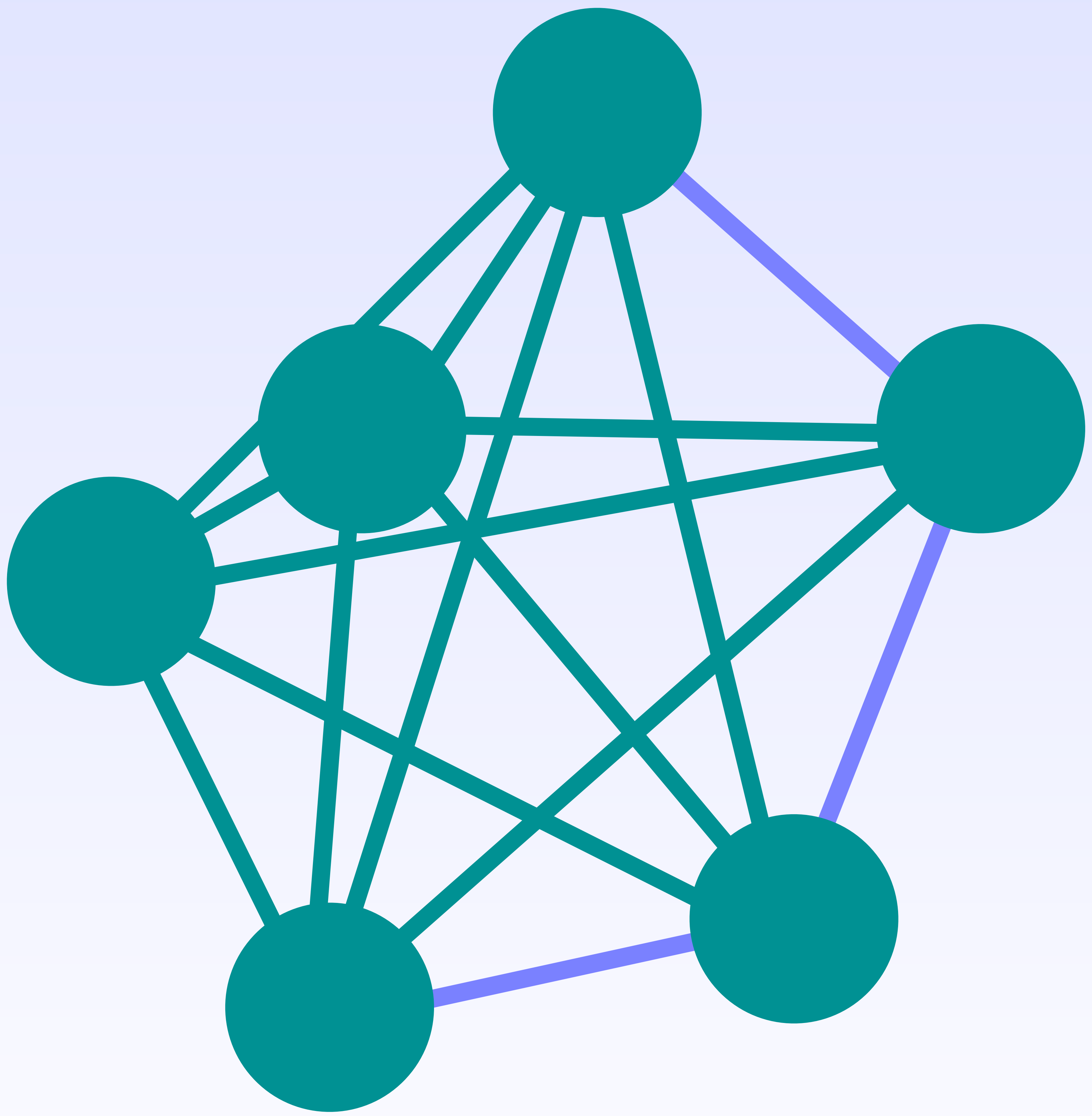
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



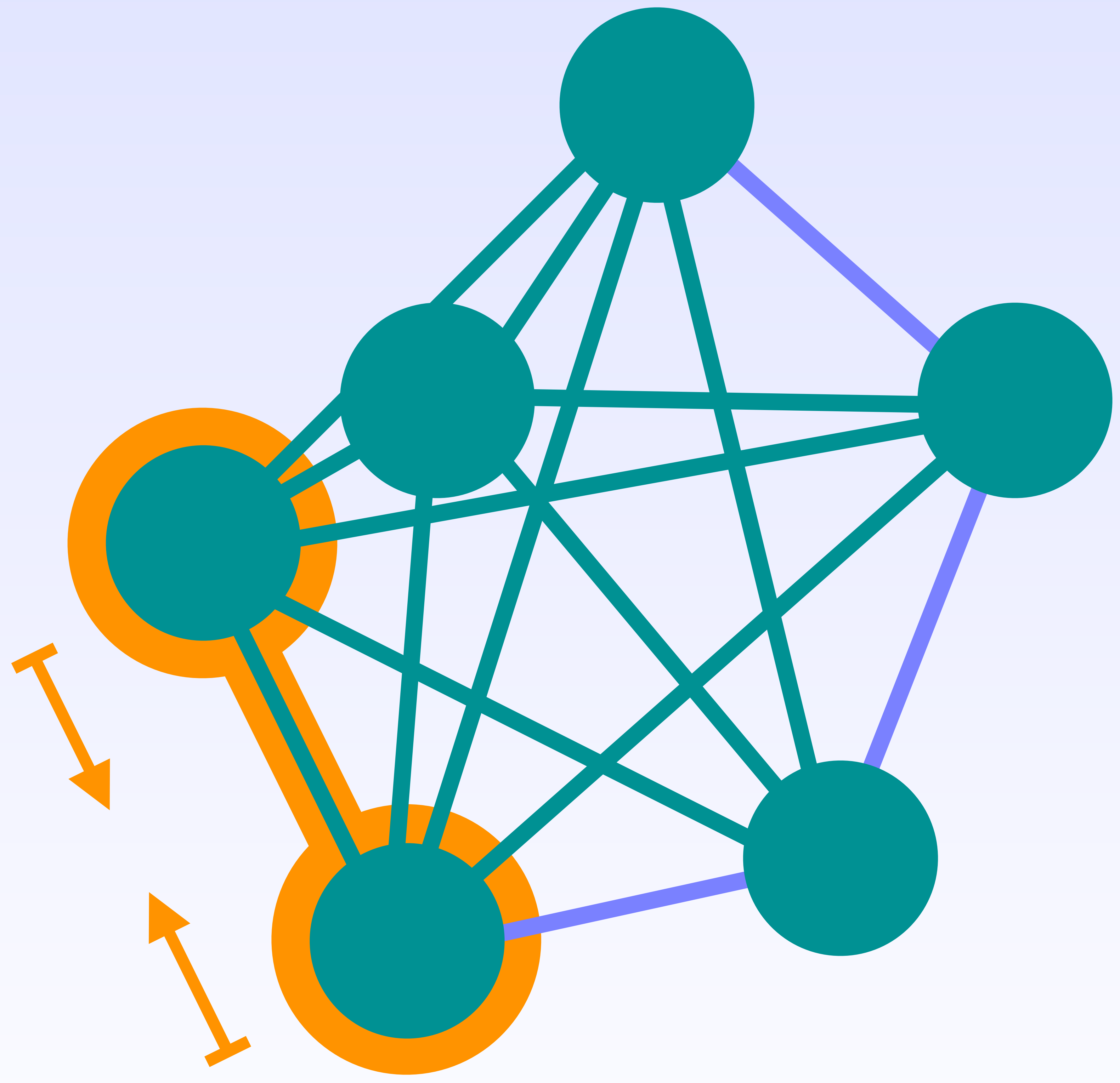
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



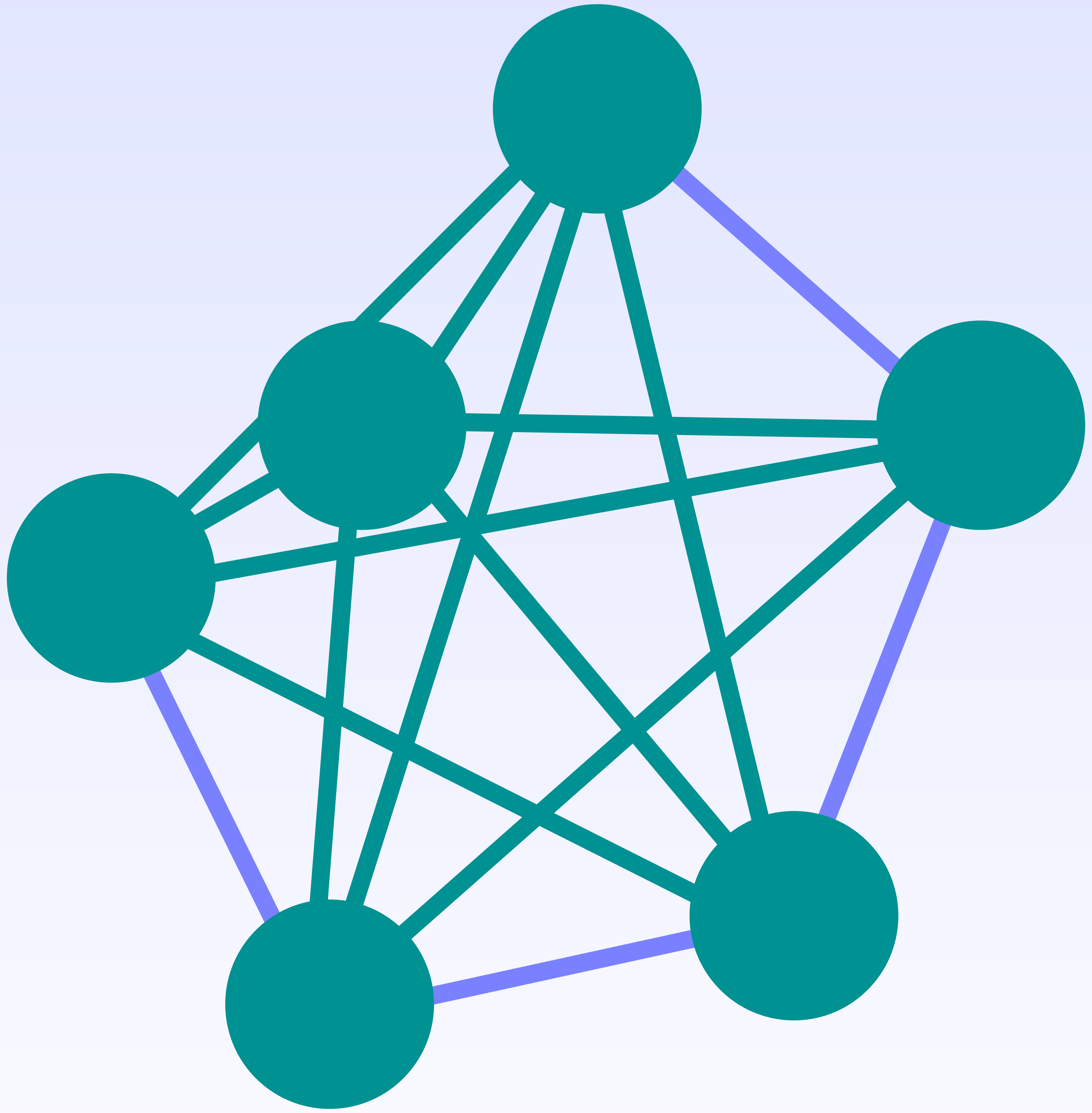
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



