Discussion of Slater determinants and project 2

Morten Hjorth-Jensen Email morten.hjorth-jensen@fys.uio.no[1]

Department of Physics and Center fo Computing in Science Education, University of Oslo, Oslo, Norway[1]

April 25, 2024

Overview of week April 21-25, 2025

1. Discussion of Slater determinants and project 2
2. Possible topics for the last two lectures, April 26 and May 3, Diffusion Monte Carlo calculations

► Video of lecture TBA
► Handwritten notes

Project 2, VMC for fermions: Efficient calculation of Slater determinants

The potentially most time-consuming part is the evaluation of the gradient and the Laplacian of an N-particle Slater determinant. We have to differentiate the determinant with respect to all spatial coordinates of all particles. A brute force differentiation would involve $N \cdot d$ evaluations of the entire determinant which would even worsen the already undesirable time scaling, making it $Nd \cdot O(N^3) \sim O(d \cdot N^4)$.

This poses serious hindrances to the overall efficiency of our code.

Matrix elements of Slater determinants

The efficiency can be improved however if we move only one electron at the time. The Slater determinant matrix $\hat{D}$ is defined by the matrix elements

$$d_{ij} = \phi_j(x_i)$$

where $\phi_j(r_i)$ is a single particle wave function. The columns correspond to the position of a given particle while the rows stand for the various quantum numbers.

Efficient calculation of Slater determinants

What we need to realize is that when differentiating a Slater determinant with respect to some given coordinate, only one row of the corresponding Slater matrix is changed.

Therefore, by recalculating the whole determinant we risk producing redundant information. The solution turns out to be an algorithm that requires to keep track of the *inverse* of the Slater matrix.

Efficient calculation of Slater determinants

Let the current position in phase space be represented by the $(N \cdot d)$-element vector r^{old} and the new suggested position by the vector r^{new}.

The inverse of $\hat{D}$ can be expressed in terms of its cofactors C_{ij} and its determinant (this our notation for a determinant) $|\hat{D}|$:

$$d_{ij}^{-1} = \frac{C_{ji}}{|\hat{D}|} \tag{1}$$

Notice that the interchanged indices indicate that the matrix of cofactors is to be transposed.

If $\hat{D}$ is invertible, then we must obviously have $\hat{D}^{-1}\hat{D} = 1$, or explicitly in terms of the individual elements of $\hat{D}$ and $\hat{D}^{-1}$:

$$\sum_{k=1}^{N} d_{ik} \, d_{kj}^{-1} = \delta_{ij} \tag{2}$$

Efficient calculation of Slater determinants

Consider the ratio, which we shall call R, between $|\hat{D}(r^{\mathrm{new}})|$ and $|\hat{D}(r^{\mathrm{old}})|$. By definition, each of these determinants can individually be expressed in terms of the i-th row of its cofactor matrix

$$R \equiv \frac{|\hat{D}(r^{\mathrm{new}})|}{|\hat{D}(r^{\mathrm{old}})|} = \frac{\sum_{j=1}^{N} d_{ij}(r^{\mathrm{new}})\, C_{ij}(r^{\mathrm{new}})}{\sum_{j=1}^{N} d_{ij}(r^{\mathrm{old}})\, C_{ij}(r^{\mathrm{old}})} \tag{3}$$

Efficient calculation of Slater determinants

Suppose now that we move only one particle at a time, meaning that r^{new} differs from r^{old} by the position of only one, say the i-th, particle . This means that $\hat{D}(r^{\text{new}})$ and $\hat{D}(r^{\text{old}})$ differ only by the entries of the i-th row. Recall also that the i-th row of a cofactor matrix $\hat{C}$ is independent of the entries of the i-th row of its corresponding matrix $\hat{D}$. In this particular case we therefore get that the i-th row of $\hat{C}(r^{\text{new}})$ and $\hat{C}(r^{\text{old}})$ must be equal. Explicitly, we have:

$$C_{ij}(r^{\text{new}}) = C_{ij}(r^{\text{old}}) \quad \forall\, j \in \{1, \ldots, N\}$$

