

Wave functions beyond Slater–Jastrow

~~Pablo Lopez Rios~~ Neil D. Drummond

University of ~~Cambridge~~ Lancaster

6th August 2014

The Slater determinant

- Basic antisymmetric wave function: a Slater determinant

$$\Psi_S(\mathbf{R}) = \begin{vmatrix} \phi_1(\mathbf{r}_1) & \phi_1(\mathbf{r}_2) & \dots & \phi_1(\mathbf{r}_N) \\ \phi_2(\mathbf{r}_1) & \phi_2(\mathbf{r}_2) & \dots & \phi_2(\mathbf{r}_N) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_N(\mathbf{r}_1) & \phi_N(\mathbf{r}_2) & \dots & \phi_N(\mathbf{r}_N) \end{vmatrix}$$

- If \hat{H} is independent of spin, can factor spatial wave function into up- and down-spin parts:

$$\begin{vmatrix} \phi_1(\mathbf{r}_1^\uparrow) & \dots & \phi_1(\mathbf{r}_{N_\uparrow}^\uparrow) \\ \vdots & \ddots & \vdots \\ \phi_{N_\uparrow}(\mathbf{r}_1^\uparrow) & \dots & \phi_{N_\uparrow}(\mathbf{r}_{N_\uparrow}^\uparrow) \end{vmatrix} \begin{vmatrix} \phi_1(\mathbf{r}_1^\downarrow) & \dots & \phi_1(\mathbf{r}_{N_\downarrow}^\downarrow) \\ \vdots & \ddots & \vdots \\ \phi_{N_\downarrow}(\mathbf{r}_1^\downarrow) & \dots & \phi_{N_\downarrow}(\mathbf{r}_{N_\downarrow}^\downarrow) \end{vmatrix}$$

- Orbitals $\{\phi\}$ can be obtained from HF, DFT, etc.

The Slater–Jastrow wave function

- Electronic correlation introduced using a multiplicative Jastrow factor,

$$\Psi_{\text{SJ}}(\mathbf{R}) = \exp[J(\mathbf{R})] \Psi_{\text{S}}(\mathbf{R})$$

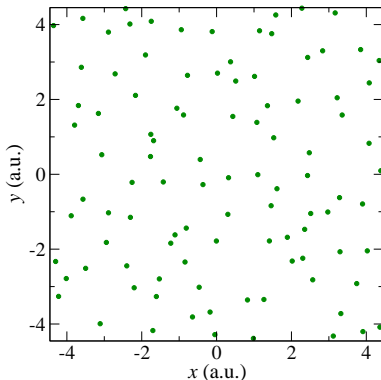
- Advantages of using a Jastrow factor:
 - **Compact** form
 - Fulfills **Kato cusp conditions** → much improved statistics
 - Good description of electronic **correlation** → ability to retrieve 70–85% of the correlation energy in VMC

Need for better wave functions

Why do we need better wave functions?

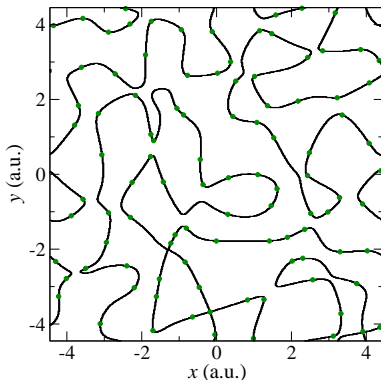
- DMC energies depend on the nodes of Ψ , but Jastrow factors do not modify the nodes of Ψ_S
- We want **chemical accuracy** (\equiv 1.5 mHa away from ground state energy at most)
- Expectation values other than the energy are **sensitive** to the quality of the wave function
- Improving Ψ and its nodes involves interesting Physics/Maths

How hard can nodes be?



An innocent 2D HEG, a “simple” system

How hard can nodes be?



An innocent 2D HEG, a “simple” system

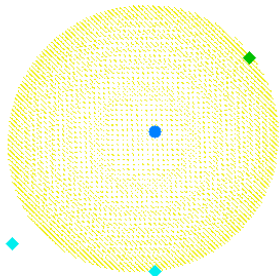
How hard can nodes be?

The nodes of Ψ_{SJ} :

- Have **no** connection with the nodes of the orbitals
- Are **rarely sampled** → are **hard to optimise** within standard schemes → but are **the source of error** anyway
- Tend to divide config space into too many regions (nodal pockets) → tend to have the **wrong topology** (should satisfy a **tiling property**)
- Are not too far away from the true answer

How hard can nodes be?

