

# Flash Cards for Quantum/Nuclear Monte Carlo

Cody L. Petrie

November 12, 2015

## Parameters in the Code

**g2bval(d2b,sxz,fij)** Given sxz this computes the d2b terms.

**hspot(???)** ???

**op2val(d2b,sp,spx)** ???

**opmult(sp)** This multiplies sp(s,i) by the 15 operators in this order 1-3 sx,sy,sz, 4-6 tx,ty,tz, 7-9 sx\*(tx,ty,tz), 10-12 sy\*(tx,ty,tz), 13-15 sz\*(tx,ty,tz). This outputs opmult(s,kop,i)

**szxupdate(sxznew(out),detrat(out),sxzold,i,opi,sp)** Here the outputted detratt is simply di(i).

**vnpsi2(w,dopot)** This subroutine ...???

**d15(kop)** = di(i) =  $\sum_k S_{ik}^{-1} \langle k | \mathcal{O}_i | \mathbf{r}, s_i \rangle = (S'/S)_{ii}$  for a specific kop (1 of 15)

**d2b(s,s',ij)** =  $\frac{\langle \Phi | R, s_1, \dots, s_{i-1}, s, s_{i+1}, \dots, s_{j-1}, s', s_{j+1}, \dots, s_A \rangle}{\langle \Phi | RS \rangle}$

**d2bip(s,s',s'',s''',ij,kl)** =  $\frac{\langle \Phi | R, s_1, \dots, s, \dots, s', \dots, s'', \dots, s''', \dots, s_A \rangle}{\langle \Phi | R, S \rangle}$  OR

**di(m)** =  $\sum_k S_{mk}^{-1} \langle k | \mathcal{O}_i | \mathbf{r}_i, s_i \rangle = \sum_s \text{opi}(s, m) \langle s | s_i \rangle$

**f2b(s,s',ij)** =  $\sum_{kop=1}^{15} f_{ij}^{kop} \langle s s' | \mathcal{O}_{ij}^{kop} | s_i s_j \rangle$

**fst(3,3,ij)** = f in front of specific operator

**opi(s,m)** =  $\sum_k S_{mk}^{-1} \langle k | \mathcal{O}_i | \mathbf{r}_i, s \rangle = \sum_{s'} \text{sxz}(s', i, m) \langle s' | \mathcal{O}_i | s \rangle$

**ph(i,4,j,idet)** =  $\sum_k S_{ik} \langle k | \mathcal{O}_j | s_j \rangle$

**sigma(3,3,npair)** I think the 3 is x,y,z.

**sigtau(3,3,3,3,npair)** I think the 3 is x,y,z.

$$\text{sp}(\mathbf{s}, \mathbf{i}) = \langle s | s_i \rangle$$

$$\text{spx}(\mathbf{s}, \mathbf{15}, \mathbf{i}) = \langle s | \mathcal{O}_i^p | s_i \rangle, \text{ where } p \text{ goes over the 15 cartesian coordinates.}$$

$$\text{stz}(\mathbf{s}, \mathbf{s}') = \text{spx}(\mathbf{s}, \sigma_\alpha \tau_\gamma, \mathbf{i})^* \text{spx}(\mathbf{s}', \sigma_\beta \tau_\gamma, \mathbf{j}) = \langle s, s' | \sigma_{\alpha i} \tau_{\gamma i} \sigma_{\beta j} \tau_{\gamma j} | s_i s_j \rangle$$

$$\text{sx15}(\mathbf{s}, \mathbf{15}, \mathbf{i}, \mathbf{j}) = \sum_k S_{jk}^{-1} \langle k | \sum_{kop=1}^{15} \mathcal{O}_{kop} | \mathbf{r}, s \rangle = \text{opmult}(\text{sxz0}), \text{ where } \text{sx15}(:, \text{kop}, :, k) = \text{opi}(\mathbf{s}, k)$$

$$\text{sxz}(\mathbf{s}, \mathbf{i}, \mathbf{j}) = \sum_k S_{jk}^{-1} \langle k | \mathbf{r}_i, s \rangle$$

$$\text{sxzi}(\mathbf{s}, \mathbf{i}, \mathbf{j}, \mathbf{iop}) = \sum_k S_{jk}^{'-1} S'_{ki}(s) = \sum_k S_{jk}^{'-1} \langle k | \mathcal{O}_i^{iop} | \mathbf{r}_i, s \rangle$$