Efficient calculation of Slater determinants

Inserting this into the numerator of eq. (3) and using eq. (1) to substitute the cofactors with the elements of the inverse matrix, we get:

$$R = \frac{\sum_{j=1}^{N} d_{ij}(\mathrm{r^{new}}) \, C_{ij}(\mathrm{r^{old}})}{\sum_{j=1}^{N} d_{ij}(\mathrm{r^{old}}) \, C_{ij}(\mathrm{r^{old}})} = \frac{\sum_{j=1}^{N} d_{ij}(\mathrm{r^{new}}) \, d_{ji}^{-1}(\mathrm{r^{old}})}{\sum_{j=1}^{N} d_{ij}(\mathrm{r^{old}}) \, d_{ji}^{-1}(\mathrm{r^{old}})}$$

Efficient calculation of Slater determinants

Now by eq. (2) the denominator of the rightmost expression must be unity, so that we finally arrive at:

$$R = \sum_{j=1}^{N} d_{ij}(\mathbf{r}^{\mathrm{new}}) \, d_{ji}^{-1}(\mathbf{r}^{\mathrm{old}}) = \sum_{j=1}^{N} \phi_j(\mathbf{r}_i^{\mathrm{new}}) \, d_{ji}^{-1}(\mathbf{r}^{\mathrm{old}}) \qquad (4)$$

What this means is that in order to get the ratio when only the i-th particle has been moved, we only need to calculate the dot product of the vector $(\phi_1(\mathbf{r}_i^{\mathrm{new}}), \, \ldots, \, \phi_N(\mathbf{r}_i^{\mathrm{new}}))$ of single particle wave functions evaluated at this new position with the i-th column of the inverse matrix $\hat{D}^{-1}$ evaluated at the original position. Such an operation has a time scaling of $O(N)$. The only extra thing we need to do is to maintain the inverse matrix $\hat{D}^{-1}(x^{\mathrm{old}})$.

Efficient calculation of Slater determinants

If the new position r^{new} is accepted, then the inverse matrix can by suitably updated by an algorithm having a time scaling of $O(N^2)$. This algorithm goes as follows. First we update all but the i-th column of $\hat{D}^{-1}$. For each column $j \neq i$, we first calculate the quantity:

$$S_j = (\hat{D}(r^{\text{new}}) \times \hat{D}^{-1}(r^{\text{old}}))_{ij} = \sum_{l=1}^{N} d_{il}(r^{\text{new}})\, d_{lj}^{-1}(r^{\text{old}})$$

Efficient calculation of Slater determinants

The new elements of the j-th column of $\hat{D}^{-1}$ are then given by:

$$d_{kj}^{-1}(\mathbf{r}^{\mathrm{new}}) = d_{kj}^{-1}(\mathbf{r}^{\mathrm{old}}) - \frac{S_j}{R} \, d_{ki}^{-1}(\mathbf{r}^{\mathrm{old}}) \qquad \begin{array}{l} \forall \;\; k \in \{1, \ldots, N\} \\ j \neq i \end{array}$$

Efficient calculation of Slater determinants

Finally the elements of the i-th column of $\hat{D}^{-1}$ are updated simply as follows:

$$d_{ki}^{-1}(r^{\text{new}}) = \frac{1}{R}\, d_{ki}^{-1}(r^{\text{old}}) \quad \forall \;\; k \in \{1, \ldots, N\}$$

We see from these formulas that the time scaling of an update of $\hat{D}^{-1}$ after changing one row of $\hat{D}$ is $O(N^2)$.

The scheme is also applicable for the calculation of the ratios involving derivatives. It turns out that differentiating the Slater determinant with respect to the coordinates of a single particle r_i changes only the i-th row of the corresponding Slater matrix.