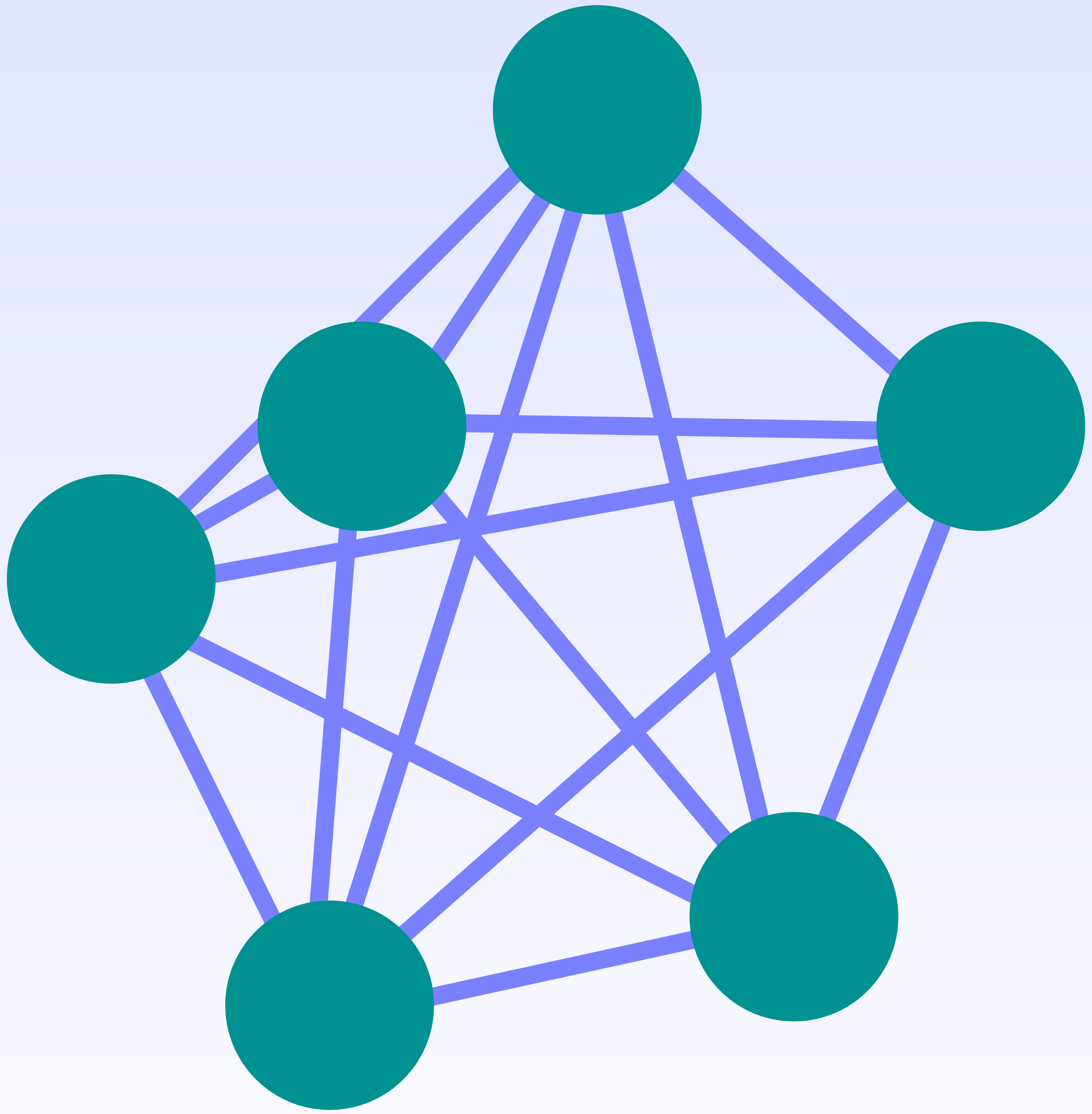
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



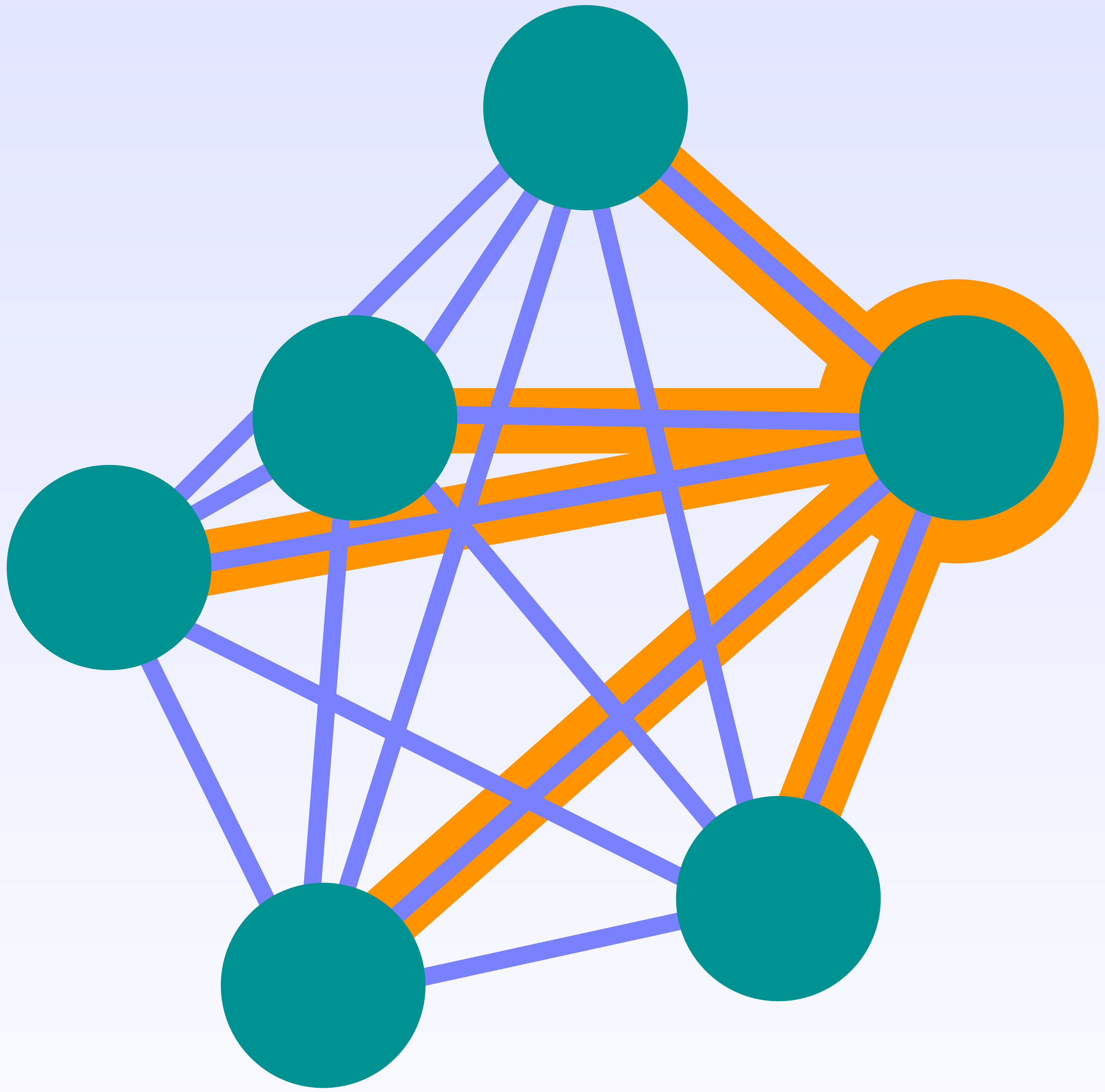
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