HF nodes vs. exact nodes



Beryllium atom (see separate animation)

Possible wave functions

- Optimise the **orbitals** in the determinants
- Borrow techniques from Quantum Chemistry
→ **multi-determinant** expansions
- Make the orbitals depend on the inter-electronic distances, by
 - Using **pairing**/(**multi**-)**geminal**/(**multi**-)**Pfaffian** wave functions
 - Using coordinate transformations → **backflow**

Orbital parametrisation

- In atomic systems, modify the radial function,

$$\phi_{nlm}(\mathbf{r}) = [\rho_{nl}^{\text{HF}}(r) + \Delta\rho_{nl}(r)] r^l Y_{lm}(\theta, \phi)$$

with

$$\Delta\rho_{nl}(r) = \left(\sum_{j=0}^{N_p} c_{jnl} r^j \right) \exp\left(\frac{-A_{nl} r^2}{1 + B_{nl} r} \right)$$

N. D. Drummond *et al.*, J. Chem. Phys. **124**, 224104 (2006)

- In small molecules one can expand the orbitals in a Slater basis
C. Filippi and C. J. Umrigar, J. Chem. Phys. **105**, 213 (1996)
- If using Gaussians, can optimise the expansion coeffs and the exponents

How well does orbital optimisation work?

- Variable degree of success in different studies
- In general the results are not all that impressive, e.g.,
 - **VMC** for AE Ne and Ne^+ → it retrieves $\sim 12\%$ of the difference between VMC with HF orbitals and the exact answer
 - **DMC** for AE Ne and Ne^+ → it retrieves $\sim 3\%$ of the difference between DMC with HF orbitals and the exact answer → **little nodal improvement**
 - **VMC** and **DMC** for PP Ne and Ne^+ → it retrieves $\sim 2\%$ of difference with exact answer

Multi-determinant expansions

- The eigenstates $\psi_n(\mathbf{r})$ of a single-particle Hamiltonian \equiv **basis** for functions of $\mathbf{r} \rightarrow$ the determinants constructed from combinations of N orbitals for N particles \equiv **basis for antisymmetric functions** of $\mathbf{R} \rightarrow$ the exact wave function is

$$\Phi_0(\mathbf{R}) = \sum_{k=0}^{\infty} c_k D_k(\mathbf{R})$$

(similarly with spin-splitting)

- Notice that $c_k = \delta_{0k}$ corresponds to the **Hartree-Fock** wave function

Applicability

- Small atoms → great results
- Small molecules → great results too
- Medium-size systems hard
- Large/crystalline systems impossible

Ideally, we would like to depend less strongly on system size

Geminal wave function

Functional form:

$$\Psi_S^{\text{gem}}(\mathbf{R}) = \det \left| \Gamma \left(\mathbf{r}_i^\uparrow, \mathbf{r}_j^\downarrow \right) \right|$$

where

$$\Gamma \left(\mathbf{r}^\uparrow, \mathbf{r}^\downarrow \right) = \sum_{kl} \gamma_{kl} \phi_k^\uparrow(\mathbf{r}^\uparrow) \phi_l^\downarrow(\mathbf{r}^\downarrow)$$

Subject of current research.

The Slater–Jastrow-backflow wave function

The backflow transformation is applied on SJ wave functions giving

$$\Psi_{\text{BF}}(\mathbf{R}) = \exp[J(\mathbf{R})] \Psi_{\text{S}}[\mathbf{X}(\mathbf{R})]$$

where \mathbf{X} is a vector of *quasi-particle* coordinates,

$$\mathbf{x}_i(\mathbf{R}) = \mathbf{r}_i + \xi_i(\mathbf{R})$$

Explicitly, $\Psi_{\text{S}}(\mathbf{R})$ is replaced by

$$\Psi_{\text{S}}^{\text{BF}}(\mathbf{R}) \equiv \Psi_{\text{S}}[\mathbf{R}(\mathbf{X})] = \begin{vmatrix} \phi_1[\mathbf{x}_1(\mathbf{R})] & \phi_1[\mathbf{x}_2(\mathbf{R})] & \dots & \phi_1[\mathbf{x}_N(\mathbf{R})] \\ \phi_2[\mathbf{x}_1(\mathbf{R})] & \phi_2[\mathbf{x}_2(\mathbf{R})] & \dots & \phi_2[\mathbf{x}_N(\mathbf{R})] \\ \vdots & \vdots & \ddots & \vdots \\ \phi_N[\mathbf{x}_1(\mathbf{R})] & \phi_N[\mathbf{x}_2(\mathbf{R})] & \dots & \phi_N[\mathbf{x}_N(\mathbf{R})] \end{vmatrix}$$

Interpreting backflow geometrically

A good way of looking at backflow transformations is as an addition which has a complementary effect to that of the Jastrow factor

- A Jastrow factor can change the “vertical” shape of a wave function, acting on it as a **positive envelope**
- A backflow transformation can **shift** the wave function in the “horizontal” direction

Parametrisation of backflow

Want to construct $\xi_i(\mathbf{R})$ as a sum of e-e, e-n and e-e-n functions:

$$\xi_i^{e-e} = \sum_{j \neq i}^{N_e} \eta_{ij} \mathbf{r}_{ij} \quad , \quad \eta_{ij} = \eta(r_{ij})$$