The gradient and the Laplacian

The gradient and the Laplacian can therefore be calculated as follows:

$$\frac{\vec{\nabla}_i |\hat{D}(r)|}{|\hat{D}(r)|} = \sum_{j=1}^{N} \vec{\nabla}_i d_{ij}(r) d_{ji}^{-1}(r) = \sum_{j=1}^{N} \vec{\nabla}_i \phi_j(r_i) d_{ji}^{-1}(r)$$

and

$$\frac{\nabla_i^2 |\hat{D}(r)|}{|\hat{D}(r)|} = \sum_{j=1}^{N} \nabla_i^2 d_{ij}(r) d_{ji}^{-1}(r) = \sum_{j=1}^{N} \nabla_i^2 \phi_j(r_i) \, d_{ji}^{-1}(r)$$

How to compute the derivates of the Slater determinant

Thus, to calculate all the derivatives of the Slater determinant, we only need the derivatives of the single particle wave functions ($\vec{\nabla}_i \phi_j(r_i)$ and $\nabla_i^2 \phi_j(r_i)$) and the elements of the corresponding inverse Slater matrix ($\hat{D}^{-1}(r_i)$). A calculation of a single derivative is by the above result an $O(N)$ operation. Since there are $d \cdot N$ derivatives, the time scaling of the total evaluation becomes $O(d \cdot N^2)$. With an $O(N^2)$ updating algorithm for the inverse matrix, the total scaling is no worse, which is far better than the brute force approach yielding $O(d \cdot N^4)$.

Important note: In most cases you end with closed form expressions for the single-particle wave functions. It is then useful to calculate the various derivatives and make separate functions for them.

The Slater determinant, example

The Slater determinant for atomic Beryllium could for example take the form

$$\Phi(r_1, r_2, , r_3, r_4, \alpha, \beta, \gamma, \delta) = \frac{1}{\sqrt{4!}} \begin{vmatrix} \psi_{100\uparrow}(r_1) & \psi_{100\uparrow}(r_2) & \psi_{100\uparrow}(r_3) & \psi_{100} \\ \psi_{100\downarrow}(r_1) & \psi_{100\downarrow}(r_2) & \psi_{100\downarrow}(r_3) & \psi_{100} \\ \psi_{200\uparrow}(r_1) & \psi_{200\uparrow}(r_2) & \psi_{200\uparrow}(r_3) & \psi_{200} \\ \psi_{200\downarrow}(r_1) & \psi_{200\downarrow}(r_2) & \psi_{200\downarrow}(r_3) & \psi_{200} \end{vmatrix}$$

This expression can lead to problems when we omit the spin degrees of freedom, as is common in for example many atomic physics calculations. Leaving out the spin degrees of freedom, the problem we may encounter is that of zero determinants. But we can rewrite it as the product of two Slater determinants, one for spin up and one for spin down.

Rewriting the Slater determinant

We can rewrite it as

$$\Phi(r_1, r_2, , r_3, r_4, \alpha, \beta, \gamma, \delta) = \det \uparrow (1,2) \det \downarrow (3,4) - \det \uparrow (1,3) \det \downarrow (2,4$$

$$- \det \uparrow (1,4) \det \downarrow (3,2) + \det \uparrow (2,3) \det \downarrow (1,4) - \det \uparrow (2,4) \det \downarrow (1,3)$$

$$+ \det \uparrow (3,4) \det \downarrow (1,2),$$

where we have defined

$$\det \uparrow (1,2) = \frac{1}{\sqrt{2}} \begin{vmatrix} \psi_{100\uparrow}(r_1) & \psi_{100\uparrow}(r_2) \\ \psi_{200\uparrow}(r_1) & \psi_{200\uparrow}(r_2) \end{vmatrix},$$

and

$$\det \downarrow (3,4) = \frac{1}{\sqrt{2}} \begin{vmatrix} \psi_{100\downarrow}(r_3) & \psi_{100\downarrow}(r_4) \\ \psi_{200\downarrow}(r_3) & \psi_{200\downarrow}(r_4) \end{vmatrix}.$$

Note that if we again leave out the spin degrees of freedom, the determinant is still zero!

