

# **Quantum Many-Body Simulations of Double Dot System**

---

Alocias Mariadason

# Contents

---

1. Introduction
2. Methods
3. Wavefunction
4. Implementation
5. Results
6. Summary and Conclusion

# Introduction

---

## 1. Small semiconductor nanostructures

# Quantum-Dot Model

---

- Schrödinger equation

# Quantum-Dot Model

---

- Schrödinger equation
  - $\mathcal{H}|\psi\rangle = E|\psi\rangle$

# Quantum-Dot Model

---

- Schrödinger equation
  - $\mathcal{H}|\psi\rangle = E|\psi\rangle$
- Hamiltonian

# Quantum-Dot Model

---

- Schrödinger equation
  - $\mathcal{H}|\psi\rangle = E|\psi\rangle$
- Hamiltonian
  - $$\mathcal{H} = -\frac{1}{2} \sum_i \nabla_i^2 + \sum_{i < j} f(\mathbf{r}_i, \mathbf{r}_j) - \frac{1}{2} \sum_k \frac{\nabla_k^2}{M_k} + \sum_{k < l} g(\mathbf{R}_k, \mathbf{R}_l) + V(\mathbf{R}, \mathbf{r})$$

# Quantum-Dot Model

---

- Schrödinger equation
  - $\mathcal{H}|\psi\rangle = E|\psi\rangle$
- Hamiltonian
  - $$\mathcal{H} = -\frac{1}{2} \sum_i \nabla_i^2 + \sum_{i < j} f(\mathbf{r}_i, \mathbf{r}_j) - \frac{1}{2} \sum_k \frac{\nabla_k^2}{M_k} + \sum_{k < l} g(\mathbf{R}_k, \mathbf{R}_l) + V(\mathbf{R}, \mathbf{r})$$
- Born-Oppenheimer Approximation

# Quantum-Dot Model

---

- Schrödinger equation
  - $\mathcal{H}|\psi\rangle = E|\psi\rangle$
- Hamiltonian
  - $$\mathcal{H} = -\frac{1}{2} \sum_i \nabla_i^2 + \sum_{i < j} f(\mathbf{r}_i, \mathbf{r}_j) - \frac{1}{2} \sum_k \frac{\nabla_k^2}{M_k} + \sum_{k < l} g(\mathbf{R}_k, \mathbf{R}_l) + V(\mathbf{R}, \mathbf{r})$$
- Born-Oppenheimer Approximation
  - Ignore Nuclei

# Quantum-Dot Model

---

- Schrödinger equation
  - $\mathcal{H}|\psi\rangle = E|\psi\rangle$
- Hamiltonian
  - $$\mathcal{H} = -\frac{1}{2} \sum_i \nabla_i^2 + \sum_{i < j} f(\mathbf{r}_i, \mathbf{r}_j) - \frac{1}{2} \sum_k \frac{\nabla_k^2}{M_k} + \sum_{k < l} g(\mathbf{R}_k, \mathbf{R}_l) + V(\mathbf{R}, \mathbf{r})$$
- Born-Oppenheimer Approximation
  - Ignore Nuclei

# Quantum-Dot Model

---

- Schrödinger equation
  - $\mathcal{H}|\psi\rangle = E|\psi\rangle$
- Hamiltonian
  - $$\mathcal{H} = -\frac{1}{2} \sum_i \nabla_i^2 + \sum_{i < j} f(\mathbf{r}_i, \mathbf{r}_j) - \frac{1}{2} \sum_k \frac{\nabla_k^2}{M_k} + \sum_{k < l} g(R_k, R_l) + V(\mathbf{R}, \mathbf{r})$$
- Born-Oppenheimer Approximation
  - Ignore Nuclei

# Quantum-Dot Model

---

- Interaction:

# Quantum-Dot Model

---

- Interaction:

- $f(\mathbf{r}_i, \mathbf{r}_j) = \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}$ , Coulomb Repulsion

# Quantum-Dot Model

- Interaction:
  - $f(r_i, r_j) = \frac{1}{|r_i - r_j|}$ , Coulomb Repulsion
- Confinement: Harmonic Oscillator<sup>1</sup>, Double-Well<sup>2</sup>

---

<sup>1</sup>S. Kvaal. "Harmonic Oscillator Eigenfunction Expansions, Quantum dots, and Effective Interactions". In: *Phys. Rev. B* (2009).

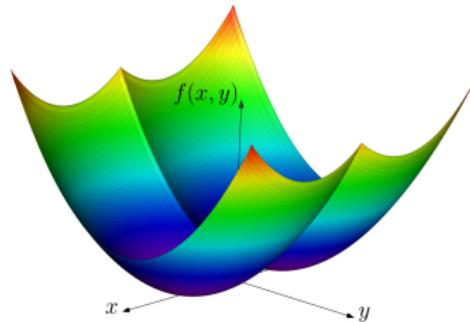
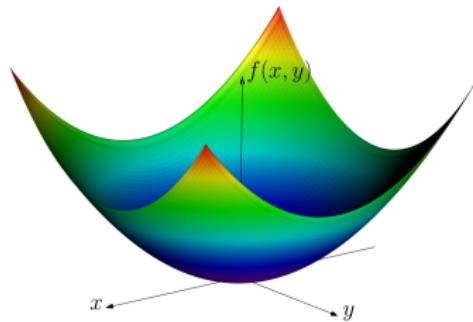
<sup>2</sup>M. J. A. Schuetz et al. "Nuclear Spin Dynamics in Double Quantum Dots: Multistability, Dynamical Polarization, Criticality, and Entanglement". In: *Phys. Rev. B* (2014).

# Quantum-Dot Model

- Interaction:
  - $f(r_i, r_j) = \frac{1}{|r_i - r_j|}$ , Coulomb Repulsion
- Confinement: Harmonic Oscillator<sup>1</sup>, Double-Well<sup>2</sup>

$$V(\mathbf{r}) = \frac{1}{2} m\omega^2 r^2$$

$$V(\mathbf{r}) = \frac{1}{2} m\omega^2 (r^2 - \delta R|x| + R^2)$$



<sup>1</sup>S. Kvaal. "Harmonic Oscillator Eigenfunction Expansions, Quantum dots, and Effective Interactions". In: *Phys. Rev. B* (2009).

<sup>2</sup>M. J. A. Schuetz et al. "Nuclear Spin Dynamics in Double Quantum Dots: Multistability, Dynamical Polarization, Criticality, and Entanglement". In: *Phys. Rev. B* (2014).

## Methods

---

## Methods

---

**Hartree-Fock  
Variational Monte-Carlo**

## Methods: Variational Principle

---

$$E_0 \leq \frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle}$$

# **Slater Determinant and Energy Functional**

## Methods: Slater Determinant and Energy Functional

---

- Pauli Principle

## Methods: Slater Determinant and Energy Functional

---

- Pauli Principle
- Slater Determinant

## Methods: Slater Determinant and Energy Functional

---

- Pauli Principle
- Slater Determinant

- $\Psi_T^{AS} = \frac{1}{\sqrt{N!}} \sum_P (-1)^P \mathcal{P}_P \prod_i \psi_i$

## Methods: Slater Determinant and Energy Functional

---

- Pauli Principle
- Slater Determinant

- $\Psi_T^{AS} = \frac{1}{\sqrt{N!}} \sum_P (-1)^P \mathcal{P}_P \prod_i \psi_i$

- $\Psi_T^S = \sqrt{\frac{\prod_{i=1}^N n_i!}{N!}} \sum_P \mathcal{P}_P \prod_i \psi_i$

# Methods: Slater Determinant and Energy Functional

- Pauli Principle
- Slater Determinant

$$\bullet \Psi_T^{AS} = \frac{1}{\sqrt{N!}} \sum_P (-1)^P \mathcal{P}_P \prod_i \psi_i$$

$$\bullet \Psi_T^S = \sqrt{\frac{\prod_{i=1}^N n_i!}{N!}} \sum_P \mathcal{P}_P \prod_i \psi_i$$

$$\bullet E[\Psi] = \frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} = \sum_p \langle p | \mathcal{H}_0 | p \rangle + \frac{1}{2} \sum_{p,q} [ \langle pq | f_{12} | pq \rangle \pm \langle pq | f_{12} | qp \rangle ]$$



# Hartree-Fock

## Methods: Hartree-Fock

---

- Assumptions

## Methods: Hartree-Fock

---

- Assumptions
  - The Born-Oppenheimer approximation holds.

## Methods: Hartree-Fock

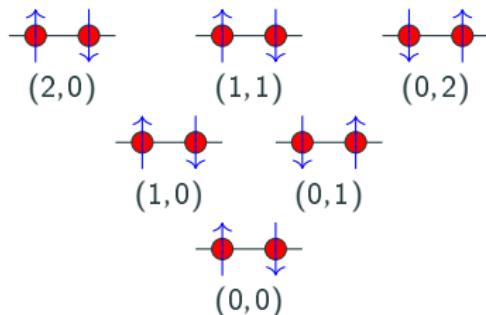
---

- Assumptions
  - The Born-Oppenheimer approximation holds.
  - All relativistic effects are negligible.

# Methods: Hartree-Fock

- Assumptions

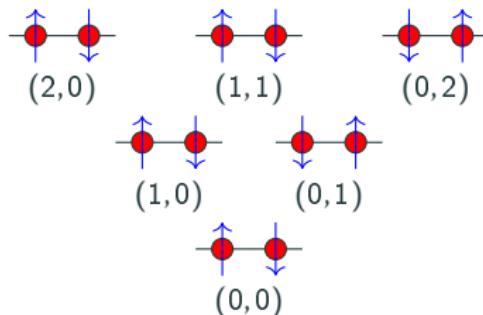
- The Born-Oppenheimer approximation holds.
- All relativistic effects are negligible.
- The wavefunction can be described by a single *Slater determinant*.



# Methods: Hartree-Fock

- Assumptions

- The Born-Oppenheimer approximation holds.
- All relativistic effects are negligible.
- The wavefunction can be described by a single *Slater determinant*.
- The Mean Field Approximation holds.