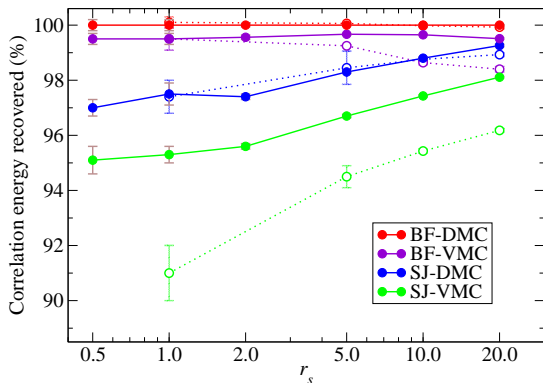
$$\xi_i^{e-n} = \sum_I^{N_n} \mu_{iI} \mathbf{r}_{iI} \quad , \quad \mu_{iI} = \mu(r_{iI})$$

$$\xi_i^{e-e-n} = \sum_{j \neq i}^{N_e} \sum_I^{N_n} (\Phi_i^{jI} \mathbf{r}_{ij} + \Theta_i^{jI} \mathbf{r}_{iI}) \quad , \quad \Phi_i^{jI} = \Phi_I(r_{ij}, r_{iI}, r_{jI})$$

$$\Theta_i^{jI} = \Theta_I(r_{ij}, r_{iI}, r_{jI})$$

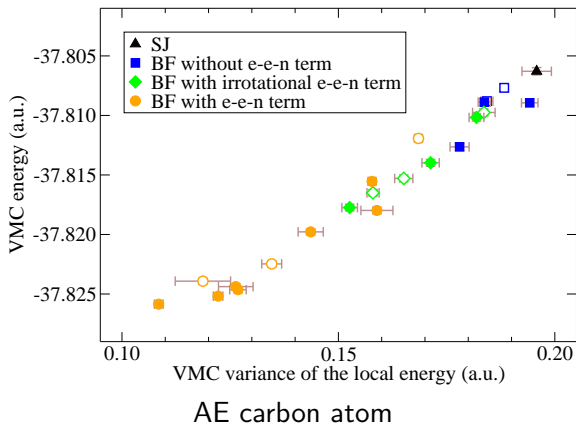
Total ξ_i is $\xi_i^{e-e} + \xi_i^{e-n} + \xi_i^{e-e-n}$

Backflow results

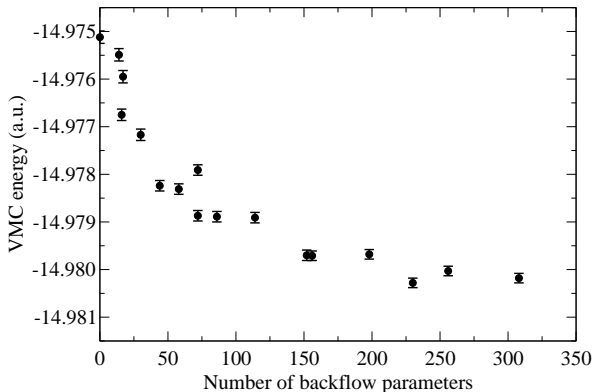


3D 54-electron HEG

Backflow results



Backflow results



AE Li₂ molecule

Backflow results

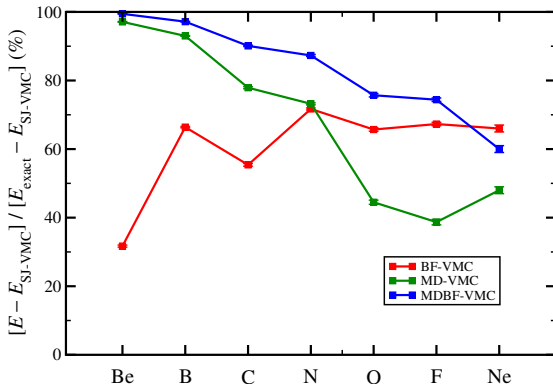
Properties of backflow:

- Variance of the local energy **reduced**
- Wave function update algorithms are more **costly**
- Non-local projection for PP energy requires lots of wave function evaluations → **costly**
- CBCA simpler, but EBEA still favourable

What goes well with what?

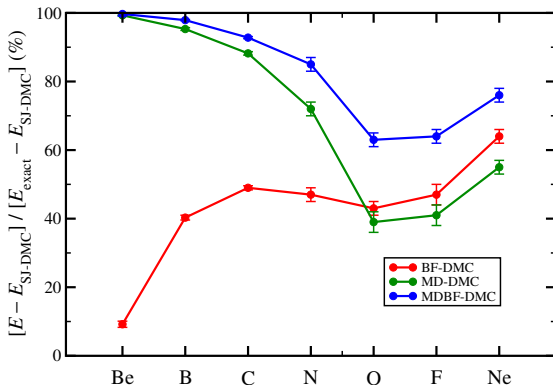
- **OO** and **BF**: orbital optimisation largely **overlaps** with backflow → only need to optimise orbitals on BF wave functions when initial orbitals fundamentally **wrong**
- **OO** and **MD**: probably the same case as OO and BF (data to compare OO, MD and OO+MD consistently?)
- **MD** and **BF**: excellent where MD is good, otherwise use BF only

Multi-determinants vs. backflow



MD expansions can be very successful for small systems

Multi-determinants vs. backflow



MD expansions can be very successful for small systems