Splitting the Slater determinant

We want to avoid to sum over spin variables, in particular when the interaction does not depend on spin.

It can be shown, see for example Moskowitz and Kalos, Int. J. Quantum Chem. **20** 1107 (1981), that for the variational energy we can approximate the Slater determinant as

$$\Phi(r_1, r_2, , r_3, r_4, \alpha, \beta, \gamma, \delta) \propto \det \uparrow (1, 2) \det \downarrow (3, 4),$$

or more generally as

$$\Phi(r_1, r_2, \ldots r_N) \propto \det \uparrow \det \downarrow,$$

where we have the Slater determinant as the product of a spin up part involving the number of electrons with spin up only (2 for beryllium and 5 for neon) and a spin down part involving the electrons with spin down.

This ansatz is not antisymmetric under the exchange of electrons with opposite spins but it can be shown (show this) that it gives the same expectation value for the energy as the full Slater determinant.

As long as the Hamiltonian is spin independent, the above approach gives us the same expectation value for the energy. It is rather straightforward to see this if you go back to the equations for the energy. We leave this as an exercise to the eager reader. Can you think of observables where not respecting the symmetry can have consequences?

Omitting or not omitting spin degrees of freedom

If we keep the spin degrees of freedom, which obviously leads to a more general code, we would need to flip spins as well and deal with the full Slater determinant. The above recipe is just a mere simplification to a case where we have identical particles, the same spatial single-particle functions and the same number of spin-up and spin-down fermions.

Restricted versus unrestricted

This is a situation which one encounters in for example a fermionic system like a closed-shell nucleus like oxygen-16 or a neutral noble gas like helium or neon with the same number of spin-up and spin-down orbitals and the same spatial single-particle functions. The example discussed above for neutral Beryllium where we fill the hydrogen-like states 1s and 2s, is yet another case.
For those of you familiar with Hartree-Fock theory, this is often referred to as **restricted** Hartree-Fock theory. Unrestricted Hartree-Fock theory represents then the more general case.

Spin up and spin down parts

The systems we will limit ourselves to, are all systems which can be described by a restricted basis set, We will thus factorize the full determinant $|\hat{D}|$ into two smaller ones, where each can be identified with $\uparrow$ and $\downarrow$ respectively:

$$|\hat{D}| = |\hat{D}|_\uparrow \cdot |\hat{D}|_\downarrow$$

Factorization

The combined dimensionality of the two smaller determinants equals the dimensionality of the full determinant. Such a factorization is advantageous in that it makes it possible to perform the calculation of the ratio R and the updating of the inverse matrix separately for $|\hat{D}|_\uparrow$ and $|\hat{D}|_\downarrow$:

$$\frac{|\hat{D}|^{\text{new}}}{|\hat{D}|^{\text{old}}} = \frac{|\hat{D}|_\uparrow^{\text{new}}}{|\hat{D}|_\uparrow^{\text{old}}} \cdot \frac{|\hat{D}|_\downarrow^{\text{new}}}{|\hat{D}|_\downarrow^{\text{old}}}$$

This reduces the calculation time by a constant factor. The maximal time reduction happens in a system of equal numbers of $\uparrow$ and $\downarrow$ particles, so that the two factorized determinants are half the size of the original one.

Number of operations

Consider the case of moving only one particle at a time which originally had the following time scaling for one transition:

$$O_R(N) + O_{\mathrm{inverse}}(N^2)$$

For the factorized determinants one of the two determinants is obviously unaffected by the change so that it cancels from the ratio R.