## Methods: Hartree-Fock

---

- Constrained minimization

## Methods: Hartree-Fock

---

- Constrained minimization
  - Spin orthogonality:  $\langle \psi_i | \psi_j \rangle = \delta_{ij}$

## Methods: Hartree-Fock

---

- Constrained minimization
  - Spin orthogonality:  $\langle \psi_i | \psi_j \rangle = \delta_{ij}$
  - Lagrange Multiplier method

# Methods: Hartree-Fock

---

- Constrained minimization
  - Spin orthogonality:  $\langle \psi_i | \psi_j \rangle = \delta_{ij}$
  - Lagrange Multiplier method
  - Fock-operator:  $\mathcal{F} \equiv \mathcal{H}_0 + \mathcal{J} \pm \mathcal{K}$ 
    - $\mathcal{J} \equiv \langle \psi_k^* | f_{12} | \psi_k \rangle = \int \psi_k^*(\mathbf{r}) f_{12} \psi_k(\mathbf{r}) d\mathbf{r}$
    - $\mathcal{K} \equiv \langle \psi_k^* | f_{12} | \psi \rangle = \int \psi_k^*(\mathbf{r}) f_{12} \psi(\mathbf{r}) d\mathbf{r}$

# Methods: Hartree-Fock

---

- Constrained minimization
  - Spin orthogonality:  $\langle \psi_i | \psi_j \rangle = \delta_{ij}$
  - Lagrange Multiplier method
  - Fock-operator:  $\mathcal{F} \equiv \mathcal{H}_0 + \mathcal{J} \pm \mathcal{K}$ 
    - $\mathcal{J} \equiv \langle \psi_k^* | f_{12} | \psi_k \rangle = \int \psi_k^*(\mathbf{r}) f_{12} \psi_k(\mathbf{r}) d\mathbf{r}$
    - $\mathcal{K} \equiv \langle \psi_k^* | f_{12} | \psi \rangle = \int \psi_k^*(\mathbf{r}) f_{12} \psi(\mathbf{r}) d\mathbf{r}$
    - $\mathcal{F} |\psi\rangle = \boldsymbol{\varepsilon} |\psi\rangle, \boldsymbol{\varepsilon} = (\varepsilon_0, \dots, \varepsilon_N)$

$$\mathcal{F} \left| \psi \right\rangle = \pmb{\varepsilon} \left| \psi \right\rangle, \quad \pmb{\varepsilon} = (\varepsilon_0, \dots, \varepsilon_N)$$

## Methods: Hartree-Fock

---

- Integrate out spin

## Methods: Hartree-Fock

---

- Integrate out spin
- Pair spins as:  $\{\psi_{2l-1}, \psi_{2l}\} = \{\phi_l(\mathbf{r})\alpha(s), \phi_l(\mathbf{r})\beta(s)\}$

## Methods: Hartree-Fock

---

- Integrate out spin
- Pair spins as:  $\{\psi_{2I-1}, \psi_{2I}\} = \{\phi_I(\mathbf{r})\alpha(s), \phi_I(\mathbf{r})\beta(s)\}$
- Expand:  $\phi_i(\mathbf{r}) = \sum_{p=1}^L C_{pi} \chi_p(\mathbf{r})$

## Methods: Hartree-Fock

---

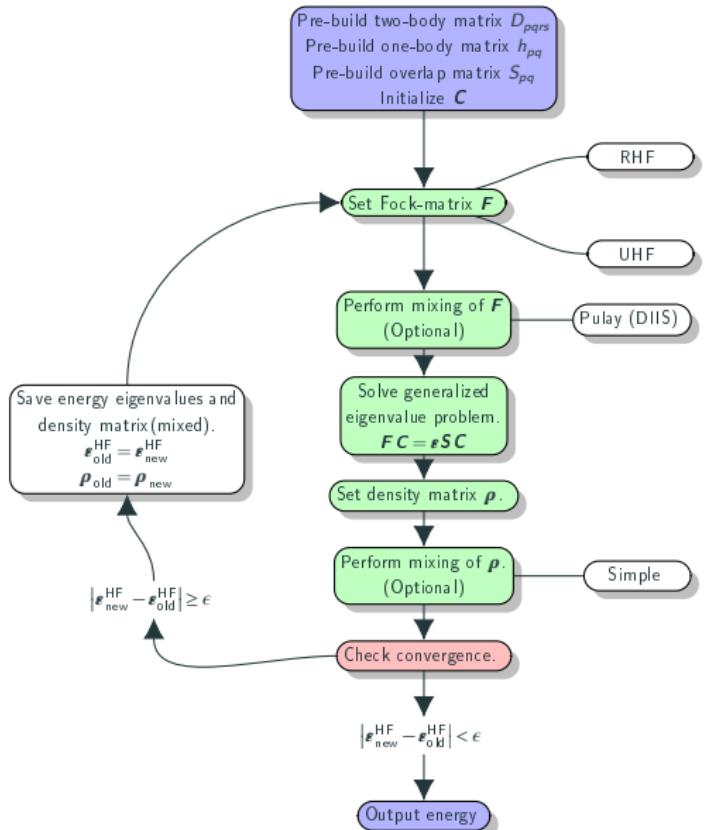
- Integrate out spin
- Pair spins as:  $\{\psi_{2I-1}, \psi_{2I}\} = \{\phi_I(\mathbf{r})\alpha(s), \phi_I(\mathbf{r})\beta(s)\}$
- Expand:  $\phi_i(\mathbf{r}) = \sum_{p=1}^L C_{pi} \chi_p(\mathbf{r})$
- Roothan-Hall:  $\mathbf{FC}_i = \epsilon S \mathbf{C}_i$ 
  - $F_{pq} = h_{pq} + \sum_{pq} \rho_{pq} (2D_{prqs} \pm D_{prsq})$
  - $h_{pq} \equiv \langle p | h | q \rangle$
  - $\rho_{pq} \equiv \sum_{i=1}^{\frac{N}{2}} C_{pi} C_{qi}^*$
  - $D_{pqrs} \equiv \langle pq | f_{12} | rs \rangle$
  - $S_{pq} \equiv \langle p | q \rangle$

## Methods: Hartree-Fock

---

- Integrate out spin
- Pair spins as:  $\{\psi_{2I-1}, \psi_{2I}\} = \{\phi_I(\mathbf{r})\alpha(s), \phi_I(\mathbf{r})\beta(s)\}$
- Expand:  $\phi_i(\mathbf{r}) = \sum_{p=1}^L C_{pi} \chi_p(\mathbf{r})$
- Roothan-Hall:  $\mathbf{F}\mathbf{C}_i = \boldsymbol{\epsilon} \mathbf{S} \mathbf{C}_i$ 
  - $F_{pq} = h_{pq} + \sum_{pq} \rho_{pq} (2D_{prqs} \pm D_{prsq})$
  - $h_{pq} \equiv \langle p | h | q \rangle$
  - $\rho_{pq} \equiv \sum_{i=1}^{\frac{N}{2}} C_{pi} C_{qi}^*$
  - $D_{pqrs} \equiv \langle pq | f_{12} | rs \rangle$
  - $S_{pq} \equiv \langle p | q \rangle$
- Poople-Nesbet:  $\mathbf{F}^+ \mathbf{C}^+ = \boldsymbol{\epsilon} \mathbf{S} \mathbf{C}^+, \mathbf{F}^- \mathbf{C}^- = \boldsymbol{\epsilon}^- \mathbf{S} \mathbf{C}^-$ 
  - $F_{pq}^\pm = h_{pq} + \sum_{k_\pm} \sum_{rs} C_{rk_\pm}^{\pm\dagger} C_{sk_\pm}^{\pm\dagger} [D_{prqs} - D_{prsq}] + \sum_{k_\mp} \sum_{rs} C_{rk_\mp}^{\mp\dagger} C_{sk_\mp}^{\mp\dagger} D_{prqs}$

# Methods: Hartree-Fock



# Variational Monte-Carlo

## Methods: Variational Monte-Carlo

---

- Variational Principle

## Methods: Variational Monte-Carlo

---

- Variational Principle
- Rewrite expectation value:  $\frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} = \int \frac{\Psi^* \mathcal{H} \Psi}{\int \Psi^* \Psi dr} dr = \int \frac{|\Psi|^2 E_L}{\int \Psi^* \Psi dr} dr$

## Methods: Variational Monte-Carlo

---

- Variational Principle
- Rewrite expectation value:  $\frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} = \int \frac{\Psi^* \mathcal{H} \Psi}{\int \Psi^* \Psi dr} dr = \int \frac{|\Psi|^2 E_L}{\int \Psi^* \Psi dr} dr$ 
  - $E_L(R; \alpha) \equiv \frac{1}{\Psi} \mathcal{H} \Psi$

## Methods: Variational Monte-Carlo

---

- Variational Principle
- Rewrite expectation value:  $\frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} = \int \frac{\Psi^* \mathcal{H} \Psi}{\int \Psi^* \Psi dr} dr = \int \frac{|\Psi|^2 E_L}{\int |\Psi|^2 dr} dr$ 
  - $E_L(R; \alpha) \equiv \frac{1}{\Psi} \mathcal{H} \Psi$
  - $P(R) \equiv \frac{|\Psi|^2}{\langle \Psi | \Psi \rangle}$

## Methods: Variational Monte-Carlo

---

- Variational Principle
- Rewrite expectation value:  $\frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} = \int \frac{\Psi^* \mathcal{H} \Psi}{\int \Psi^* \Psi dr} dr = \int \frac{|\Psi|^2 E_L}{\int |\Psi|^2 dr} dr$ 
  - $E_L(R; \alpha) \equiv \frac{1}{\Psi} \mathcal{H} \Psi$
  - $P(R) \equiv \frac{|\Psi|^2}{\langle \Psi | \Psi \rangle}$
- Metropolis-Hastings Algorithm

## Methods: Variational Monte-Carlo

---

- Variational Principle
- Rewrite expectation value:  $\frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} = \int \frac{\Psi^* \mathcal{H} \Psi}{\int \Psi^* \Psi dr} dr = \int \frac{|\Psi|^2 E_L}{\int |\Psi|^2 dr} dr$ 
  - $E_L(R; \alpha) \equiv \frac{1}{\Psi} \mathcal{H} \Psi$
  - $P(R) \equiv \frac{|\Psi|^2}{\langle \Psi | \Psi \rangle}$
- Metropolis-Hastings Algorithm
  - $r^{\text{new}} = r^{\text{old}} + \Delta t \xi$