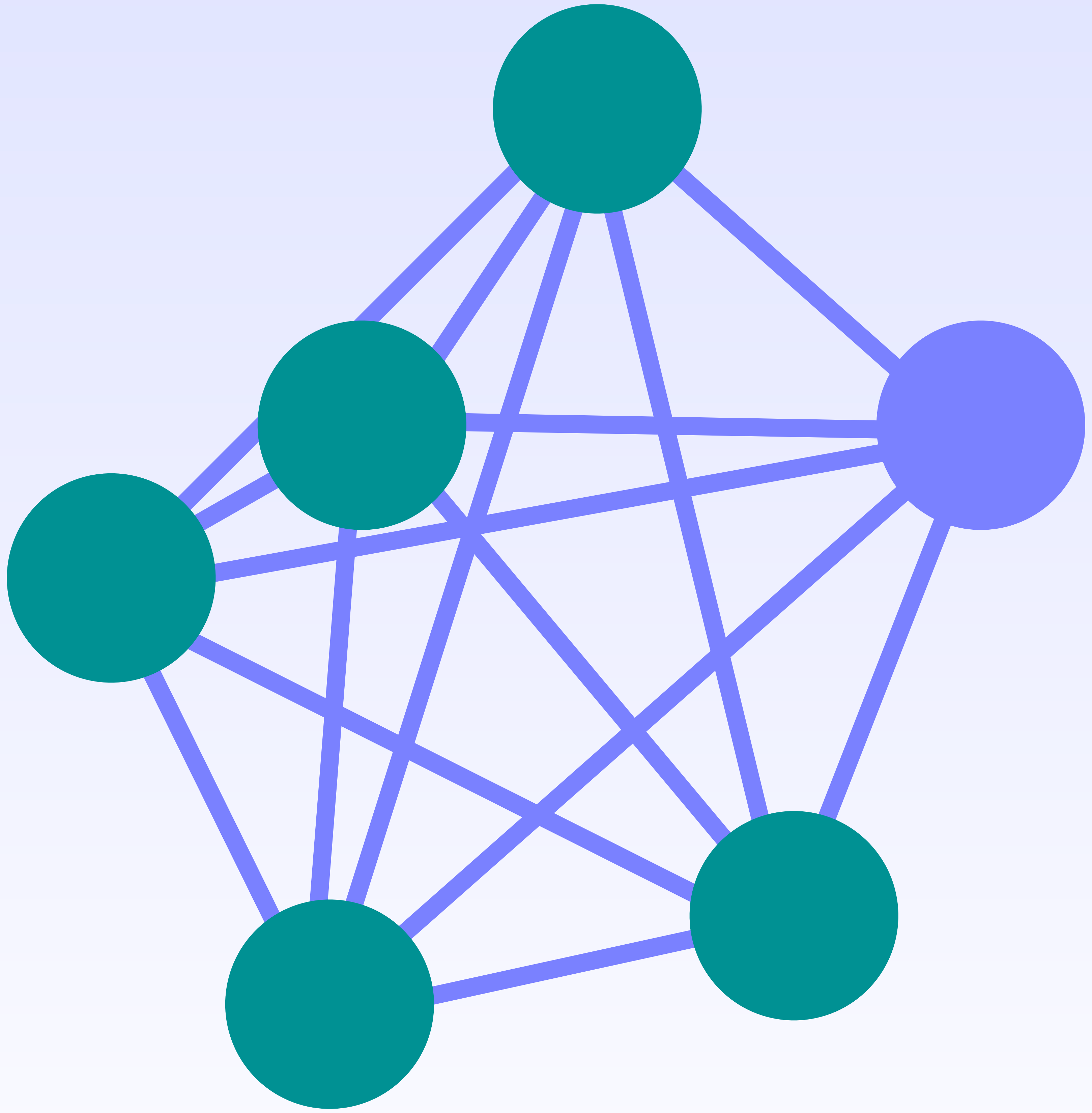
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



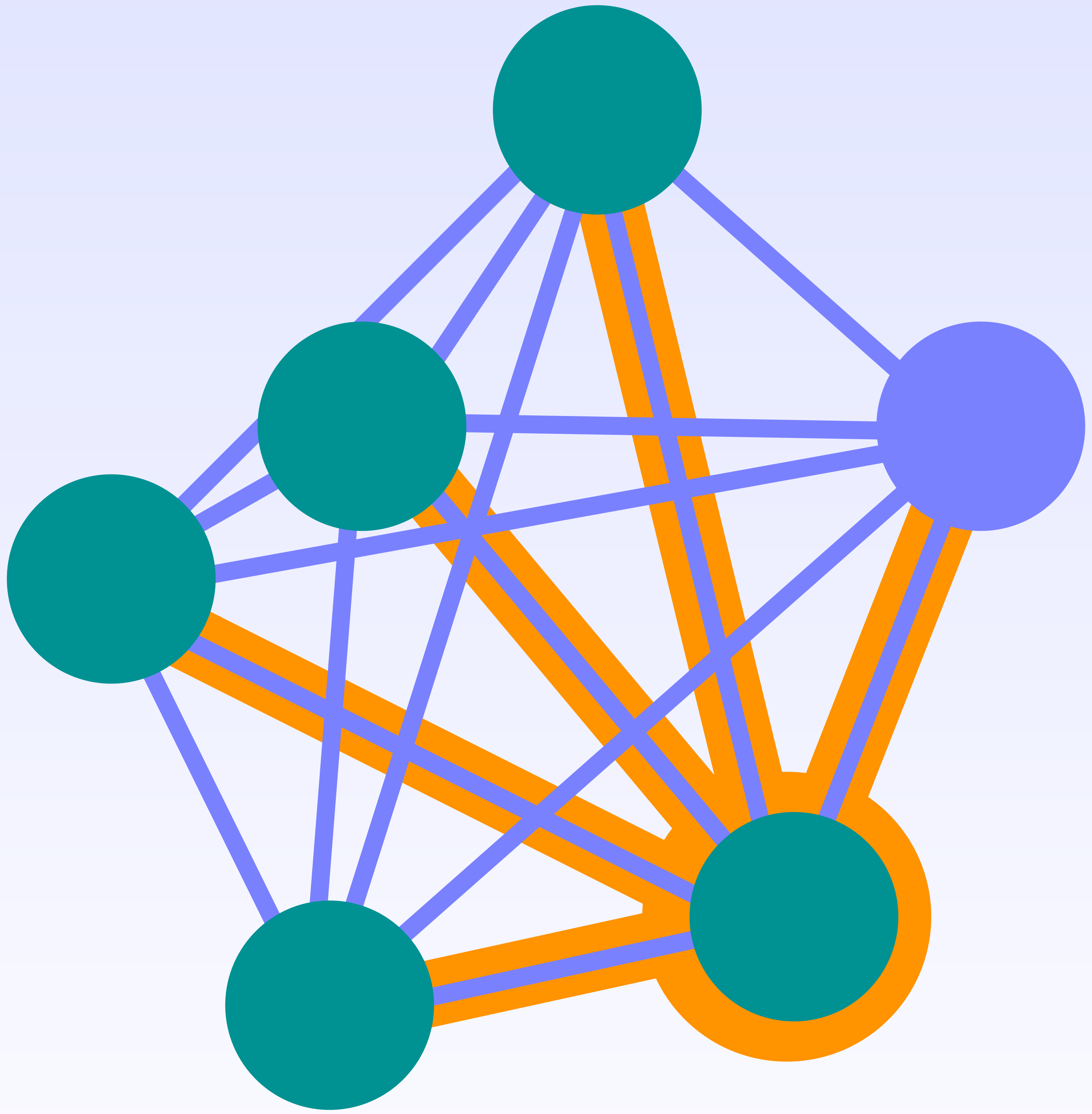
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



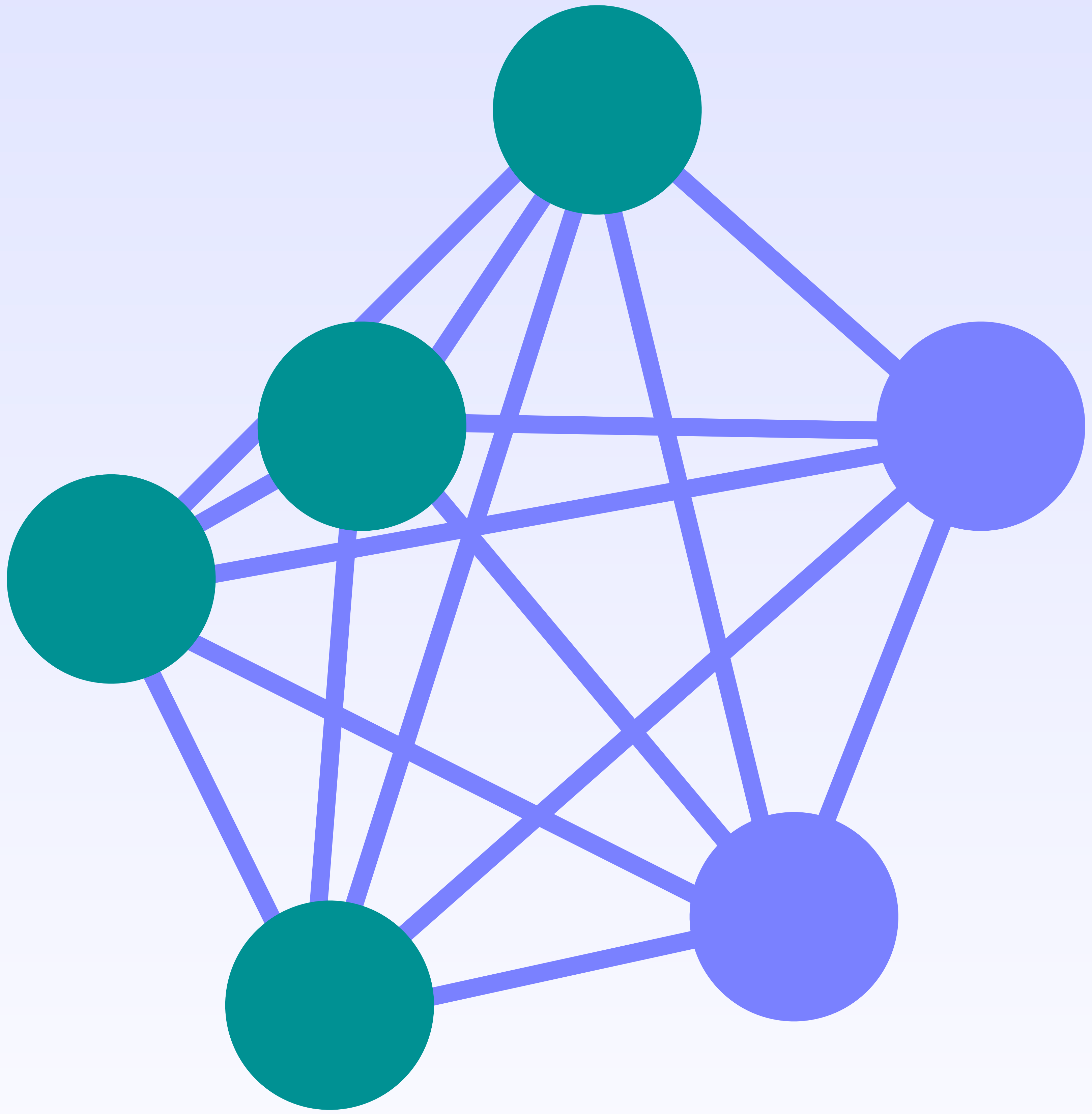
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