Counting the number of FLOPS

Therefore, only one determinant of size $N/2$ is involved in each calculation of R and update of the inverse matrix. The scaling of each transition then becomes:

$$O_R(N/2) + O_{\text{inverse}}(N^2/4)$$

and the time scaling when the transitions for all N particles are put together:

$$O_R(N^2/2) + O_{\text{inverse}}(N^3/4)$$

which gives the same reduction as in the case of moving all particles at once.

Computation of ratios

Computing the ratios discussed above requires that we maintain the inverse of the Slater matrix evaluated at the current position. Each time a trial position is accepted, the row number i of the Slater matrix changes and updating its inverse has to be carried out. Getting the inverse of an $N \times N$ matrix by Gaussian elimination has a complexity of order of $\mathcal{O}(N^3)$ operations, a luxury that we cannot afford for each time a particle move is accepted. We will use the expression

$$
d_{kj}^{-1}(x^{\text{new}}) = \begin{cases} d_{kj}^{-1}(x^{\text{old}}) - \frac{d_{ki}^{-1}(x^{\text{old}})}{R} \sum_{l=1}^{N} d_{il}(x^{\text{new}}) d_{lj}^{-1}(x^{\text{old}}) & \text{if } j \neq i \\[4mm] \frac{d_{ki}^{-1}(x^{\text{old}})}{R} \sum_{l=1}^{N} d_{il}(x^{\text{old}}) d_{lj}^{-1}(x^{\text{old}}) & \text{if } j = i \end{cases}
$$

Scaling properties

This equation scales as $O(N^2)$. The evaluation of the determinant of an $N \times N$ matrix by standard Gaussian elimination requires $O(N^3)$ calculations. As there are Nd independent coordinates we need to evaluate Nd Slater determinants for the gradient (quantum force) and Nd for the Laplacian (kinetic energy). With the updating algorithm we need only to invert the Slater determinant matrix once. This can be done by standard LU decomposition methods.

How to get the determinant

Determining a determinant of an $N \times N$ matrix by standard Gaussian elimination is of the order of $O(N^3)$ calculations. As there are $N \cdot d$ independent coordinates we need to evaluate Nd Slater determinants for the gradient (quantum force) and $N \cdot d$ for the Laplacian (kinetic energy)

With the updating algorithm we need only to invert the Slater determinant matrix once. This is done by calling standard LU decomposition methods.

Expectation value of the kinetic energy

The expectation value of the kinetic energy expressed in atomic units for electron i is

$$\langle \hat{K}_i \rangle = -\frac{1}{2} \frac{\langle \Psi | \nabla_i^2 | \Psi \rangle}{\langle \Psi | \Psi \rangle},$$

$$K_i = -\frac{1}{2} \frac{\nabla_i^2 \Psi}{\Psi}.$$

$$\frac{\nabla^2 \Psi}{\Psi} = \frac{\nabla^2 (\Psi_D \Psi_C)}{\Psi_D \Psi_C} = \frac{\nabla \cdot [\nabla (\Psi_D \Psi_C)]}{\Psi_D \Psi_C} = \frac{\nabla \cdot [\Psi_C \nabla \Psi_D + \Psi_D \nabla \Psi_C]}{\Psi_D \Psi_C}$$
$$= \frac{\nabla \Psi_C \cdot \nabla \Psi_D + \Psi_C \nabla^2 \Psi_D + \nabla \Psi_D \cdot \nabla \Psi_C + \Psi_D \nabla^2 \Psi_C}{\Psi_D \Psi_C}$$

$$\tag{5}$$

$$\frac{\nabla^2 \Psi}{\Psi} = \frac{\nabla^2 \Psi_D}{\Psi_D} + \frac{\nabla^2 \Psi_C}{\Psi_C} + 2 \frac{\nabla \Psi_D}{\Psi_D} \cdot \frac{\nabla \Psi_C}{\Psi_C} \tag{6}$$