# Methods: Variational Monte-Carlo

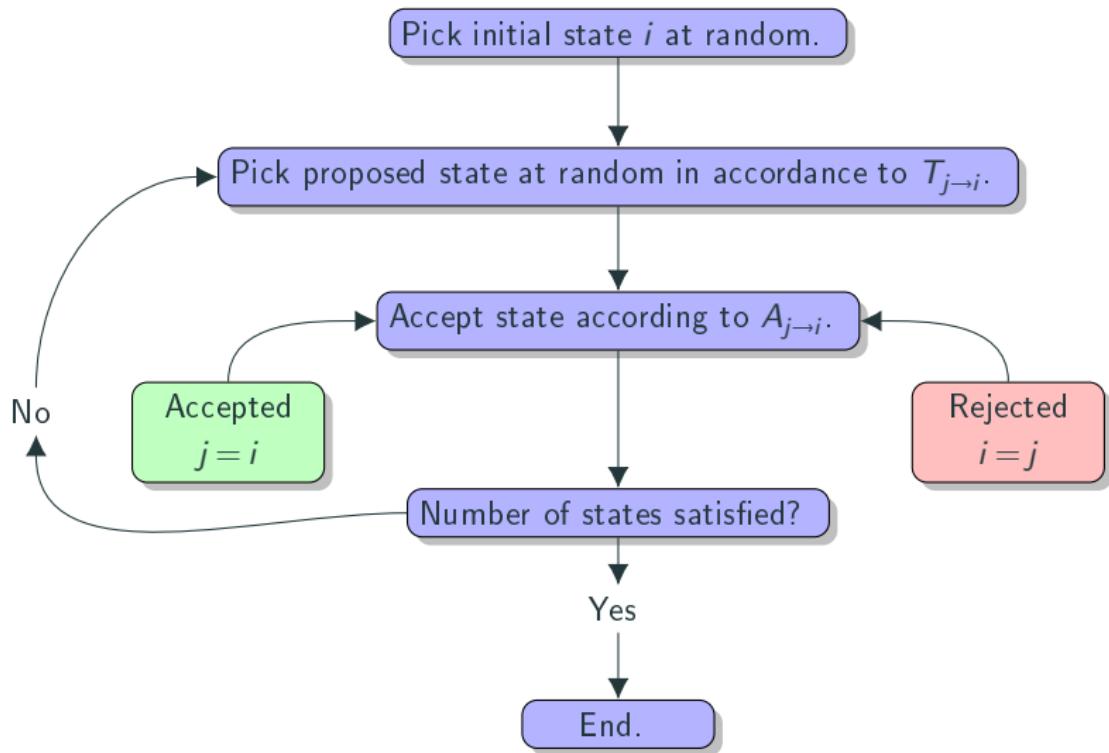
---

- Variational Principle
- Rewrite expectation value:  $\frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} = \int \frac{\Psi^* \mathcal{H} \Psi}{\int \Psi^* \Psi dr} dr = \int \frac{|\Psi|^2 E_L}{\int \Psi^* \Psi dr} dr$ 
  - $E_L(R; \alpha) \equiv \frac{1}{\Psi} \mathcal{H} \Psi$
  - $P(R) \equiv \frac{|\Psi|^2}{\langle \Psi | \Psi \rangle}$
- Metropolis-Hastings Algorithm
  - $r^{\text{new}} = r^{\text{old}} + \Delta t \xi$
  - $A_{i \rightarrow j} = \min\left(\frac{P_{i \rightarrow j}}{P_{j \rightarrow i}}, \frac{T_{i \rightarrow j}}{T_{j \rightarrow i}}, 1\right)$

# Methods: Variational Monte-Carlo

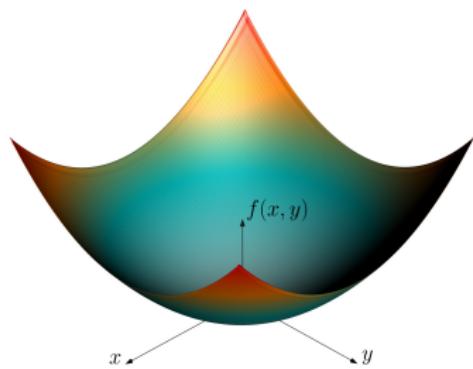
- Variational Principle
- Rewrite expectation value:  $\frac{\langle \Psi | \mathcal{H} | \Psi \rangle}{\langle \Psi | \Psi \rangle} = \int \frac{\Psi^* \mathcal{H} \Psi}{\int \Psi^* \Psi dr} dr = \int \frac{|\Psi|^2 E_L}{\int \Psi^* \Psi dr} dr$ 
  - $E_L(R; \alpha) \equiv \frac{1}{\Psi} \mathcal{H} \Psi$
  - $P(R) \equiv \frac{|\Psi|^2}{\langle \Psi | \Psi \rangle}$
- Metropolis-Hastings Algorithm
  - $r^{\text{new}} = r^{\text{old}} + \Delta t \xi$
  - $A_{i \rightarrow j} = \min\left(\frac{P_{i \rightarrow j}}{P_{j \rightarrow i}}, \frac{T_{i \rightarrow j}}{T_{j \rightarrow i}}, 1\right)$
  - Importance Sampling
    - $r^{\text{new}} = r^{\text{old}} + D \Delta t F^{\text{old}} + \sqrt{\Delta t} \xi$
    - $F = \frac{2}{\Psi} \nabla \Psi$
    - $\frac{T(b, a, \Delta t)}{T(a, b, \Delta t)} = \sum_i \exp\left(-\frac{(r_i^{(b)} - r_i^{(a)} - D \Delta t F_i^{(a)})^2}{4D \Delta t} + \frac{(r_i^{(a)} - r_i^{(b)} - D \Delta t F_i^{(b)})^2}{4D \Delta t}\right)$

## Methods: Variational Monte-Carlo

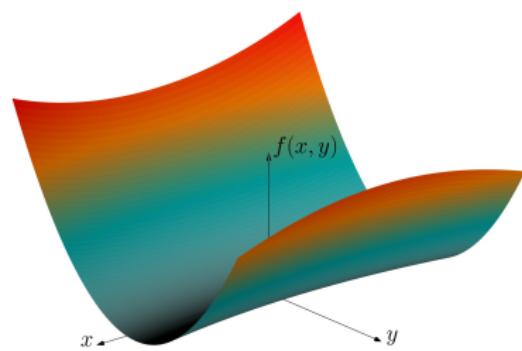


# Minimization

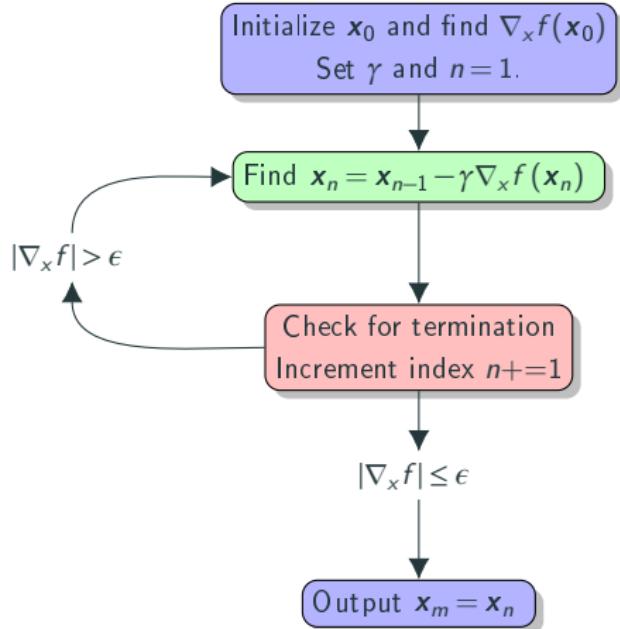
## Single-Well



## Rosenbrock



# Minimization: Gradient Descent



## Minimization: Gradient Descent

$x_0$	$\gamma$	Iterations	$x_m$	$f(x_m)$
(5, 5)	0.9	20	(-0.072, -0.072)	0.010
(5, 5)	0.9	50	( $-8.920 \times 10^{-5}$ , $-8.920 \times 10^{-5}$ )	$1.591 \times 10^{-8}$
(5, 5)	0.9	100	( $-1.273 \times 10^{-9}$ , $-1.273 \times 10^{-9}$ )	$3.242 \times 10^{-18}$
(5, 5)	0.5	20	(0.0, 0.0)	0.0
(5, 5)	0.5	50	(0.0, 0.0)	0.0
(5, 5)	0.5	100	(0.0, 0.0)	0.0
(5, 5)	0.1	20	(0.072, 0.072)	0.010
(5, 5)	0.1	50	( $8.920 \times 10^{-5}$ , $8.920 \times 10^{-5}$ )	$1.591 \times 10^{-8}$
(5, 5)	0.1	100	( $1.273 \times 10^{-9}$ , $1.273 \times 10^{-9}$ )	$3.242 \times 10^{-18}$

$x_0$	$\gamma$	Iterations	$x_m$	$f(x_m)$
(0, 0.5)	0.001	100	(0.181, 0.030)	0.034
(0, 0.5)	0.001	500	(0.512, 0.258)	0.327
(0, 0.5)	0.001	1000	(0.675, 0.454)	0.106
(0, 0.5)	0.001	100000	(1.000, 1.000)	0.0
(0, 0.5)	0.0001	100	(0.027, 0.068)	1.399
(0, 0.5)	0.0001	500	(0.105, 0.009)	0.801
(0, 0.5)	0.0001	1000	(0.184, 0.031)	0.666
(0, 0.5)	0.0001	100000	(0.994, 0.989)	$3.131 \times 10^{-5}$