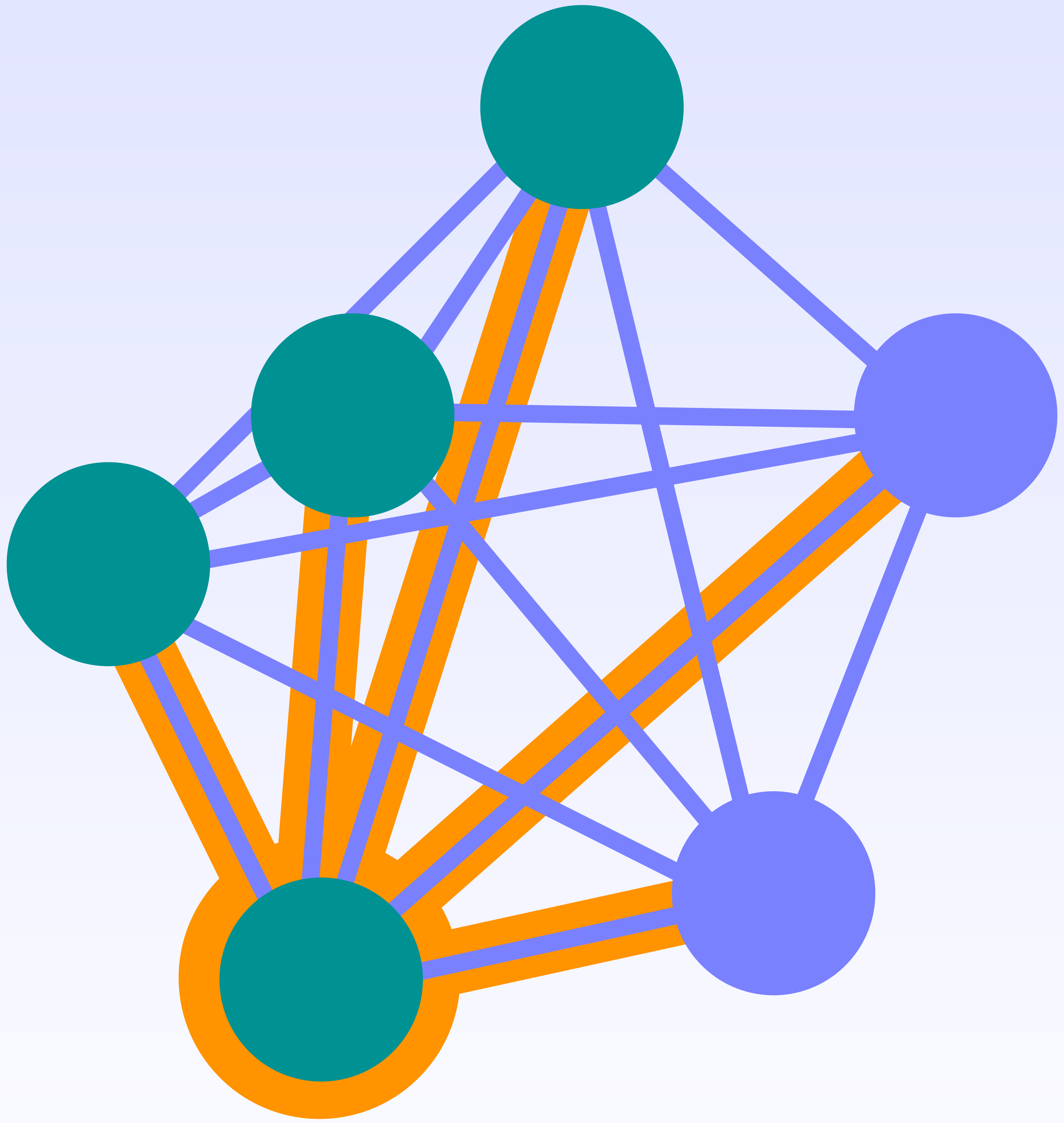
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



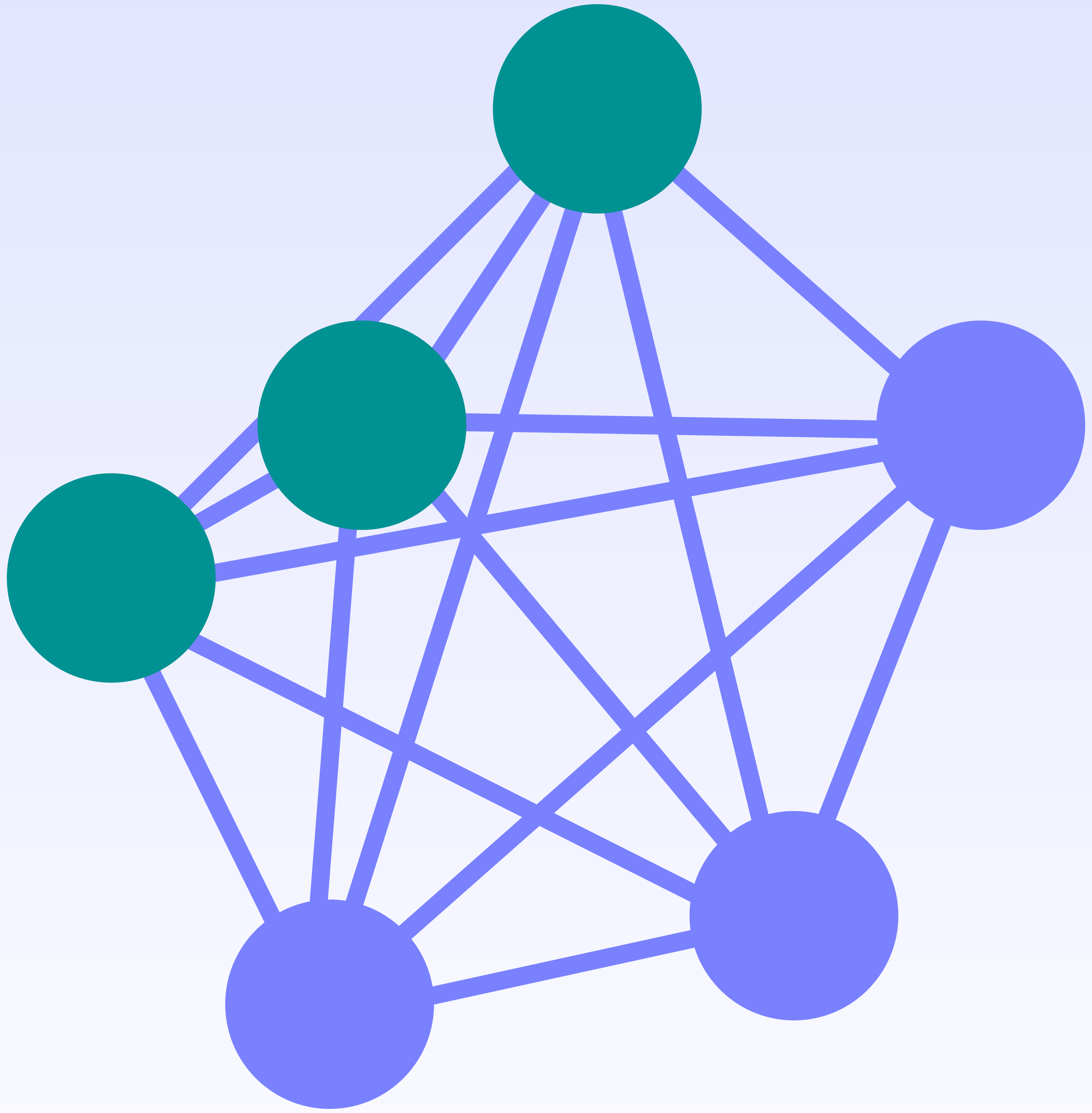
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



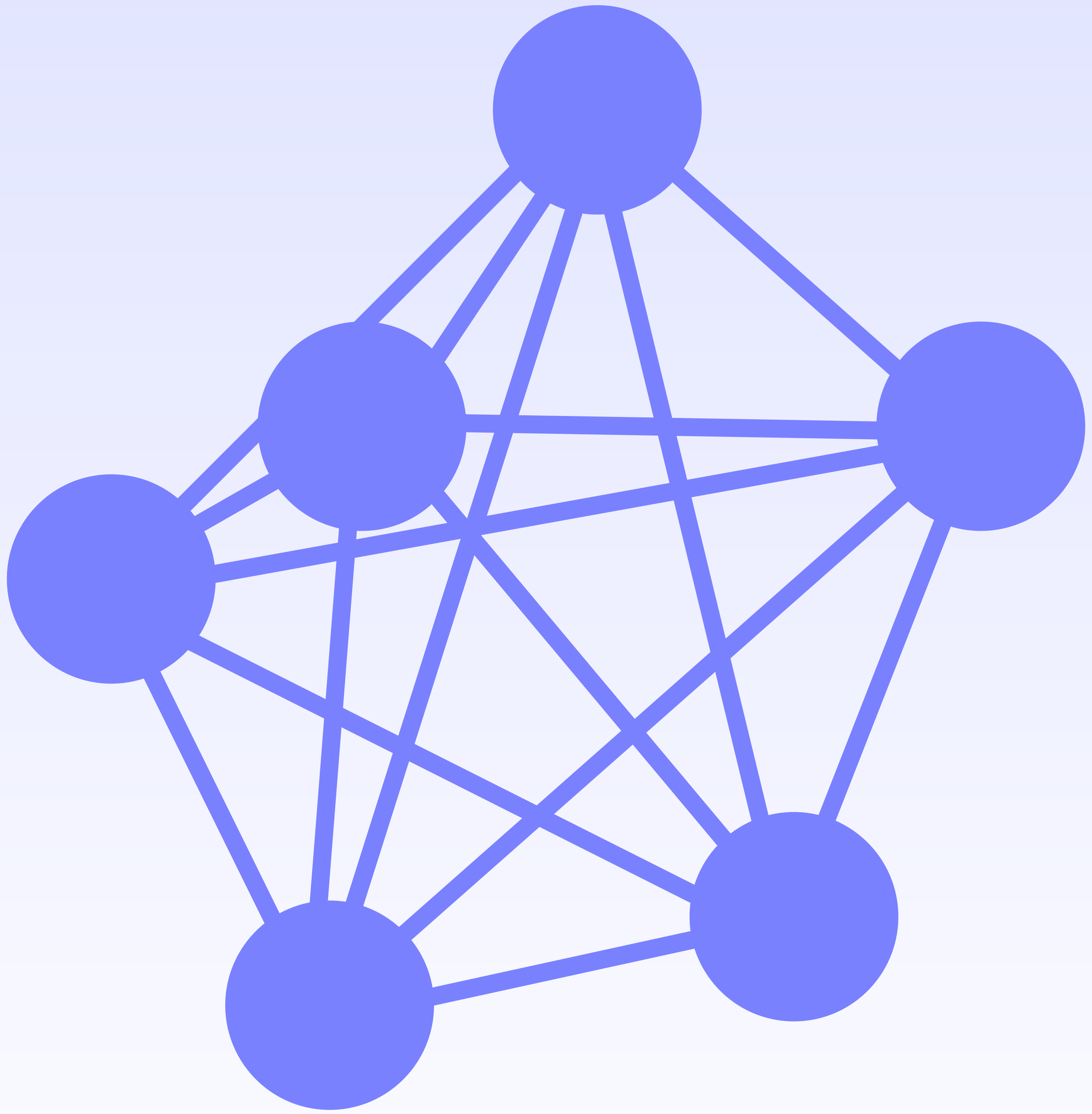
- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