Second derivative of the Jastrow factor

The second derivative of the Jastrow factor divided by the Jastrow factor (the way it enters the kinetic energy) is

$$\left[\frac{\nabla^2 \Psi_C}{\Psi_C}\right]_x = 2\sum_{k=1}^{N}\sum_{i=1}^{k-1}\frac{\partial^2 g_{ik}}{\partial x_k^2} + \sum_{k=1}^{N}\left(\sum_{i=1}^{k-1}\frac{\partial g_{ik}}{\partial x_k} - \sum_{i=k+1}^{N}\frac{\partial g_{ki}}{\partial x_i}\right)^2$$

Functional form

But we have a simple form for the function, namely

$$\Psi_C = \prod_{i<j} \exp f(r_{ij}) = \exp \left\{ \sum_{i<j} \frac{ar_{ij}}{1 + \beta r_{ij}} \right\},$$

and it is easy to see that for particle k we have

$$\frac{\nabla_k^2 \Psi_C}{\Psi_C} = \sum_{ij \neq k} \frac{(\mathbf{r}_k - \mathbf{r}_i)(\mathbf{r}_k - \mathbf{r}_j)}{r_{ki} r_{kj}} f'(r_{ki}) f'(r_{kj}) + \sum_{j \neq k} \left(f''(r_{kj}) + \frac{2}{r_{kj}} f'(r_{kj}) \right)$$

Second derivative of the Jastrow factor

Using
$$f(r_{ij}) = \frac{ar_{ij}}{1 + \beta r_{ij}},$$

and $g'(r_{kj}) = dg(r_{kj})/dr_{kj}$ and $g''(r_{kj}) = d^2 g(r_{kj})/dr_{kj}^2$ we find that for particle k we have

$$\frac{\nabla_k^2 \Psi_C}{\Psi_C} = \sum_{ij \neq k} \frac{(\mathbf{r}_k - \mathbf{r}_i)(\mathbf{r}_k - \mathbf{r}_j)}{r_{ki} r_{kj}} \frac{a}{(1 + \beta r_{ki})^2} \frac{a}{(1 + \beta r_{kj})^2} + \sum_{j \neq k} \left(\frac{2a}{r_{kj}(1 + \beta r_{kj}} \right.$$

Gradient and Laplacian

The gradient and Laplacian can be calculated as follows:

$$\frac{\nabla_i |\hat{D}(\mathsf{r})|}{|\hat{D}(\mathsf{r})|} = \sum_{j=1}^{N} \vec{\nabla}_i d_{ij}(\mathsf{r}) \, d_{ji}^{-1}(\mathsf{r}) = \sum_{j=1}^{N} \vec{\nabla}_i \phi_j(\mathsf{r}_i) \, d_{ji}^{-1}(\mathsf{r})$$

and

$$\frac{\nabla_i^2 |\hat{D}(\mathsf{r})|}{|\hat{D}(\mathsf{r})|} = \sum_{j=1}^{N} \nabla_i^2 d_{ij}(\mathsf{r}) \, d_{ji}^{-1}(\mathsf{r}) = \sum_{j=1}^{N} \nabla_i^2 \phi_j(\mathsf{r}_i) \, d_{ji}^{-1}(\mathsf{r})$$

The gradient for the determinant

The gradient for the determinant is

$$\frac{\nabla_i |\hat{D}(\mathbf{r})|}{|\hat{D}(\mathbf{r})|} = \sum_{j=1}^{N} \nabla_i d_{ij}(\mathbf{r}) \, d_{ji}^{-1}(\mathbf{r}) = \sum_{j=1}^{N} \nabla_i \phi_j(\mathbf{r}_i) \, d_{ji}^{-1}(\mathbf{r}).$$

We have

$$\Psi_C = \prod_{i<j} g(r_{ij}) = \exp\left\{\sum_{i<j} \frac{ar_{ij}}{1 + \beta r_{ij}}\right\},$$

the gradient needed for the quantum force and local energy is easy to compute. We get for particle k

$$\frac{\nabla_k \Psi_C}{\Psi_C} = \sum_{j\neq k} \frac{r_{kj}}{r_{kj}} \frac{a}{(1 + \beta r_{kj})^2},$$

which is rather easy to code. Remember to sum over all particles when you compute the local energy.