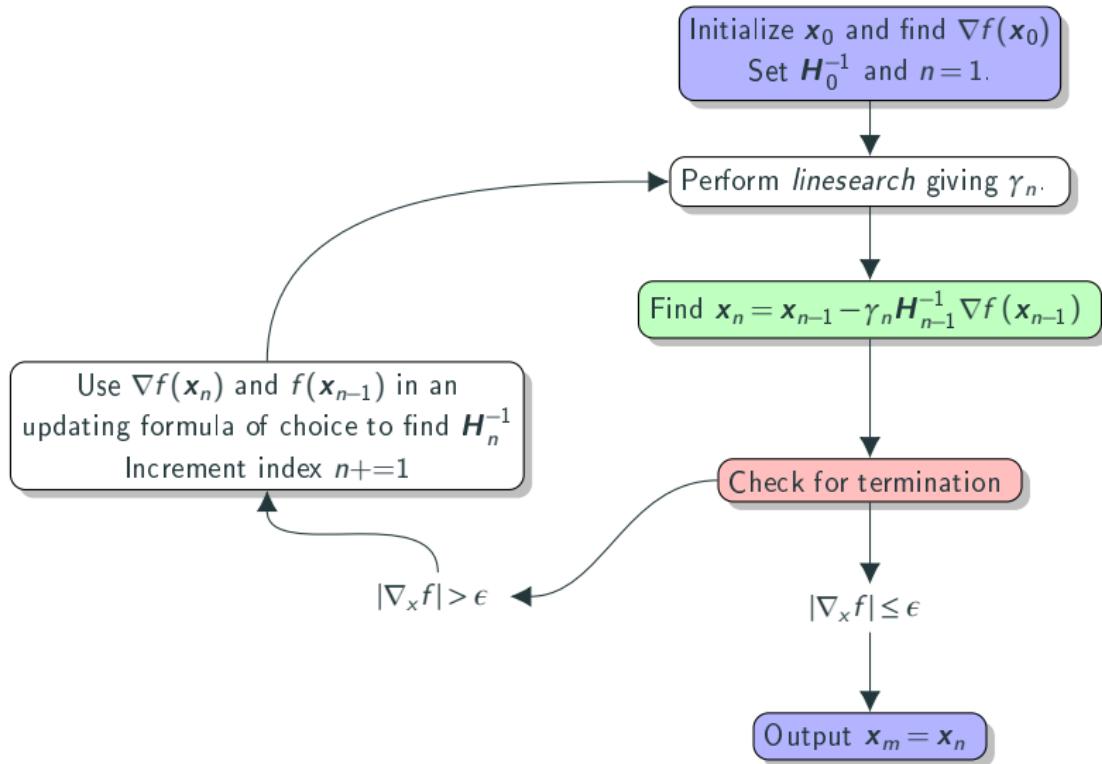
# Minimization: Gradient Descent

$x_0$	$\gamma$	Iterations	$x_m$	$f(x_m)$
(5, 5)	0.9	20	(-0.072, -0.072)	0.010
(5, 5)	0.9	50	( $-8.920 \times 10^{-5}$ , $-8.920 \times 10^{-5}$ )	$1.591 \times 10^{-8}$
(5, 5)	0.9	100	( $-1.273 \times 10^{-9}$ , $-1.273 \times 10^{-9}$ )	$3.242 \times 10^{-18}$
(5, 5)	0.5	20	(0.0, 0.0)	0.0
(5, 5)	0.5	50	(0.0, 0.0)	0.0
(5, 5)	0.5	100	(0.0, 0.0)	0.0
(5, 5)	0.1	20	(0.072, 0.072)	0.010
(5, 5)	0.1	50	( $8.920 \times 10^{-5}$ , $8.920 \times 10^{-5}$ )	$1.591 \times 10^{-8}$
(5, 5)	0.1	100	( $1.273 \times 10^{-9}$ , $1.273 \times 10^{-9}$ )	$3.242 \times 10^{-18}$

$x_0$	$\gamma$	Iterations	$x_m$	$f(x_m)$
(0, 0.5)	0.001	100	(0.181, 0.030)	0.034
(0, 0.5)	0.001	500	(0.512, 0.258)	0.327
(0, 0.5)	0.001	1000	(0.675, 0.454)	0.106
(0, 0.5)	0.001	100000	(1.000, 1.000)	0.0
(0, 0.5)	0.0001	100	(0.027, 0.068)	1.399
(0, 0.5)	0.0001	500	(0.105, 0.009)	0.801
(0, 0.5)	0.0001	1000	(0.184, 0.031)	0.666
(0, 0.5)	0.0001	100000	(0.994, 0.989)	$3.131 \times 10^{-5}$

# Minimization: Quasi-Newton BFGS



## Minimization: Quasi-Newton BFGS

$x_0$	Iterations	$x_m$	$f(x_m)$
(1,1)	1	(-0.071,-0.071)	1.000
(-1,2)	1	(0.447,-0.894)	1.000
(1,1)	2	(0.000,0.000)	0.000
(-1,2)	2	(0.000,0.000)	0.000
(10,10)	1	(-0.071,-0.071)	1.000
(10,10)	2	(0.000,0.000)	0.000
(100,100)	1	(-0.071,-0.071)	1.000
(100,100)	2	(0.000,0.000)	0.000

$x_0$	Iterations	$x_m$	$f(x_m)$
(-0.5,2.0)	1	(-0.706,0.708)	7.280
(-0.5,2.0)	2	(-0.780,0.649)	3.342
(-0.5,2.0)	10	(0.238,0.051)	0.584
(-0.5,2.0)	30	(1.000,1,000)	0.000
(5.5,-10.0)	1	(-0.996,0.091)	85.214
(5.5,-10.0)	2	(-0.908,1.087)	10.549
(5.5,-10.0)	10	(0.027,0.012)	0.9613
(5.5,-10.0)	30	(1.000,1,000)	0.000

## Minimization: Quasi-Newton BFGS

$x_0$	Iterations	$x_m$	$f(x_m)$
(1,1)	1	(-0.071,-0.071)	1.000
(-1,2)	1	(0.447,-0.894)	1.000
(1,1)	2	(0.000,0.000)	0.000
(-1,2)	2	(0.000,0.000)	0.000
(10,10)	1	(-0.071,-0.071)	1.000
(10,10)	2	(0.000,0.000)	0.000
(100,100)	1	(-0.071,-0.071)	1.000
(100,100)	2	(0.000,0.000)	0.000

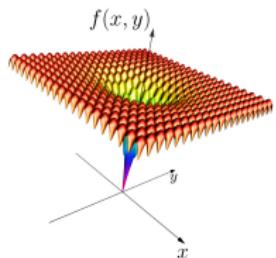
$x_0$	Iterations	$x_m$	$f(x_m)$
(-0.5,2.0)	1	(-0.706,0.708)	7.280
(-0.5,2.0)	2	(-0.780,0.649)	3.342
(-0.5,2.0)	10	(0.238,0.051)	0.584
(-0.5,2.0)	30	(1.000,1,000)	0.000
(5.5,-10.0)	1	(-0.996,0.091)	85.214
(5.5,-10.0)	2	(-0.908,1.087)	10.549
(5.5,-10.0)	10	(0.027,0.012)	0.9613
(5.5,-10.0)	30	(1.000,1,000)	0.000

## Minimization: Simulated Annealing

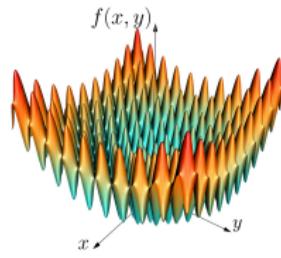
---

# Minimization: Simulated Annealing

Ackley



Rastrigin



# Wavefunction

---

## Wavefunction

---

$$\phi_i(\mathbf{r}) = \sum_{p=1}^L C_{pi} \chi_p(\mathbf{r})$$

## Wavefunction

---

$$\phi_i(\mathbf{r}) = \sum_{p=1}^L C_{pi} \chi_p(\mathbf{r}) \quad \phi_i(\mathbf{r}) = \sum_{p=1}^L C_{pi} \chi_p(\mathbf{r})$$

## Wavefunction: Integral Elements

---

$$\langle \phi_i(\mathbf{r}) | \phi_j(\mathbf{r}) \rangle$$

$$\langle \phi_i(\mathbf{r}) | x_d^k | \phi_j(\mathbf{r}) \rangle$$

$$\langle \phi_i(\mathbf{r}) | \nabla^2 | \phi_j(\mathbf{r}) \rangle$$

$$\langle \phi_i(\mathbf{r}_1) \phi_j(\mathbf{r}_2) | f_{12} | \phi_k(\mathbf{r}_1) \phi_l(\mathbf{r}_2) \rangle$$

## Wavefunction: Single-Well

---

- Hermite Function:  $\psi_n(\mathbf{r}) \equiv \prod_d N_d H_{n_d}(\sqrt{\omega}x_d) \exp\left(-\frac{\omega}{2}x_d^2\right)$

## Wavefunction: Single-Well

- Hermite Function:  $\psi_n(\mathbf{r}) \equiv \prod_d N_d H_{n_d}(\sqrt{\omega}x_d) \exp\left(-\frac{\omega}{2}x_d^2\right)$
- Solution in polar<sup>3</sup>

---

<sup>3</sup>E. Anisimovas and A. Matulis. “Energy spectra of few-electron quantum dots”. In: *Journal of Physics: Condensed Matter* (1998), p. 601.

## Wavefunction: Single-Well

- Hermite Function:  $\psi_n(\mathbf{r}) \equiv \prod_d N_d H_{n_d}(\sqrt{\omega}x_d) \exp\left(-\frac{\omega}{2}x_d^2\right)$
- Solution in polar<sup>3</sup>
- $\psi_n(\mathbf{r}) = \prod_d N_d \sum_{l=1}^{n_d} C_{n_d l}^{\text{Hermite}} g_l\left(\frac{\omega}{2}, \mathbf{r}, \mathbf{0}\right)$

---

<sup>3</sup>E. Anisimovas and A. Matulis. “Energy spectra of few-electron quantum dots”. In: *Journal of Physics: Condensed Matter* (1998), p. 601.

# Wavefunction: Single-Well

- Hermite Function:  $\psi_n(\mathbf{r}) \equiv \prod_d N_d H_{n_d}(\sqrt{\omega}x_d) \exp\left(-\frac{\omega}{2}x_d^2\right)$
- Solution in polar<sup>3</sup>
- $\psi_n(\mathbf{r}) = \prod_d N_d \sum_{l=1}^{n_d} C_{n_d l}^{\text{Hermite}} g_l\left(\frac{\omega}{2}, \mathbf{r}, \mathbf{0}\right)$
- Solution in Cartesian<sup>4</sup>

$$\langle g_i(\mathbf{r}) | g_j(\mathbf{r}) \rangle$$

$$\langle g_i(\mathbf{r}) | x_d^k | g_j(\mathbf{r}) \rangle$$

$$\langle g_i(\mathbf{r}) | \nabla^2 | g_j(\mathbf{r}) \rangle$$

$$\langle g_i(\mathbf{r}_1) g_j(\mathbf{r}_2) | f_{12} | g_k(\mathbf{r}_1) g_l(\mathbf{r}_2) \rangle$$

---

<sup>3</sup>E. Anisimovas and A. Matulis. “Energy spectra of few-electron quantum dots”. In: *Journal of Physics: Condensed Matter* (1998), p. 601.

<sup>4</sup>J. Olsen T. Helgaker P. Jørgensen. *Molecular Electronic-Structure Theory*. Wiley, 2014. isbn: 978-0-47-196755-2. doi: 10.1002/9781119019572.