Message-Passing Neural Network



- Used in our study of the homogenous electron gas (Pescia et al. arXiv:2305.07240)
- Type of graph neural network
- Must be permutation equivariant to maintain antisymmetry
- Iteratively build correlations into new one-body and two-body features from original ones
- Nodes = one-body features
- Edges = two-body features

$\mathbf{v}_i$
$\mathbf{h}_i^{(0)}$
$\mathbf{h}_i^{(1)}$
$\mathbf{h}_i^{(T)}$
one-body stream
$\mathbf{m}_{ij}^{(1)}$
$\mathbf{m}_{ij}^{(2)}$
$\mathbf{m}_{ij}^{(T)}$
messages
$\mathbf{g}_{ij}$
$\mathbf{v}_{ij}$
$\mathbf{h}_{ij}^{(0)}$
$\mathbf{h}_{ij}^{(1)}$
$\mathbf{h}_{ij}^{(T)}$
two-body stream
layer 1
layer 2
layer T

Pfaffian Wave Function with Backflow

- Use output of MPNN as input to pairing orbital instead of raw coordinates

$$\Phi(X) = \mathrm{pf}\left[\phi(\mathbf{g}_{ij})\right]$$

- Jastrow correlator based on a “Deep Set” (Zaheer et al. - arXiv:1703.06114)

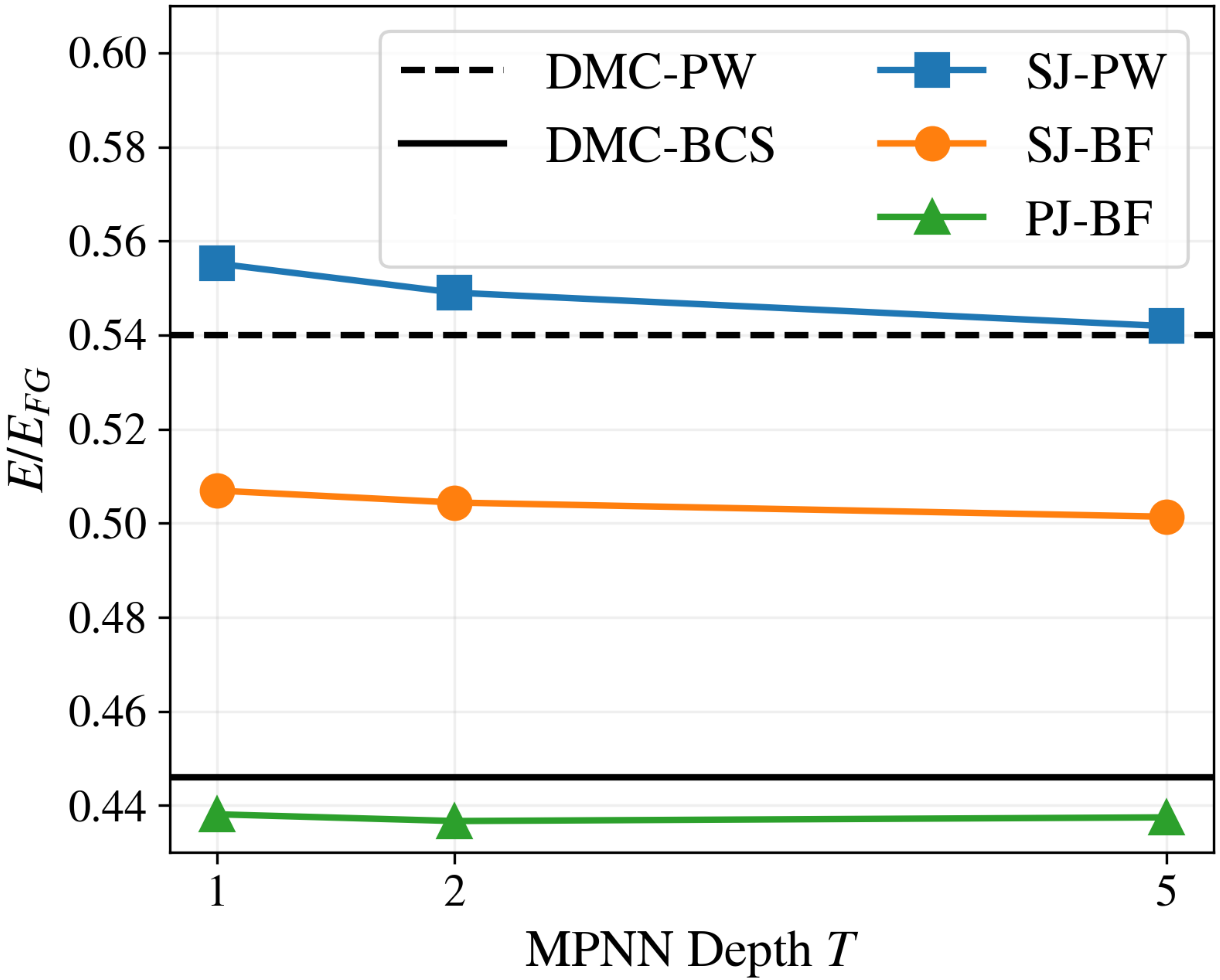
$$J(X) = \rho\left(\sum_{i \neq j} \zeta(\mathbf{g}_{ij})\right)$$

  where ρ and ζ are neural networks.

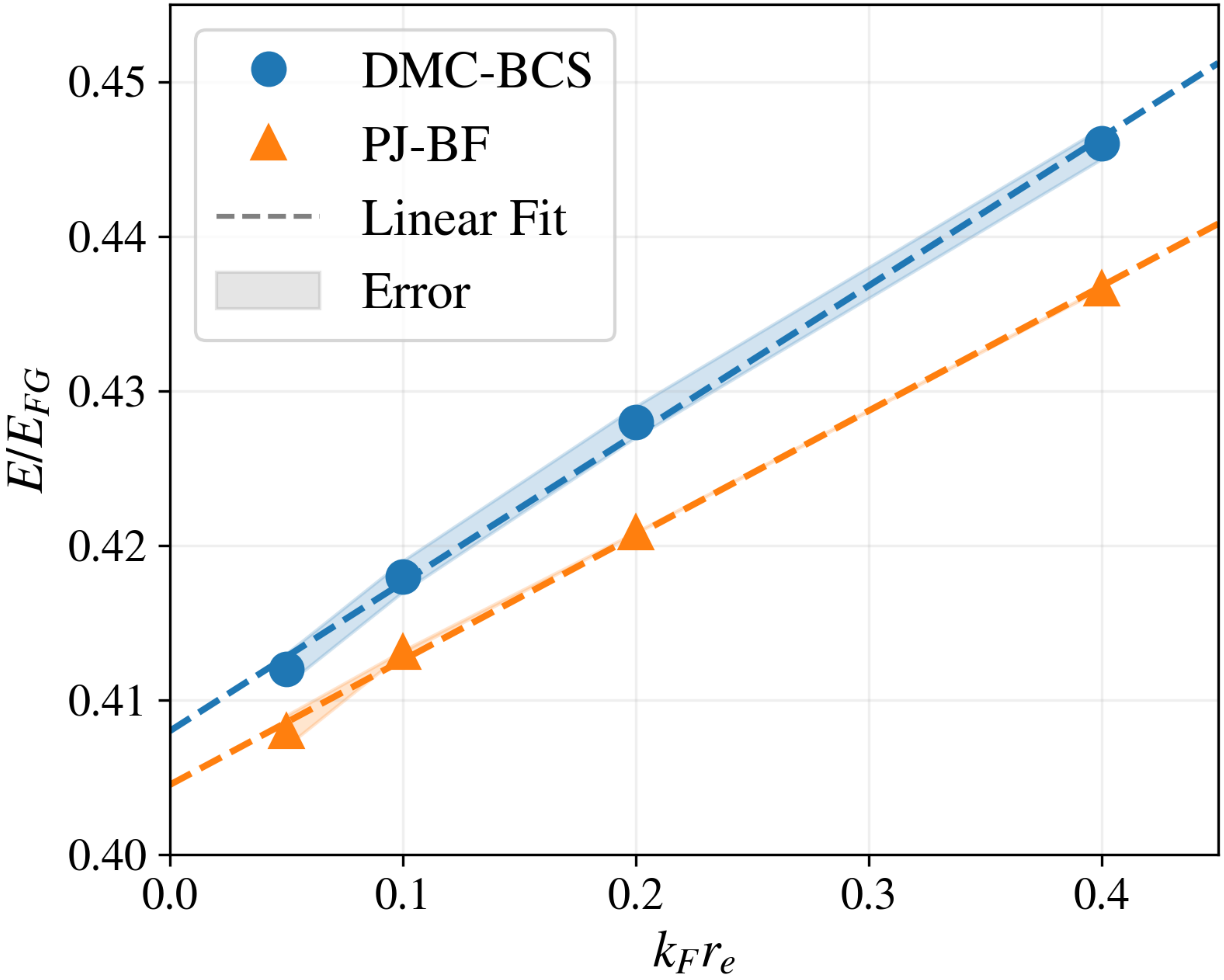
- Most results shown use an MPNN with 2 iterations (~8500 parameters total)