We need to compute the ratio between wave functions, in particular for the Slater determinants.

$$R = \sum_{j=1}^{N} d_{ij}(\mathrm{r}^{\mathrm{new}}) \, d_{ji}^{-1}(\mathrm{r}^{\mathrm{old}}) = \sum_{j=1}^{N} \phi_j(\mathrm{r}_i^{\mathrm{new}}) \, d_{ji}^{-1}(\mathrm{r}^{\mathrm{old}})$$

What this means is that in order to get the ratio when only the i-th particle has been moved, we only need to calculate the dot product of the vector $(\phi_1(\mathrm{r}_i^{\mathrm{new}}), \ldots, \phi_N(\mathrm{r}_i^{\mathrm{new}}))$ of single particle wave functions evaluated at this new position with the i-th column of the inverse matrix $\hat{D}^{-1}$ evaluated at the original position. Such an operation has a time scaling of $O(N)$. The only extra thing we need to do is to maintain the inverse matrix $\hat{D}^{-1}(\mathrm{x}^{\mathrm{old}})$.

Proof for updating algorithm for Slater determinant

As a starting point we may consider that each time a new position is suggested in the Metropolis algorithm, a row of the current Slater matrix experiences some kind of perturbation. Hence, the Slater matrix with its orbitals evaluated at the new position equals the old Slater matrix plus a perturbation matrix,

$$d_{jk}(\mathbf{x}^{\text{new}}) = d_{jk}(\mathbf{x}^{\text{old}}) + \Delta_{jk},$$

where

$$\Delta_{jk} = \delta_{ik}[\phi_j(\mathbf{x}_i^{\text{new}}) - \phi_j(\mathbf{x}_i^{\text{old}})] = \delta_{ik}(\Delta\phi)_j.$$

Computing the inverse of the transposed matrix we arrive at

$$d_{kj}(x^{\text{new}})^{-1} = [d_{kj}(x^{\text{old}}) + \Delta_{kj}]^{-1}. \tag{7}$$

Proof for updating algorithm for Slater determinant

The evaluation of the right hand side (rhs) term above is carried out by applying the identity $(A + B)^{-1} = A^{-1} - (A + B)^{-1} B A^{-1}$. In compact notation it yields

$$
\begin{aligned}
[D^T(x^{\text{new}})]^{-1} &= [D^T(x^{\text{old}}) + \Delta^T]^{-1} \\
&= [D^T(x^{\text{old}})]^{-1} - [D^T(x^{\text{old}}) + \Delta^T]^{-1} \Delta^T [D^T(x^{\text{old}})]^{-1} \\
&= [D^T(x^{\text{old}})]^{-1} - \underbrace{[D^T(x^{\text{new}})]^{-1}}_{\text{By Eq.7}} \Delta^T [D^T(x^{\text{old}})]^{-1}.
\end{aligned}
$$