# Wavefunction: Single-Well Integral Elements

$$\begin{aligned}\langle \psi_i^{\text{HO}} | \psi_j^{\text{HO}} \rangle &= N_i \delta_{ij} \\ \langle \psi_i^{\text{HO}} | h^{\text{HO}} | \psi_j^{\text{HO}} \rangle &= N_i \epsilon_i^{\text{HO}} \delta_{ij}\end{aligned}$$

$$\langle \psi_i^{\text{HO}} \psi_j^{\text{HO}} | \frac{1}{r_{12}} | \psi_k^{\text{HO}} \psi_l^{\text{HO}} \rangle = \frac{aN_{ijkl}}{\sqrt{2\omega}} \sum_{tuvw}^{ijkl} H_{tuvw}^{ijkl} \sum_{pq}^{t+v,u+w} E_p^{tv} E_q^{uw} (-1)^q \xi_{p+q} \left( \frac{\omega}{2}, \mathbf{0} \right)$$

$$E_t^{i+1,j} = \frac{1}{2(\alpha + \beta)} E_{t-1}^{ij} - \frac{\beta}{\alpha + \beta} (A_x - B_x) E_t^{ij} + (t+1) E_{t+1}^{ij}$$

$$E_t^{i,j+1} = \frac{1}{2(\alpha + \beta)} E_{t-1}^{ij} - \frac{\alpha}{\alpha + \beta} (A_y - B_y) E_t^{ij} + (t+1) E_{t+1}^{ij}$$

$$E_0^{00} = K_{AB}$$

$$\xi_{t+1,u}^n = t \xi_{t-1,u}^{n+1} + X_{AB} \xi_{t,u}^{n+1}$$

$$\xi_{t,u+1}^n = u \xi_{t,u-1}^{n+1} + Y_{AB} \xi_{t,u}^{n+1}$$

$$\xi_{00}^n = \left( \frac{-2\alpha\beta}{\alpha + \beta} \right)^n \zeta_n \left( \frac{\alpha\beta}{\alpha + \beta} R_{AB}^2 \right)$$

$$\zeta_n(x) = \int_{-1}^1 \frac{u^{2n}}{\sqrt{1-u^2}} e^{-u^2 x} du$$

$$\xi_{t+1,u,v}^n = t \xi_{t-1,u,v}^{n+1} + X_{AB} \xi_{t,u,v}^{n+1}$$

$$\xi_{t,u+1,v}^n = u \xi_{t,u-1,v}^{n+1} + Y_{AB} \xi_{t,u,v}^{n+1}$$

$$\xi_{t,u,v+1}^n = u \xi_{t,u,v-1}^{n+1} + Y_{AB} \xi_{t,u,v}^{n+1}$$

$$\xi_{000}^n = (-2\alpha\beta)^n \zeta_n \left( \frac{\alpha\beta}{\alpha + \beta} R_{AB}^2 \right)$$

$$\zeta_n(x) = \int_{-1}^1 u^{2n} e^{-u^2 x} du$$

## Wavefunction: Double-Well

---

- Perturbation of harmonic oscillator:  $U^{\text{DW}}(r) = V^{\text{HO}}(r) + V_n^{\text{DW}}(r)$

## Wavefunction: Double-Well

---

- Perturbation of harmonic oscillator:  $U^{\text{DW}}(r) = V^{\text{HO}}(r) + V_n^{\text{DW}}(r)$
- Expand in HO-functions:  $\left| \psi_p^{\text{DW}} \right\rangle = \sum_l C_{lp}^{\text{DW}} \left| \psi_l^{\text{HO}} \right\rangle$

## Wavefunction: Double-Well

---

- Perturbation of harmonic oscillator:  $U^{\text{DW}}(r) = V^{\text{HO}}(r) + V_n^{\text{DW}}(r)$
- Expand in HO-functions:  $\left| \psi_p^{\text{DW}} \right\rangle = \sum_l C_{lp}^{\text{DW}} \left| \psi_l^{\text{HO}} \right\rangle$
- Eigenvalue equation:  $H^{\text{DW}} C^{\text{DW}} = \epsilon^{\text{DW}} C^{\text{DW}}$ 
  - $H_{ij}^{\text{DW}} = \epsilon_i^{\text{HO}} \delta_{ij} + \langle \psi_i^{\text{HO}} | V_n^{\text{DW}} | \psi_j^{\text{HO}} \rangle$

# Wavefunction: Double-Well

---

- Perturbation of harmonic oscillator:  $U^{\text{DW}}(r) = V^{\text{HO}}(r) + V_n^{\text{DW}}(r)$
- Expand in HO-functions:  $\left| \psi_p^{\text{DW}} \right\rangle = \sum_l C_{lp}^{\text{DW}} \left| \psi_l^{\text{HO}} \right\rangle$
- Eigenvalue equation:  $H^{\text{DW}} C^{\text{DW}} = \epsilon^{\text{DW}} C^{\text{DW}}$ 
  - $H_{ij}^{\text{DW}} = \epsilon_i^{\text{HO}} \delta_{ij} + \langle \psi_i^{\text{HO}} \left| V_n^{\text{DW}} \right| \psi_j^{\text{HO}} \rangle$
- Integral-Elements

$$\langle \psi_p^{\text{DW}} \left| \psi_q^{\text{DW}} \right\rangle = \delta_{pq}$$

$$\langle \psi_p^{\text{DW}} \left| h^{\text{DW}} \right| \psi_q^{\text{DW}} \rangle = \epsilon_p^{\text{DW}} \delta_{pq}$$

$$\left\langle \psi_p^{\text{DW}} \psi_q^{\text{DW}} \left| \frac{1}{r_{12}} \right| \psi_r^{\text{DW}} \psi_s^{\text{DW}} \right\rangle = \sum_{tuvw}^{ijkl} C_{tp}^{\text{DW}} C_{uq}^{\text{DW}} C_{vr}^{\text{DW}} C_{ws}^{\text{DW}} \left\langle \psi_t^{\text{HO}} \psi_u^{\text{HO}} \left| \frac{1}{r_{12}} \right| \psi_v^{\text{HO}} \psi_w^{\text{HO}} \right\rangle$$

## Wavefunction: Slater-Jastrow

---

- Slater determinant:  $\psi_T = \det(\Phi(\mathbf{R}; \boldsymbol{\alpha}))\xi(s)$

## Wavefunction: Slater-Jastrow

---

- Slater determinant:  $\psi_T = \det(\Phi(\mathbf{R}; \boldsymbol{\alpha}))\xi(s)$
- Modified Hermite:  $\Phi_{ij} = \psi_{n_j}^{\text{HO}}(\sqrt{\alpha\omega}r_i) = \prod_d N_d H_{n_d}(\sqrt{\alpha\omega}x_d) e^{-\frac{\alpha\omega}{2}x_d^2}$

## Wavefunction: Slater-Jastrow

---

- Slater determinant:  $\psi_T = \det(\Phi(\mathbf{R}; \boldsymbol{\alpha}))\xi(s)$
- Modified Hermite:  $\Phi_{ij} = \psi_{n_j}^{\text{HO}}(\sqrt{\alpha\omega}r_i) = \prod_d N_d H_{n_d}(\sqrt{\alpha\omega}x_d) e^{-\frac{\alpha\omega}{2}x_d^2}$
- Hartree-Fock:  $\Phi_{ij} = \sum_l C_{jl} \psi_{n_l}^{\text{HO}}(\sqrt{\omega}r_i)$

## Wavefunction: Slater-Jastrow

---

- Slater determinant:  $\psi_T = \det(\Phi(\mathbf{R}; \boldsymbol{\alpha}))\xi(s)$
- Modified Hermite:  $\Phi_{ij} = \psi_{n_j}^{\text{HO}}(\sqrt{\alpha\omega}r_i) = \prod_d N_d H_{n_d}(\sqrt{\alpha\omega}x_d)e^{-\frac{\alpha\omega}{2}x_d^2}$
- Hartree-Fock:  $\Phi_{ij} = \sum_l C_{jl} \psi_{n_l}^{\text{HO}}(\sqrt{\omega}r_i)$
- Modified Hartree-Fock:  $\Phi_{ij} = \sum_l C_{jl} \psi_{n_l}^{\text{HO}}(\sqrt{\alpha\omega}r_i)$

## Wavefunction: Slater-Jastrow

---

- Slater determinant:  $\psi_T = \det(\Phi(\mathbf{R}; \boldsymbol{\alpha}))\xi(s)$
- Modified Hermite:  $\Phi_{ij} = \psi_{n_j}^{\text{HO}}(\sqrt{\alpha\omega}r_i) = \prod_d N_d H_{n_d}(\sqrt{\alpha\omega}x_d)e^{-\frac{\alpha\omega}{2}x_d^2}$
- Hartree-Fock:  $\Phi_{ij} = \sum_l C_{jl} \psi_{n_l}^{\text{HO}}(\sqrt{\omega}r_i)$
- Modified Hartree-Fock:  $\Phi_{ij} = \sum_l C_{jl} \psi_{n_l}^{\text{HO}}(\sqrt{\alpha\omega}r_i)$
- Padé-Jastrow:  $J_{\text{Padé}} = \prod_{i < j} e^{\frac{a_{ij} r_{ij}}{1 + \beta r_{ij}}}$

## Wavefunction: Slater-Jastrow

---

- Slater determinant:  $\psi_T = \det(\Phi(\mathbf{R}; \boldsymbol{\alpha})) \xi(s)$
- Modified Hermite:  $\Phi_{ij} = \psi_{n_j}^{\text{HO}}(\sqrt{\alpha\omega}r_i) = \prod_d N_d H_{n_d}(\sqrt{\alpha\omega}x_d) e^{-\frac{\alpha\omega}{2}x_d^2}$
- Hartree-Fock:  $\Phi_{ij} = \sum_l C_{jl} \psi_{n_l}^{\text{HO}}(\sqrt{\omega}r_i)$
- Modified Hartree-Fock:  $\Phi_{ij} = \sum_l C_{jl} \psi_{n_l}^{\text{HO}}(\sqrt{\alpha\omega}r_i)$
- Padé-Jastrow:  $J_{\text{Padé}} = \prod_{i < j} e^{\frac{a_{ij} r_{ij}}{1 + \beta r_{ij}}}$
- NQS:  $J_{\text{NQS}} = e^{-\sum_{i=1}^N \frac{(r_i - a_i)^2}{2\sigma^2}} \prod_j^M \left( 1 + e^{b_j + \sum_{i=1}^N \sum_{d=1}^D \frac{x_i^{(d)} w_{i+d,j}}{\sigma^2}} \right)$

## Wavefunction: Slater-Jastrow

---

- Slater determinant:  $\psi_T = \det(\Phi(\mathbf{R}; \boldsymbol{\alpha})) \xi(s)$
- Modified Hermite:  $\Phi_{ij} = \psi_{n_j}^{\text{HO}}(\sqrt{\alpha\omega}r_i) = \prod_d N_d H_{n_d}(\sqrt{\alpha\omega}x_d) e^{-\frac{\alpha\omega}{2}x_d^2}$
- Hartree-Fock:  $\Phi_{ij} = \sum_l C_{jl} \psi_{n_l}^{\text{HO}}(\sqrt{\omega}r_i)$
- Modified Hartree-Fock:  $\Phi_{ij} = \sum_l C_{jl} \psi_{n_l}^{\text{HO}}(\sqrt{\alpha\omega}r_i)$
- Padé-Jastrow:  $J_{\text{Padé}} = \prod_{i < j} e^{\frac{a_{ij} r_{ij}}{1 + \beta r_{ij}}}$
- NQS:  $J_{\text{NQS}} = e^{-\sum_{i=1}^N \frac{(r_i - a_i)^2}{2\sigma^2}} \prod_j^M \left( 1 + e^{b_j + \sum_{i=1}^N \sum_{d=1}^D \frac{x_j^{(d)} w_{i+d,j}}{\sigma^2}} \right)$
- Padé-NQS:  $J = J_{\text{Padé}} J_{\text{NQS}}$