- We also enforce periodicity, translational invariance, parity and time-reversal symmetry

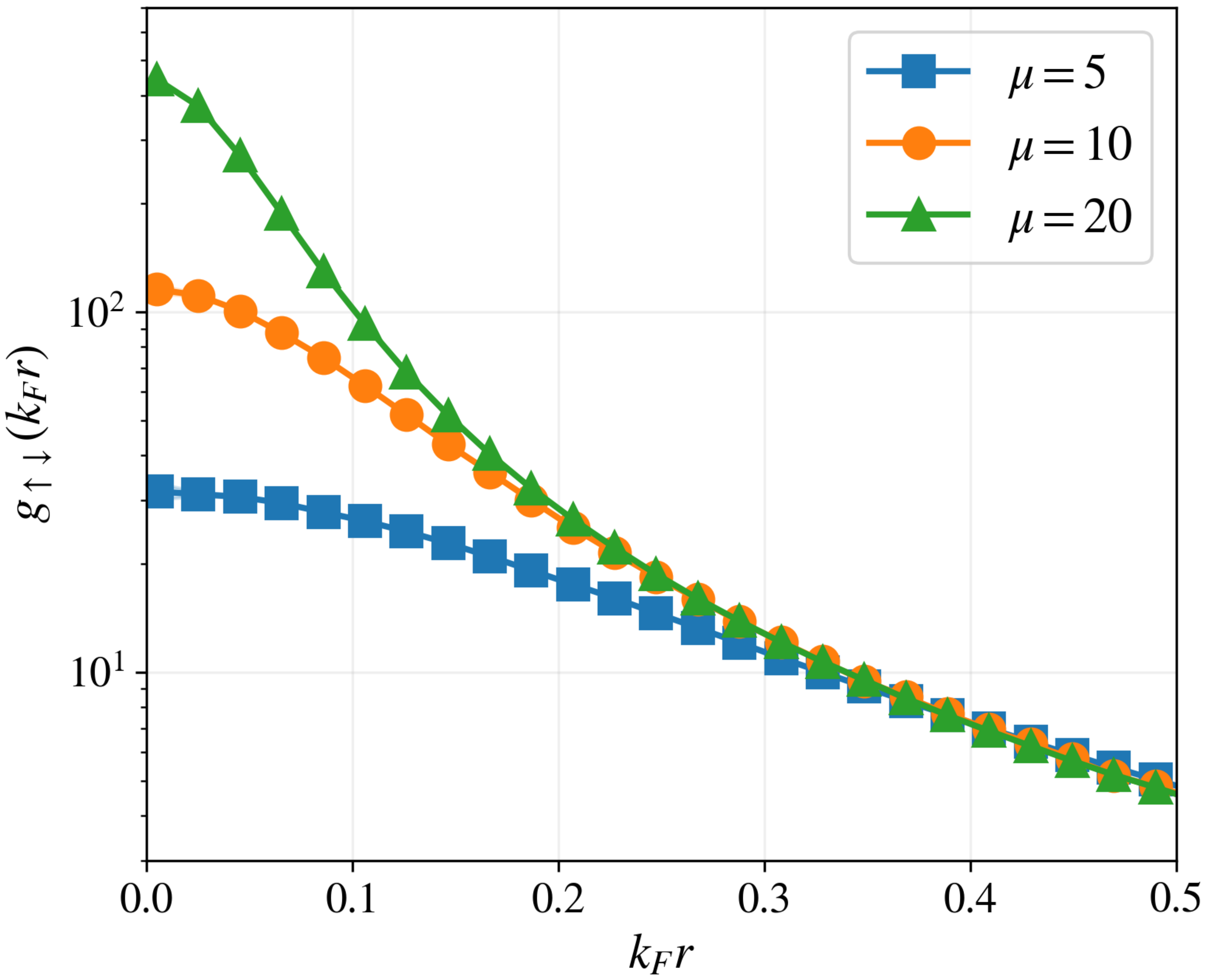
- This is the first time <u>neural</u> backflow transformations have been applied to the Pfaffian :)



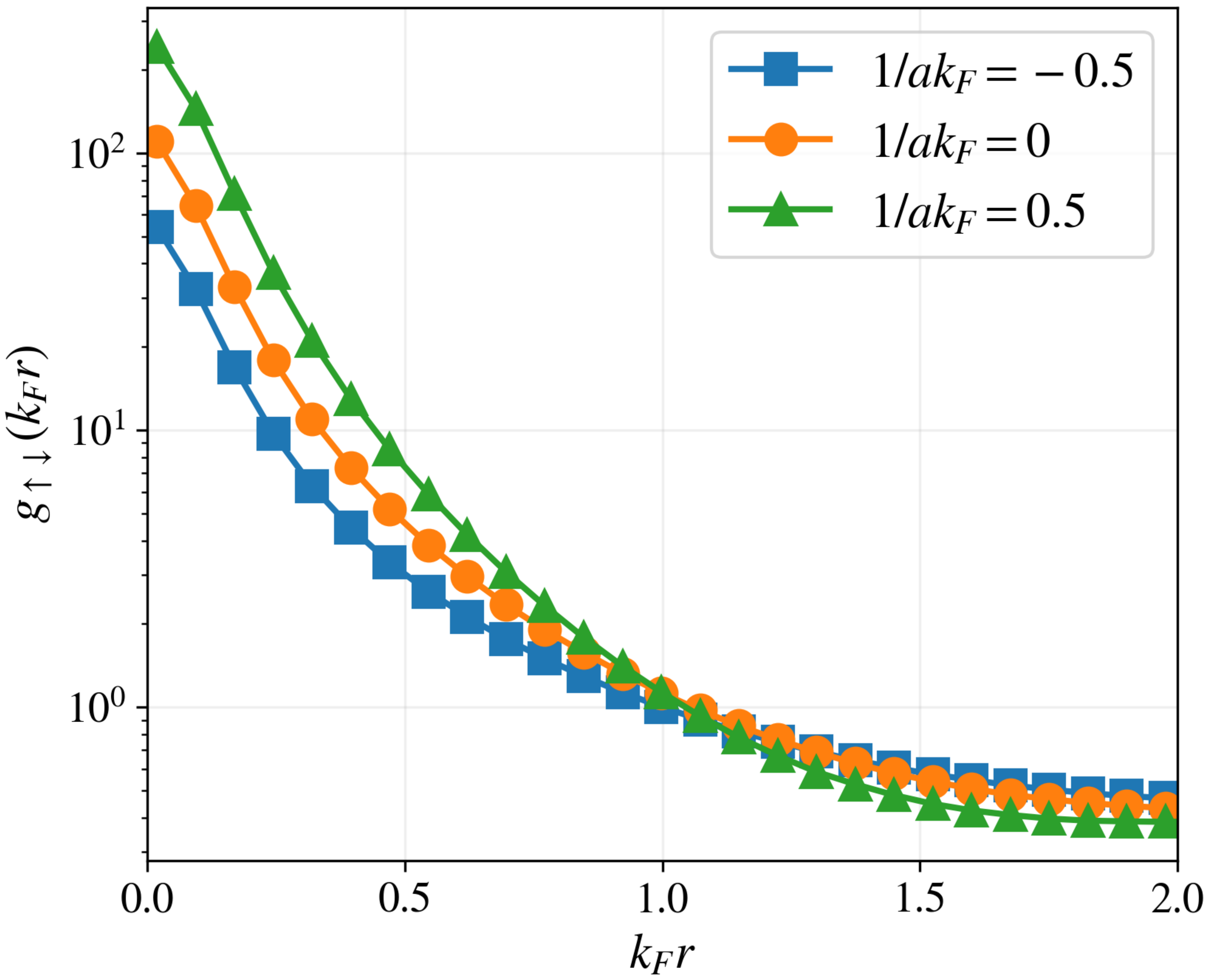
0.60
0.58
0.56
0.54
0.52
0.50
0.48
0.46
0.44
E/E_{FG}
1
2
5
MPNN Depth T
DMC-PW
DMC-BCS
SJ-PW
SJ-BF
PJ-BF