Proof for updating algorithm for Slater determinant

Using index notation, the last result may be expanded by

$$
\begin{aligned}
d_{kj}^{-1}(x^{\text{new}}) &= d_{kj}^{-1}(x^{\text{old}}) - \sum_l \sum_m d_{km}^{-1}(x^{\text{new}}) \Delta_{ml}^T d_{lj}^{-1}(x^{\text{old}}) \\
&= d_{kj}^{-1}(x^{\text{old}}) - \sum_l \sum_m d_{km}^{-1}(x^{\text{new}}) \Delta_{lm} d_{lj}^{-1}(x^{\text{cur}}) \\
&= d_{kj}^{-1}(x^{\text{old}}) - \sum_l \sum_m d_{km}^{-1}(x^{\text{new}}) \delta_{im}(\Delta\phi)_l d_{lj}^{-1}(x^{\text{old}}) \\
&= d_{kj}^{-1}(x^{\text{old}}) - d_{ki}^{-1}(x^{\text{new}}) \sum_{l=1}^{N} (\Delta\phi)_l d_{lj}^{-1}(x^{\text{old}}) \\
&= d_{kj}^{-1}(x^{\text{old}}) - d_{ki}^{-1}(x^{\text{new}}) \sum_{l=1}^{N} [\phi_l(r_i^{\text{new}}) - \phi_l(r_i^{\text{old}})] D_{lj}^{-1}(x^{\text{old}}).
\end{aligned}
$$

Proof for updating algorithm for Slater determinant

Using

$$D^{-1}(x^{\text{old}}) = \frac{adj D}{|D(x^{\text{old}})|} \quad \text{and} \quad D^{-1}(x^{\text{new}}) = \frac{adj D}{|D(x^{\text{new}})|},$$

and dividing these two equations we get

$$\frac{D^{-1}(x^{\text{old}})}{D^{-1}(x^{\text{new}})} = \frac{|D(x^{\text{new}})|}{|D(x^{\text{old}})|} = R \Rightarrow d_{ki}^{-1}(x^{\text{new}}) = \frac{d_{ki}^{-1}(x^{\text{old}})}{R}.$$

Proof for updating algorithm for Slater determinant

We have

$$d_{kj}^{-1}(x^{\text{new}}) = d_{kj}^{-1}(x^{\text{old}}) - \frac{d_{ki}^{-1}(x^{\text{old}})}{R} \sum_{l=1}^{N} [\phi_l(r_i^{\text{new}}) - \phi_l(r_i^{\text{old}})] d_{lj}^{-1}(x^{\text{old}}),$$

or

$$
\begin{aligned}
d_{kj}^{-1}(x^{\text{new}}) = d_{kj}^{-1}(x^{\text{old}}) \quad &- \quad \frac{d_{ki}^{-1}(x^{\text{old}})}{R} \sum_{l=1}^{N} \phi_l(r_i^{\text{new}}) d_{lj}^{-1}(x^{\text{old}}) \\
&+ \quad \frac{d_{ki}^{-1}(x^{\text{old}})}{R} \sum_{l=1}^{N} \phi_l(r_i^{\text{old}}) d_{lj}^{-1}(x^{\text{old}}) \\
= d_{kj}^{-1}(x^{\text{old}}) \quad &- \quad \frac{d_{ki}^{-1}(x^{\text{old}})}{R} \sum_{l=1}^{N} d_{il}(x^{\text{new}}) d_{lj}^{-1}(x^{\text{old}}) \\
&+ \quad \frac{d_{ki}^{-1}(x^{\text{old}})}{R} \sum_{l=1}^{N} d_{il}(x^{\text{old}}) d_{lj}^{-1}(x^{\text{old}}).
\end{aligned}
$$

Proof for updating algorithm for Slater determinant

In this equation, the first line becomes zero for $j = i$ and the second for $j \neq i$. Therefore, the update of the inverse for the new Slater matrix is given by

$$d_{kj}^{-1}(x^{\text{new}}) = \begin{cases} d_{kj}^{-1}(x^{\text{old}}) - \dfrac{d_{ki}^{-1}(x^{\text{old}})}{R} \sum_{l=1}^{N} d_{il}(x^{\text{new}}) d_{lj}^{-1}(x^{\text{old}}) & \text{if } j \neq i \\[4mm] \dfrac{d_{ki}^{-1}(x^{\text{old}})}{R} \sum_{l=1}^{N} d_{il}(x^{\text{old}}) d_{lj}^{-1}(x^{\text{old}}) & \text{if } j = i \end{cases}$$