# Implementation

---

# Implementation

---

- C++ and Eigen

# Implementation

---

- C++ and Eigen
  - Performance

# Implementation

---

- C++ and Eigen
  - Performance
  - Generalization

# Implementation

---

- C++ and Eigen
  - Performance
  - Generalization
- Python

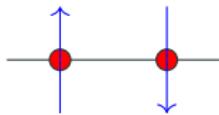
# Implementation

---

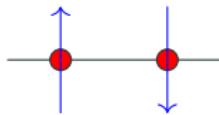
- C++ and Eigen
  - Performance
  - Generalization
- Python
  - Generate C++ code

## Implementation: Cartesian

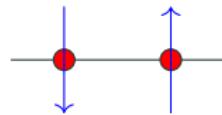
---



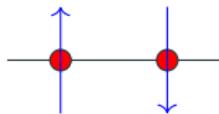
$(2, 0)$



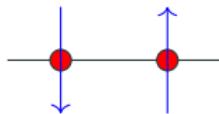
$(1, 1)$



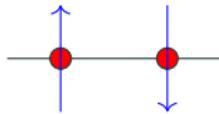
$(0, 2)$



$(1, 0)$

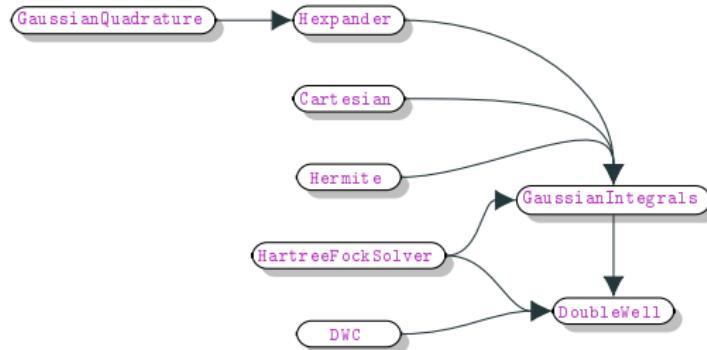


$(0, 1)$

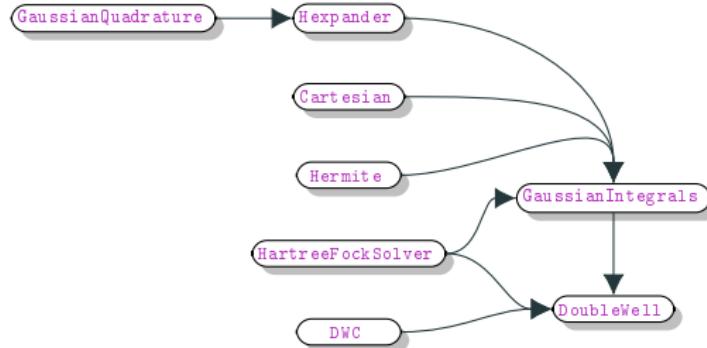


$(0, 0)$

# Implementation: Hartree-Fock

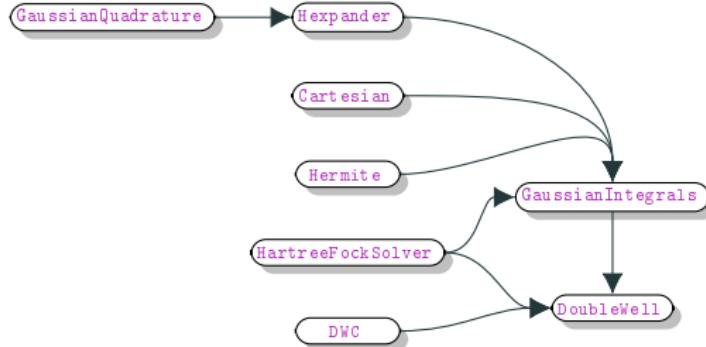


# Implementation: Hartree-Fock



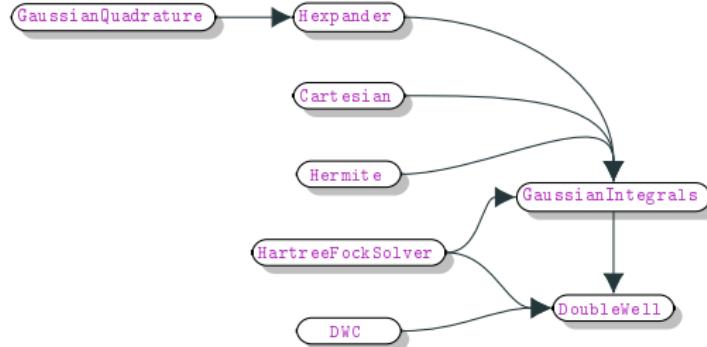
- Parallelization

# Implementation: Hartree-Fock



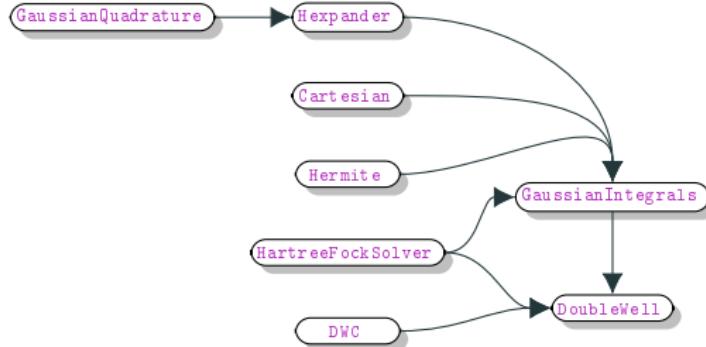
- Parallelization
  - Two-body element is computationally expensive
  - $S_i = \sum_{j=0}^{P_i} \prod_d (n_{jd} + 1)$

# Implementation: Hartree-Fock



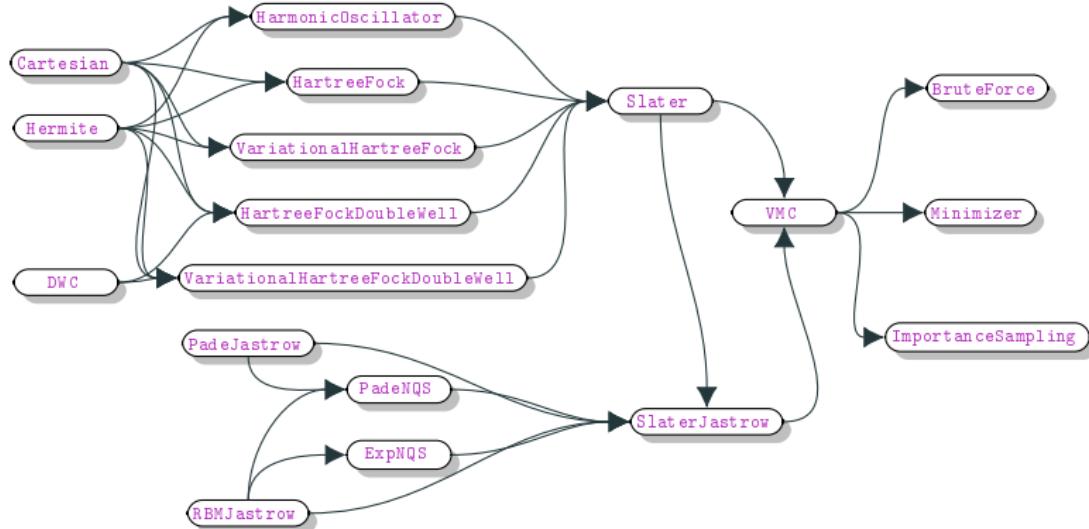
- Parallelization
  - Two-body element is computationally expensive
  - $S_i = \sum_{j=0}^{P_i} \prod_d (n_{jd} + 1)$
- Hartree-Fock algorithm only run on one process

# Implementation: Hartree-Fock

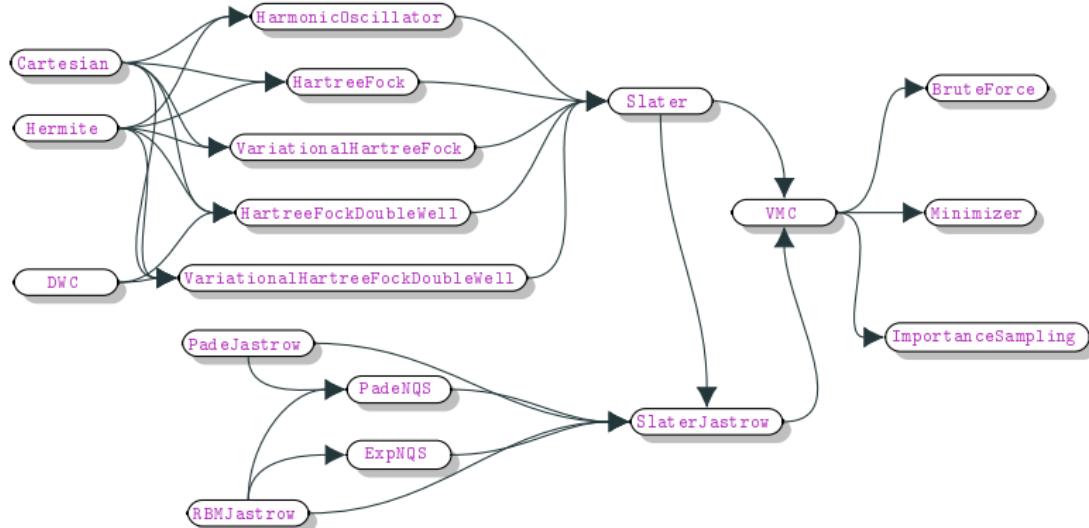


- Parallelization
  - Two-body element is computationally expensive
  - $S_i = \sum_{j=0}^{P_i} \prod_d (n_{jd} + 1)$
- Hartree-Fock algorithm only run on one process
- Tabulation of Two-Body matrix