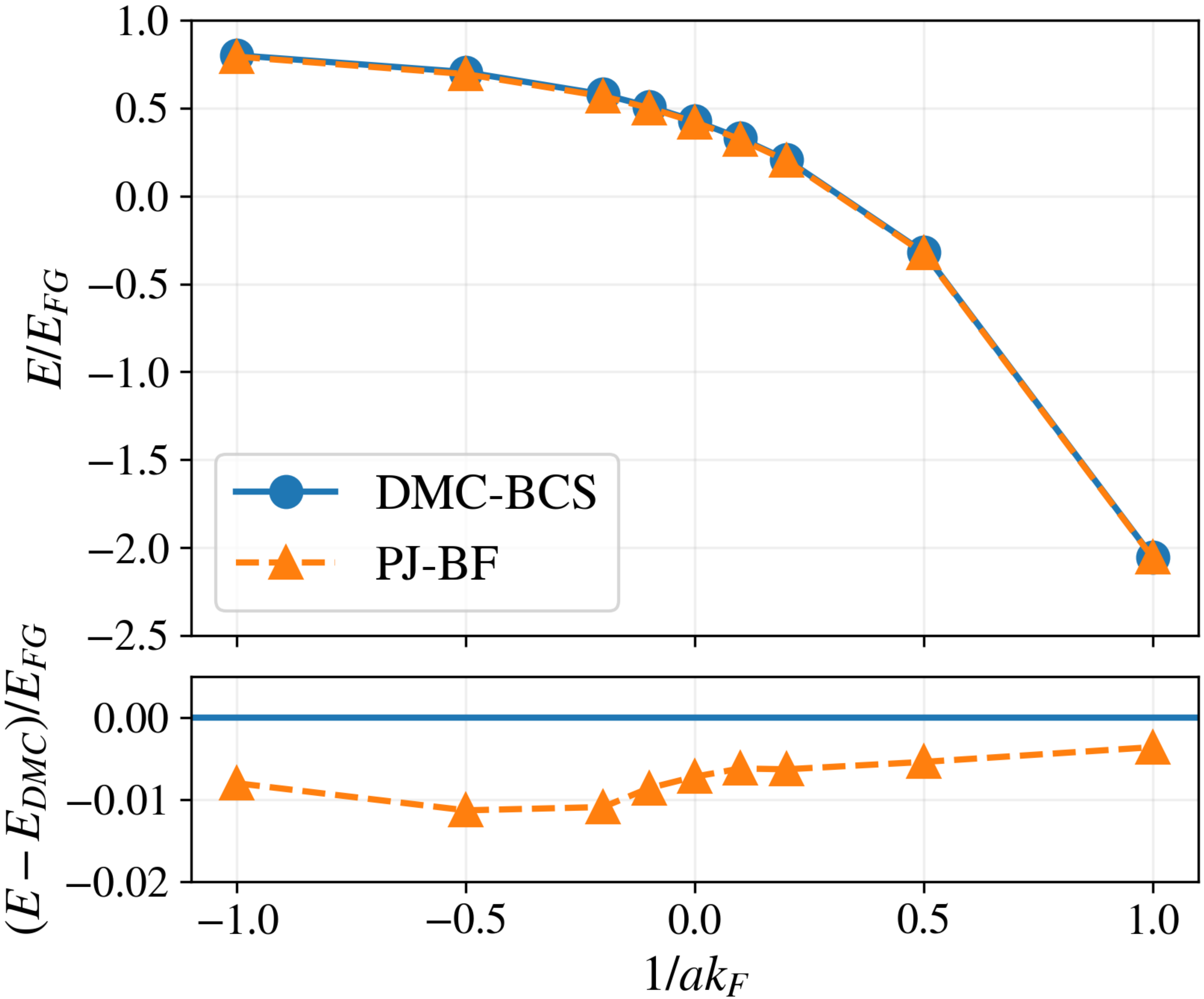

DMC-BCS
PJ-BF
Linear Fit
Error
E/E_{FG}
$k_F r_e$
0.40
0.41
0.42
0.43
0.44
0.45
0.0
0.1
0.2
0.3
0.4

Different effective ranges $r_e = 2/\mu$ at unitarity $(1/ak_F = 0)$



Different scattering lengths near unitarity (fixed $r_e = 0.2$)





1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
E/E_{FG}
DMC-BCS
PJ-BF
0.00
−0.01
−0.02
$(E - E_{DMC})/E_{FG}$
−1.0
−0.5
0.0
0.5
1.0
$1/ak_F$

Conclusions

"Neural-network quantum states for ultra-cold Fermi gases"
arXiv:2305.08831

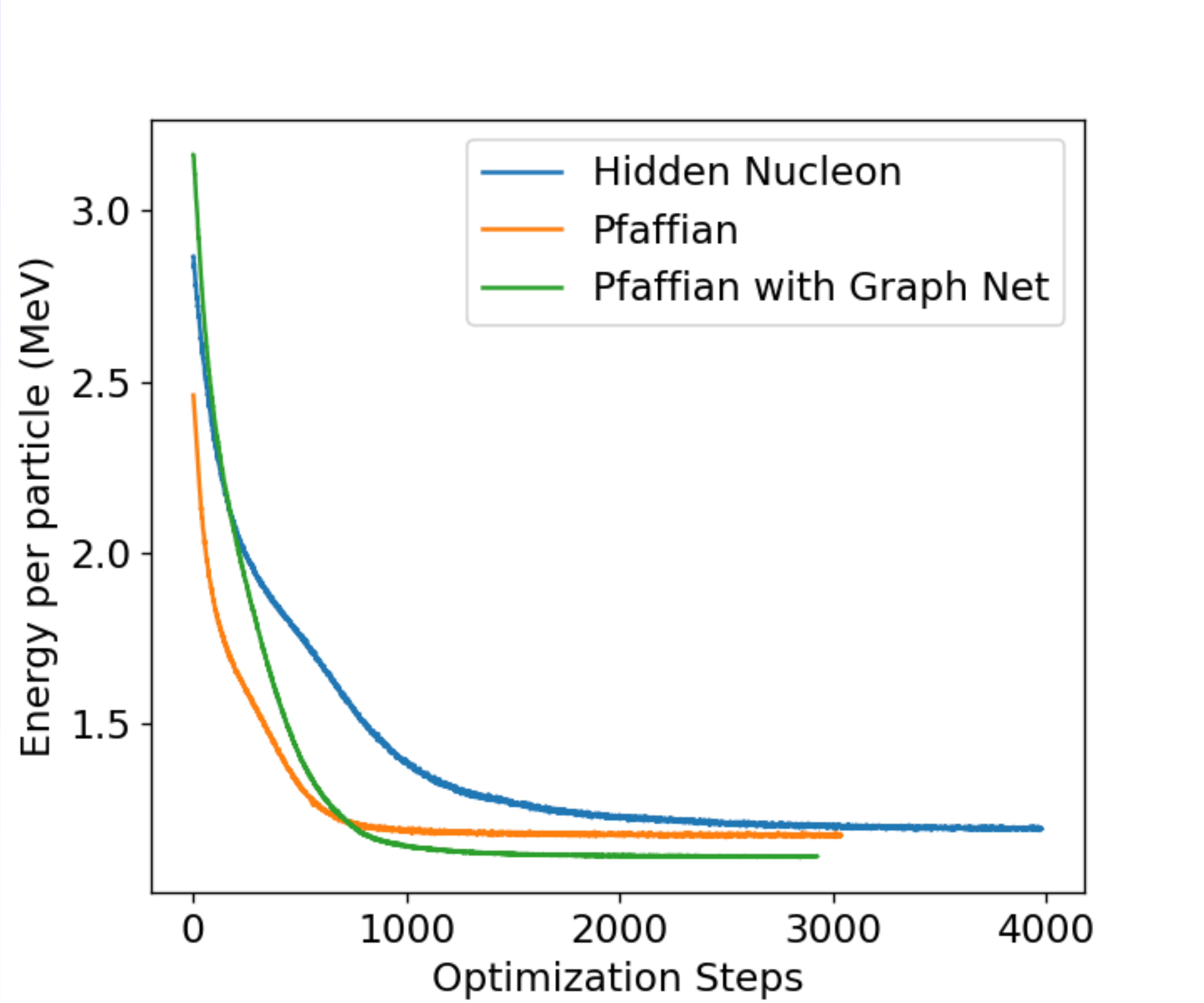
- Our Pfaffian ansatz is very general — works for any unpolarized system and any Hamiltonian (even those that exchange spin!)
- Can obtain lower energies than state-of-the-art diffusion Monte Carlo methods
- Our message-passing neural network efficiently builds pairing and backflow correlations
- We require far fewer parameters compared to other NQS applied to similar problems (~8500 vs millions)
- Future work:
  - Calculate the gap: expand the Pfaffian matrix to include one unpaired single-particle orbital
  - Smaller r_e: better linear extrapolation to the $r_e \to 0$ limit
  - Larger N: As an initial test, we only used $N = 14$

Thank you!

Questions?

Infinite Neutron Matter

- Preliminary tests at low density $\rho \approx 0.002\ \mathrm{fm}^{-3}$
- Pfaffian with 1-layer MPNN (Graph Net) performs better than hidden nucleon and pfaffian ansatzë

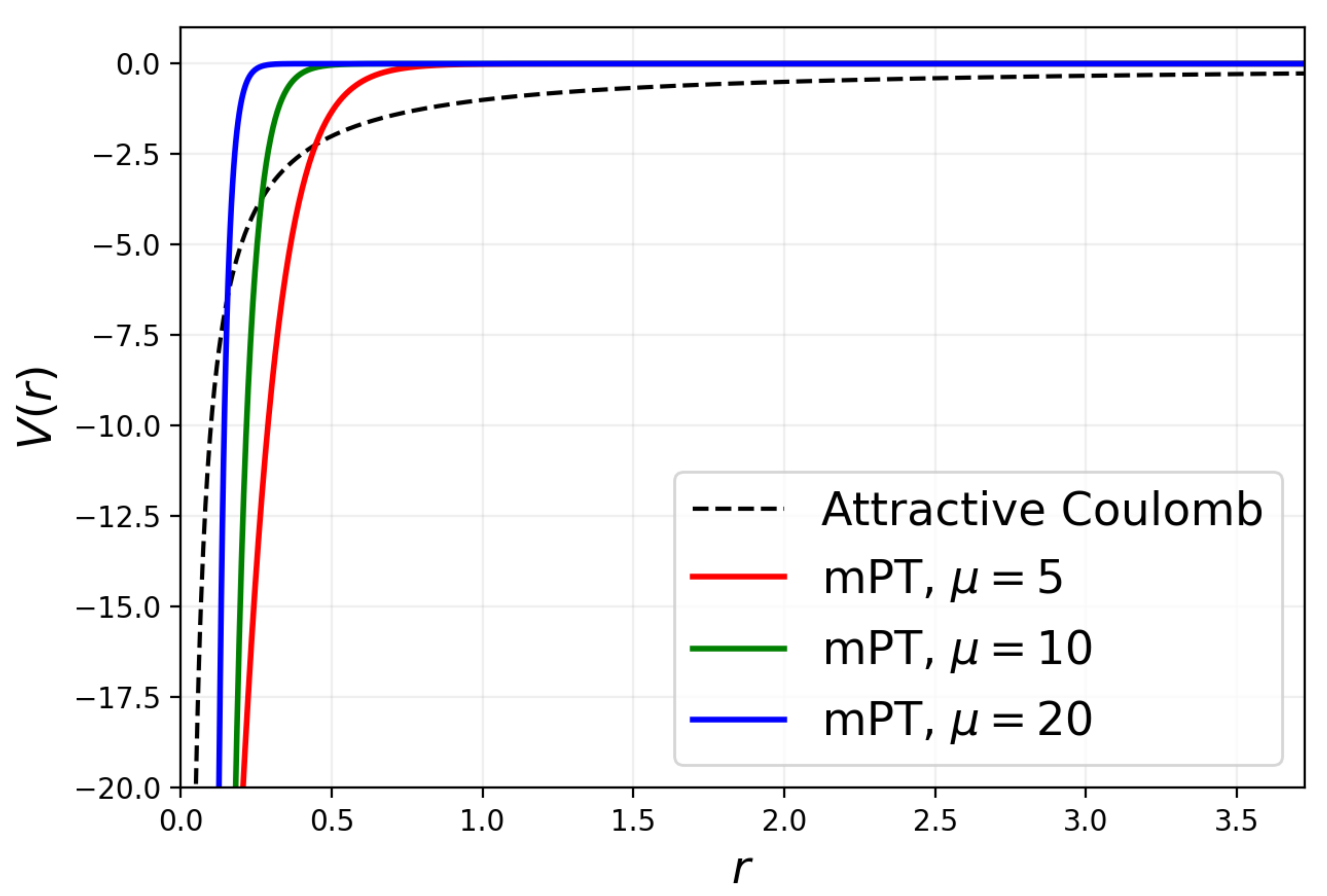


Pöschl-Teller Potential

- Regularized, short-range attraction between opposite-spin pairs

$$V_{ij} = (s_i^z s_j^z - 1)\ v_0 \frac{\hbar^2}{2m} \frac{\mu^2}{\cosh^2(\mu r_{ij})}$$