$$\text{sxzj}(\mathbf{s}, \mathbf{i}, \mathbf{j}) = \sum_k S_{jk}^{'-1} S''_{ki}(s) = \sum_k S_{jk}^{'-1} \langle k | \mathcal{O}_{ij}^{iopjop} | \mathbf{r}_i, s \rangle, \text{ each } iop \text{ and } jop \text{ is looped over and added to d2b using call g2bval(d2b, sxzj, fij).}$$

$$\text{sz}(\mathbf{s}, \mathbf{s}') = \text{spx}(\mathbf{s}, \sigma_\alpha, \mathbf{i})^* \text{spx}(\mathbf{s}', \sigma_\beta, \mathbf{j}) = \langle s, s' | \sigma_{\alpha i} \sigma_{\beta j} | s_i s_j \rangle$$

$$\text{tau}(\mathbf{3}, \mathbf{3}, \mathbf{npair}) \text{ I think the 3 is x,y,z.}$$

$$\text{tz}(\mathbf{s}, \mathbf{s}') = \text{spx}(\mathbf{s}, \tau_\alpha, \mathbf{i})^* \text{spx}(\mathbf{s}', \tau_\alpha, \mathbf{j}) = \langle s, s' | \tau_{\alpha i} \tau_{\alpha j} | s_i s_j \rangle$$

## Variational Monte Carlo

### Steps for Metropolis Algorithm:

1. Start with some random walker configuration  $\mathbf{R}$
2. Propose a move to a new walker  $\mathbf{R}'$  from the distribution  $T(\mathbf{R}' \leftarrow \mathbf{R})$
3. The probability of accepting the move is given by

$$A(\mathbf{R}' \leftarrow \mathbf{R}) = \min \left( 1, \frac{T(\mathbf{R}' \leftarrow \mathbf{R}) P(\mathbf{R}')}{T(\mathbf{R} \leftarrow \mathbf{R}') P(\mathbf{R})} \right).$$

The move is accepted if  $U[0, 1] < A(\mathbf{R}' \leftarrow \mathbf{R})$ .

4. Repeat from step 2.

**Variational Energy (In terms of  $E_L(\mathbf{R})$  and  $P(\mathbf{R})$ ),  $\mathbf{x2} + E_L$  and  $P$ :**

$$E_V = \frac{\int \Psi^*(\mathbf{R}) \hat{H} \Psi(\mathbf{R}) d\mathbf{R}}{\int \Psi^*(\mathbf{R}) \Psi(\mathbf{R}) d\mathbf{R}} = \int P(\mathbf{R}) E_L(\mathbf{R}) d\mathbf{R}$$

$$P(\mathbf{R}) = |\Psi_T(\mathbf{R})|^2 / \int |\Psi_T(\mathbf{R})|^2 d\mathbf{R}$$

$$E_L(\mathbf{R}) = \Psi_T^{-1}(\mathbf{R}) \hat{H} \Psi_T(\mathbf{R})$$

**Sampled Variational Energy:**

$$E_V \approx \frac{1}{N} \sum_{n=1}^N E_L(\mathbf{R}_n)$$

where  $\mathbf{R}_n$  are drawn from  $P(\mathbf{R})$ .

## General Physics

### Atomic Shell Model

- **Shells** are areas in which an electron can "orbit" a nucleus. They correspond to the principle quantum numbers ( $n=1,2,3,4,\dots$ ) where  $n=1$  is the closest shell.
- The number of allowed electrons is controlled by the spin statistics the other allowed quantum numbers being  $l=0,1,\dots,n-1$  and  $ml=-l,\dots,l$ . The number allowed for each shell is all the possible times two since  $m_s=-1/2,1/2$ .
- **Subshells** are given by the  $l$  quantum numbers where  $l=1,2,3,4$  are s,p,d,f,g.
- List of how many each shell can hold

Shell name	Subshell name	Subshell max electrons	Shell max electrons
K	1s	2	<b>2</b>
L	2s	2	<b>2 + 6 = 8</b>
	2p	6	
M	3s	2	<b>2 + 6 + 10 = 18</b>
	3p	6	
	3d	10	
N	4s	2	<b>2 + 6 + 10 + 14 = 32</b>
	4p	6	
	4d	10	
	4f	14	

### Nuclear Shell Model

The nuclear shell model is similar to the atomic shell model except that  $n=1,2,3,4,\dots$  and  $l=0,1,2,3,\dots$  independently from  $n$ . So you can have states like 1g

## Definitions

**Pedagogical:** Of or relating to teachers or education.

**Coherence length:** It sounds like it's something like the length over which correlations/interactions can happen.

**Cluster decomposition:** If a system is cluster decomposable it means that the expectation value of a product of operators (each operating in separate regions) is equal to the product of their individual expectation values.