# Implementation: Variational Monte-Carlo



# Implementation: Variational Monte-Carlo



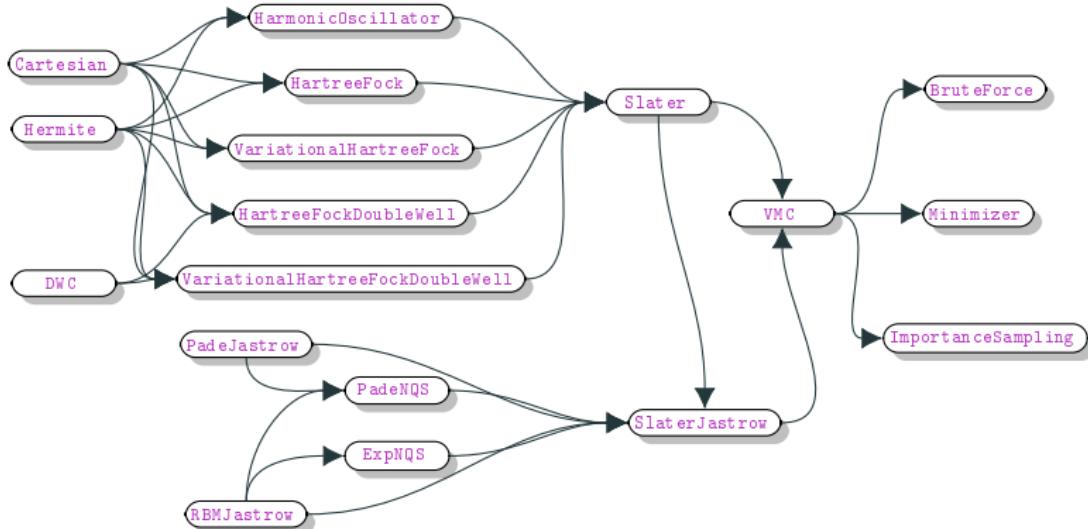
- Hermite generated with Python and SymPy

# Implementation: Variational Monte-Carlo

---

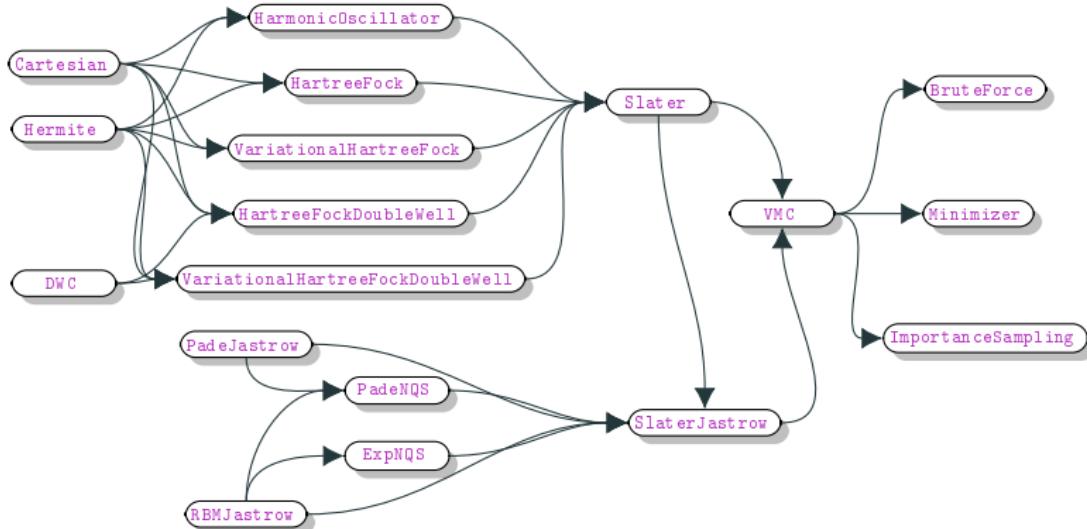
- `set`: Called during initialization (before each sampling)
- `reSetAll`: Sets all matrices to zero (used in testing)
- `initializeMatrices`: Allocate memory
- `update`: Update positions and wavefunction
- `reset`: Revert to previous positions and wavefunction
- `resetGradient`: Revert to previous gradient
- `acceptState`: Update previous positions and wavefunction to current
- `acceptGradient`: Update previous gradient to current one

# Implementation: Variational Monte-Carlo



- Hermite generated with Python and SymPy

# Implementation: Variational Monte-Carlo



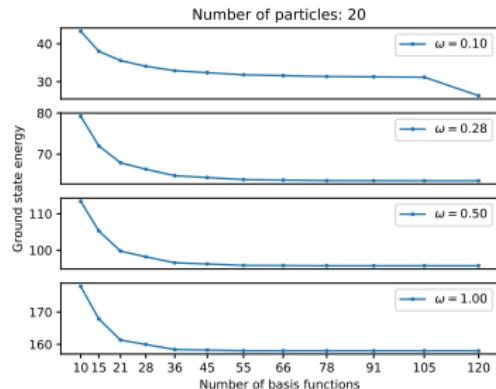
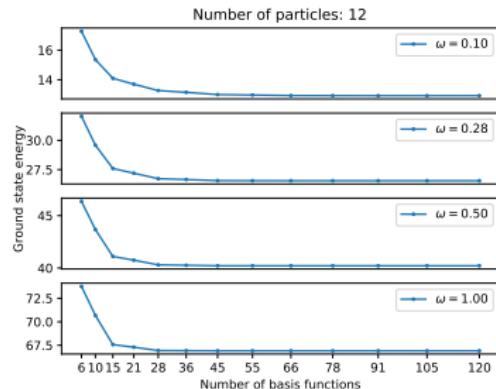
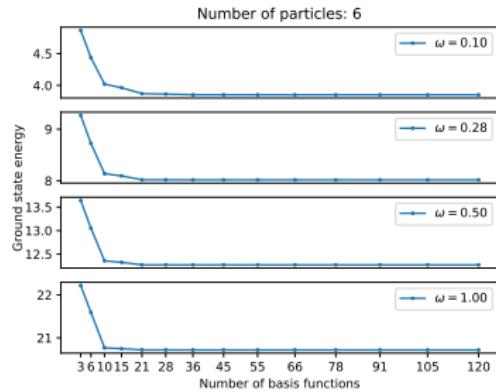
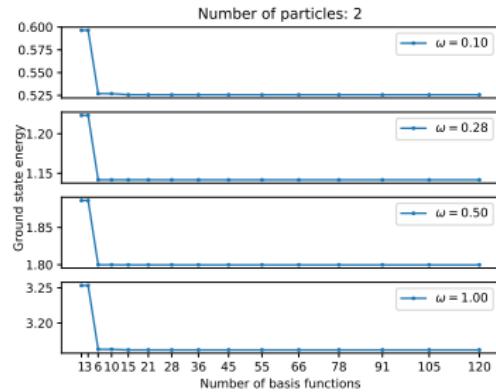
- Hermite generated with Python and SymPy
- Wavefunction class can be created with Python

# Results

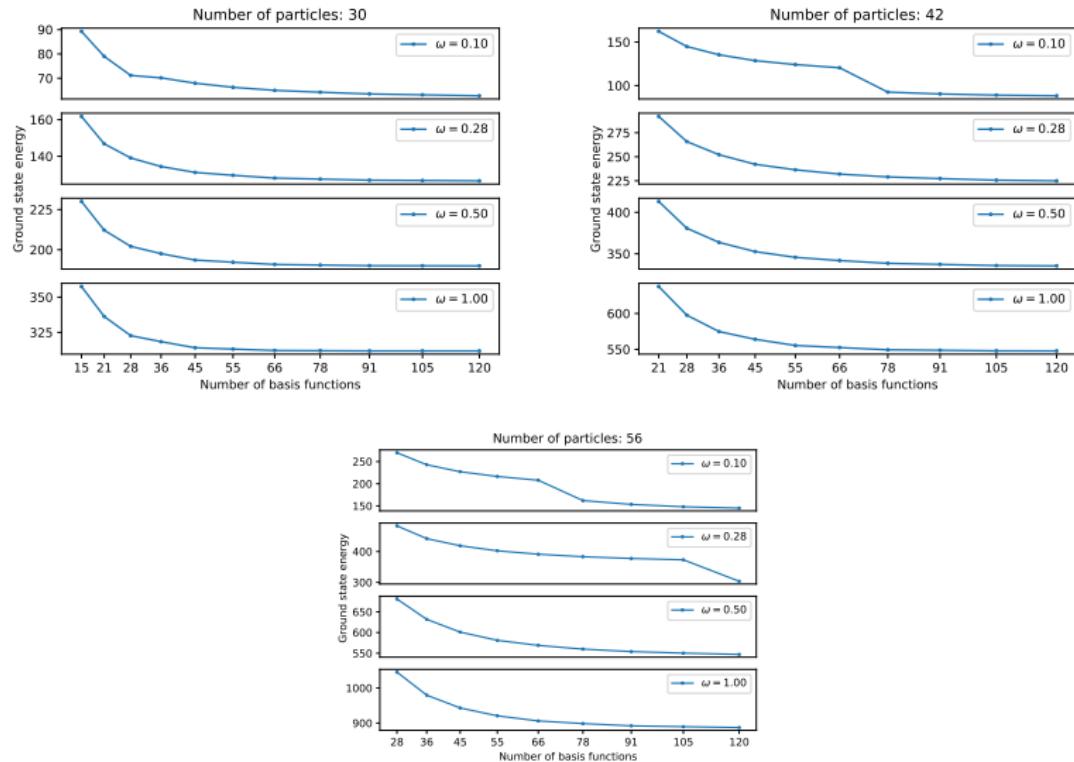
---

# Benchmark

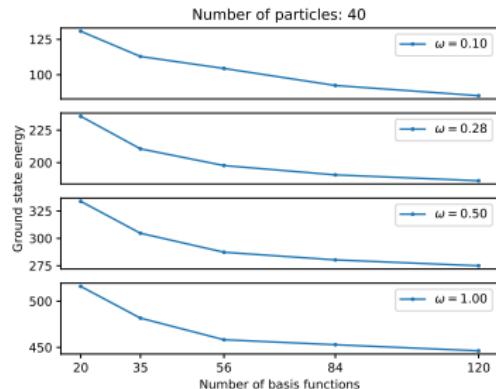
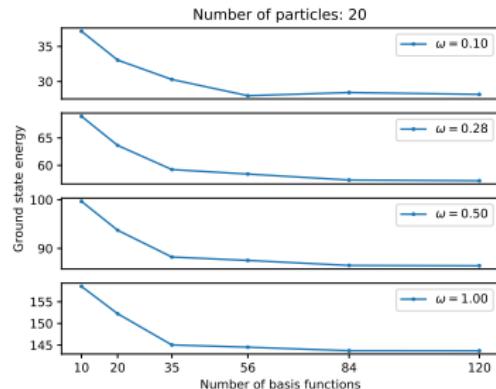
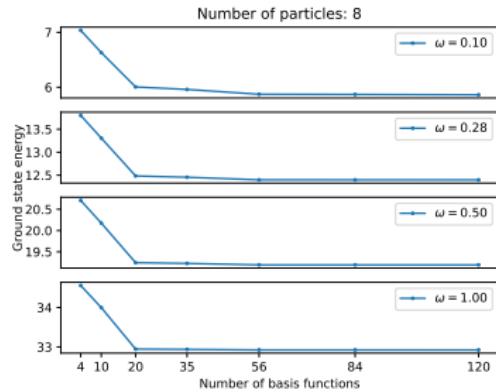
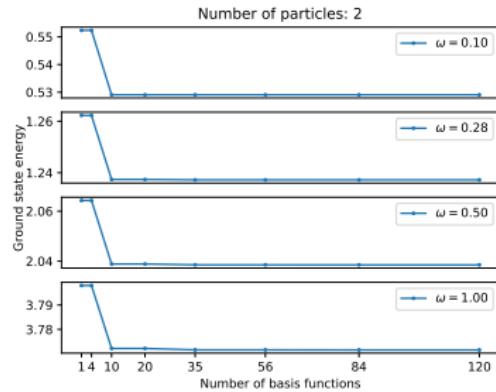
# Results: Benchmark