- Effective range $\approx 2/\mu$
- Keep v_0 fixed, pretrain with small μ
- Provides exact solution of two-body problem
- Other interaction potentials give similar results near unitarity as long as effective range is the same



Slater Determinant

$$\Phi(X) = \det[\phi_\alpha(\mathbf{x}_i)] = \det \begin{bmatrix} \phi_1(\mathbf{x}_1) & \phi_1(\mathbf{x}_2) & \cdots & \phi_1(\mathbf{x}_N) \\ \phi_2(\mathbf{x}_1) & \phi_2(\mathbf{x}_2) & \cdots & \phi_2(\mathbf{x}_N) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_N(\mathbf{x}_1) & \phi_N(\mathbf{x}_2) & \cdots & \phi_N(\mathbf{x}_N) \end{bmatrix}$$

Number-projected BCS Wave Function

$$\Phi(X) = \det\left[\phi(\mathbf{x}_{i\uparrow}, \mathbf{x}_{j\downarrow})\right] = \det\begin{bmatrix} \phi(\mathbf{x}_{1\uparrow}, \mathbf{x}_{1\downarrow}) & \phi(\mathbf{x}_{1\uparrow}, \mathbf{x}_{2\downarrow}) & \cdots & \phi(\mathbf{x}_{1\uparrow}, \mathbf{x}_{N/2\downarrow}) \\ \phi(\mathbf{x}_{2\uparrow}, \mathbf{x}_{1\downarrow}) & \phi(\mathbf{x}_{2\uparrow}, \mathbf{x}_{2\downarrow}) & \cdots & \phi(\mathbf{x}_{2\uparrow}, \mathbf{x}_{N/2\downarrow}) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(\mathbf{x}_{N/2\uparrow}, \mathbf{x}_{1\downarrow}) & \phi(\mathbf{x}_{N/2\uparrow}, \mathbf{x}_{2\downarrow}) & \cdots & \phi(\mathbf{x}_{N/2\uparrow}, \mathbf{x}_{N/2\downarrow}) \end{bmatrix}$$

Periodic Boundary Conditions

- Separation vector

$$\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j \;\longmapsto\; \Big(\cos(2\pi\mathbf{r}_{ij}/L),\; \sin(2\pi\mathbf{r}_{ij}/L)\Big)$$

- Distance

$$\|\mathbf{r}_{ij}\| \;\longmapsto\; \|\sin(2\pi\mathbf{r}_{ij}/L)\|$$

- Absolute positions are ignored to enforce translational invariance

Parity and Time-Reversal Symmetry

- We carry out the VMC calculations for the unpolarized gas using $\Psi^{PT}(R, S)$ given by

$$\Psi^{P}(R, S) = \Psi(R, S) + \Psi(-R, S)$$

$$\Psi^{PT}(R, S) = \Psi^{P}(R, S) + (-1)^{N/2}\Psi^{P}(R, -S)$$

where R and S are the set of all positions and spins, respectively.

Message-Passing Neural Network Equations

- Iteratively builds correlations into new one- and two-body features from old ones
- Skip connections help stabilize training and avoid vanishing gradients
- Has been effective for the electron gas, despite having orders of magnitude fewer parameters compared to FermiNet

$$\mathbf{v}_i = (s_i^z)$$
$$\mathbf{v}_{ij} = (r_{ij}, \mathbf{r}_{ij}, s_i^z \cdot s_j^z)$$

$$\mathbf{h}_i^{(0)} = (\mathbf{v}_i, A\mathbf{v}_i)$$
$$\mathbf{h}_{ij}^{(0)} = (\mathbf{v}_{ij}, B\mathbf{v}_{ij})$$

for $t = 1, \ldots, T$:

$$\mathbf{m}_{ij}^{(t)} = \mathbf{M}_t\left(\mathbf{h}_i^{(t-1)}, \mathbf{h}_j^{(t-1)}, \mathbf{h}_{ij}^{(t-1)}\right)$$
$$\mathbf{h}_i^{(t)} = \left(\mathbf{v}_i, \mathbf{F}_t\left(\mathbf{h}_i^{(t-1)}, \mathbf{m}_i^{(t)}\right)\right), \ \mathbf{m}_i^{(t)} = \mathrm{Pool}\left(\{\mathbf{m}_{ij}^{(t)} | j \neq i\}\right)$$
$$\mathbf{h}_{ij}^{(t)} = \left(\mathbf{x}_{ij}, \mathbf{G}_t\left(\mathbf{h}_{ij}^{(t-1)}, \mathbf{m}_{ij}^{(t)}\right)\right)$$

Determinant

Defined for $n \times n$ matrices

$$\det(A) = \sum_{\sigma \in S_n} \operatorname{sgn}(\sigma) \prod_{i=1}^{n} a_{i,\sigma(i)}$$

$$\det(A^T) = \det(A)$$

$$\det(A)\det(B) = \det(AB)$$

Pfaffian

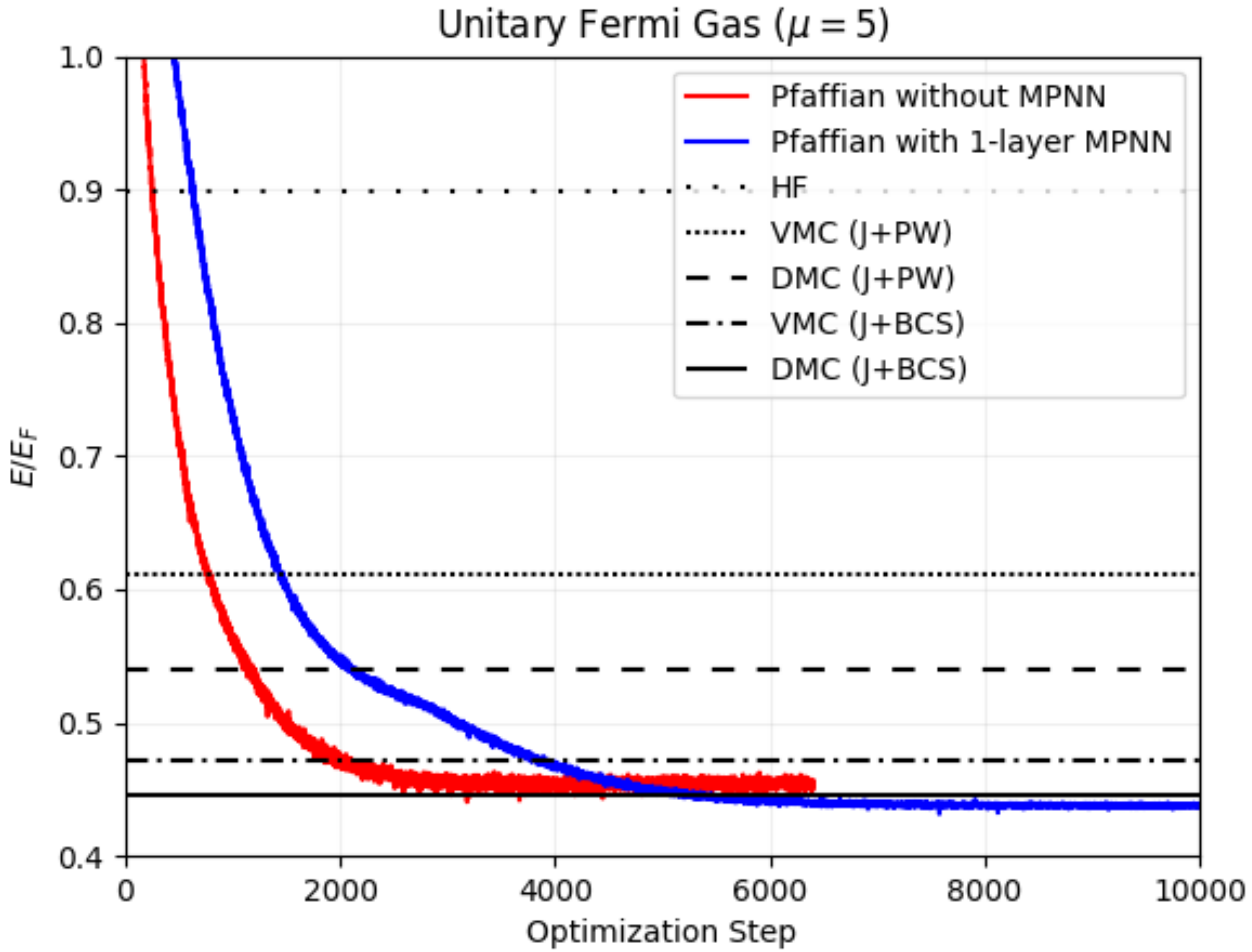
Defined for $2n \times 2n$ skew-symmetric matrices

$$\operatorname{pf}(A) = \frac{1}{2^n n!} \sum_{\sigma \in S_{2n}} \operatorname{sgn}(\sigma) \prod_{i=1}^{n} a_{\sigma(2i-1),\sigma(2i)}$$

$$\operatorname{pf}(A^T) = (-1)^n \operatorname{pf}(A)$$

$$\operatorname{pf}(A)\operatorname{pf}(B) = \exp\left(\frac{1}{2}\operatorname{tr}\log(A^T B)\right)$$

$$\det(A) = \operatorname{pf}(A)^2$$


Unitary Fermi Gas ($\mu = 5$)
Pfaffian without MPNN
Pfaffian with 1-layer MPNN
HF
VMC (J+PW)
DMC (J+PW)
VMC (J+BCS)
DMC (J+BCS)
E/E_F
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0
2000
4000
6000
8000
10000
Optimization Step