# Results: Benchmark



# Results: Benchmark



## Results: Benchmark

$\omega$ [a.u]	$N$			
	2	6	12	20
0.1	0.4407(4)	3.5650(4)	12.3164(4)	30.0480(4)
0.28	1.0020(4)	7.6198(4)	25.5948(3)	61.8090(3)
0.5	1.6650(4)	11.8017(4)	39.3166(3)	93.9240(2)
1.0	3.0000(5)	20.2863(3)	68.1465(3)	156.2778(2)

$\omega$ [a.u]	$N$	
	2	8
0.1	0.50006(5)	5.80479(4)
0.28	1.20156(5)	12.48178(4)
0.5	2.00027(5)	19.33356(4)
1.0	3.72985(5)	33.30958(4)

$$\psi = \psi^{\text{HO}}(\sqrt{\alpha\omega}) J_{\text{Pad\'e}}$$

## Results: Benchmark

$\omega$ [a.u]	$N$			
	2	6	12	20
0.1	0.46552(5){15}	3.70137(4){36}	12.64342(4){91}	-
0.28	1.04939(4){6}	7.89627(4){36}	26.21301(4){66}	62.93503(5){120}
0.5	1.70130(4){6}	12.02776(4){21}	39.76442(3){45}	95.21976(3){91}
1.0	3.05625(4){6}	20.45876(3){36}	66.37115(3){45}	157.41119(3){78}

$\omega$ [a.u]	$N$			
	2	6	12	20
0.10	0.44473(5){15}	3.63897(4){36}	12.46408(4){91}	-
0.28	1.04978(4){6}	7.72929(4){36}	25.96595(4){66}	62.65652(3){120}
0.50	1.66418(4){6}	11.97781(4){21}	39.57182(3){45}	94.76303(3){91}
1.00	3.00624(4){6}	20.38811(3){36}	66.28996(3){45}	157.46167(3){78}

$$\psi_p = \sum_l C_{lp} \psi_l^{\text{HO}}(\sqrt{\omega} r) J_{\text{Pad\'e}}, \quad \psi_p = \sum_l C_{lp} \psi_l^{\text{HO}}(\sqrt{\alpha\omega} r) J_{\text{Pad\'e}}$$

## Results: Benchmark

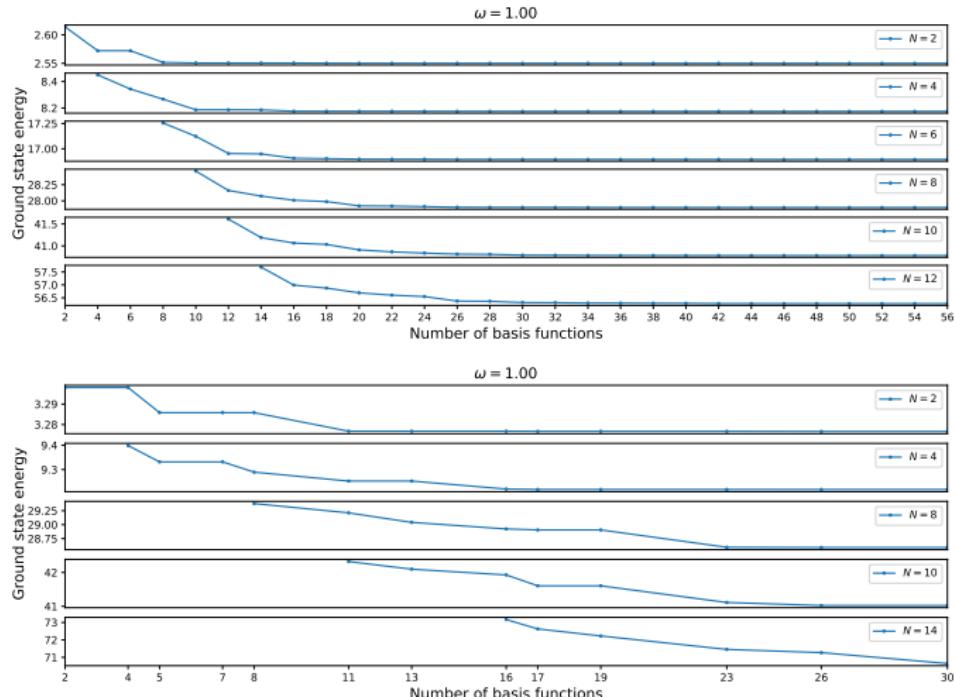
$\omega$	$N$	
	2	8
0.1	0.51122(5){70}	5.87372(4){120}
0.28	1.21844(5){70}	12.36177(4){168}
0.5	2.02030(4){20}	19.15006(4){112}
1.0	3.72918(5){20}	33.58046(4){168}

$\omega$	$N$	
	2	8
0.1	0.50751(5){70}	5.84082(4){240}
0.28	1.20320(5){20}	12.37435(4){168}
0.5	2.01439(4){20}	19.09917(4){112}
1.0	3.72959(5){70}	33.04162(4){168}

$$\psi_p = \sum_l C_{lp} \psi_l^{\text{HO}}(\sqrt{\omega} r) J_{\text{Pad\'e}}, \quad \psi_p = \sum_l C_{lp} \psi_l^{\text{HO}}(\sqrt{\alpha\omega} r) J_{\text{Pad\'e}}$$

# Results: Double-Well Hartree-Fock



## Results: Double-Well Variational Monte-Carlo

$\omega$	$N$			
	2	4	6	8
1.0	2.42238(4){10}	7.95247(4){42}	16.61419(4){44}	27.54453(3){50}

$$\psi_p = \sum_l C_{lp}^{\text{HF}} \sum_k C_{kl}^{\text{DW}} \psi_k^{\text{HO}} (\sqrt{\omega} r) J_{\text{Pad\'e}}$$

$\omega$	$N$			
	2	4	6	8
1.0	2.36618(4){10}	7.90232(4){42}	16.55609(4){44}	27.58524(4){50}

$$\psi_p = \sum_l C_{lp}^{\text{HF}} \sum_k C_{kl}^{\text{DW}} \psi_k^{\text{HO}} (\sqrt{\alpha \omega} r) J_{\text{Pad\'e}}$$

## Results: Double-Well Variational Monte-Carlo

$\omega$	$N$		
	2	4	8
1.0	3.25118(4){11}	9.17489(4){17}	28.49671(4){26}

$$\psi_p = \sum_l C_{lp}^{\text{HF}} \sum_k C_{kl}^{\text{DW}} \psi_k^{\text{HO}} (\sqrt{\omega} r) J_{\text{Pad\'e}}$$

$\omega$	$N$		
	2	4	8
1.0	3.22226(4){11}	9.17013(4){17}	28.62826(4){26}

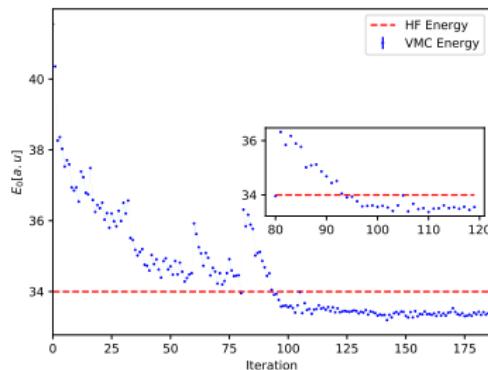
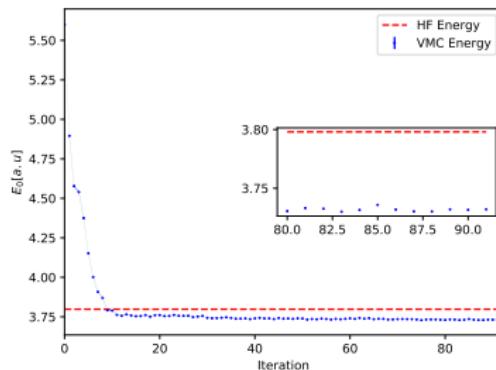
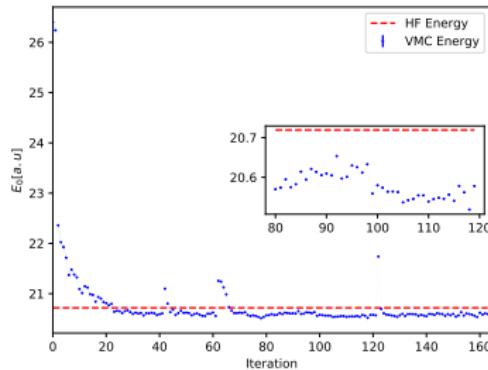
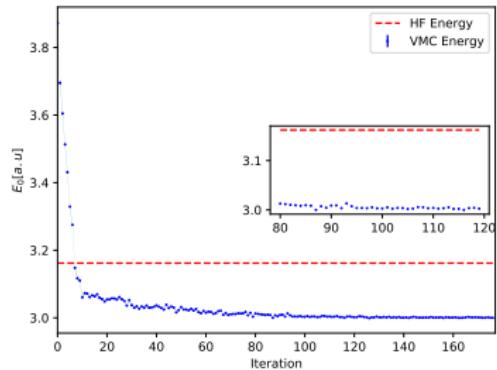
$$\psi_p = \sum_l C_{lp}^{\text{HF}} \sum_k C_{kl}^{\text{DW}} \psi_k^{\text{HO}} (\sqrt{\alpha\omega} r) J_{\text{Pad\'e}}$$

## Results: NQS-Jastrow

---

$$J_{\text{NQS}} = e^{-\sum_{i=1}^N \frac{(r_i - a_i)^2}{2\sigma^2}} \prod_j^M \left( 1 + e^{b_j + \sum_{i=1}^N \sum_{d=1}^D \frac{x_i^{(d)} w_{i+d,j}}{\sigma^2}} \right)$$

# Results: NQS-Jastrow Harmonic Oscillator



## Summary and Conclusion

---

**Questions?**

# Questions

Questions?