

UNIVERSITÀ DEGLI STUDI DI CAGLIARI
DIPARTIMENTO DI SCIENZE FISICHE

**MONTE CARLO QUANTISTICO
PER SISTEMI DI SOLA VALENZA**

Andrea Bosin

Tutore
Prof. Franco Meloni

Coordinatore
Prof. Francesco Casula

FEBBRAIO 1995
DOTTORATO DI RICERCA IN FISICA
VII CICLO - A.A. 1991-94

Indice

Indice	iii
Indice delle Figure	vii
Indice delle Tavole	ix
Introduzione	1
1 Operatore pseudohamiltoniano	5
1.1 Introduzione	6
1.2 Teoria	8
1.2.1 Lo pseudohamiltoniano	8
1.2.2 Le equazioni di Kohn-Sham	13
1.2.3 La conservazione della norma	15
1.2.4 Pseudohamiltoniano e pseudopotenziali non locali	18
1.3 Tecnica di costruzione	20
1.3.1 Aspetti fondamentali	20
1.3.2 Il funzionale di costo e la sua minimizzazione	24
1.3.3 Rappresentazione di $a(r)$, $c(r)$ e $v(r)$	28
1.4 Risultati e discussione	31
1.4.1 Considerazioni generali sull'annealing simulato	31
1.4.2 Alcuni esempi	33
1.4.3 Il problema della trasferibilità	37
1.4.4 Proprietà dei cristalli	52
1.5 Conclusioni	57
1.6 Ringraziamenti	59
2 Monte Carlo Quantistico	61
2.1 Introduzione	62

2.2	Monte Carlo Variazionale	64
2.2.1	Minimizzazione della varianza	67
2.3	Monte Carlo di Diffusione	71
2.3.1	Funzione di prova	80
2.3.2	Bilancio dettagliato	81
2.3.3	L'algoritmo di Reynolds	82
2.3.4	L'algoritmo di Umrigar	84
2.4	Calcolo dei valori medi	87
2.5	Correlazione e calcolo delle incertezze	89
2.6	Risultati per sistemi a due elettroni	91
2.6.1	Hamiltoniano e funzione di prova	92
2.6.2	Tempo di correlazione e autocorrelazione.	93
2.6.3	Risultati per il VMC	96
2.6.4	Risultati per il DMC	99
2.6.5	Confronto fra diversi DMC	106
2.7	Conclusioni	111
2.8	Ringraziamenti	112
3	Calcolo parallelo	113
3.1	Introduzione	114
3.2	Calcolo parallelo	115
3.3	Architetture parallele SIMD e MIMD	117
3.4	Architetture parallele distribuite	118
3.5	Limite di velocità	120
3.6	Macchina Parallela Virtuale	122
3.7	VMC e DMC sulla macchina virtuale	126
3.7.1	Parallelizzazione del DMC	126
3.7.2	Parallelizzazione del VMC	131
3.8	Risultati per sistemi a due elettroni	132
3.8.1	Risultati delle simulazioni VMC	132
3.8.2	Risultati delle simulazioni DMC	134
3.8.3	Confronto fra diversi DMC	136
3.8.4	Tempi di calcolo ed efficienza	140
3.8.5	Analisi dell'esecuzione del programma parallelo sulla macchina virtuale	144
3.8.6	Confronto con APE100	152
3.9	Conclusioni	152
3.10	Ringraziamenti	155

Appendici	157
A Integrazione delle equazioni di Kohn-Sham	159
B Pseudohamiltoniani e Monte Carlo della Funzione di Green	161
C Funzione di Green per gli pseudopotenziali non locali	165
D Elementi di matrice delle onde piane per lo pseudohamiltoniano	169
Bibliografia	171

Indice delle Figure

1.1	Pseudohamiltoniano per Si	38
1.2	Pseudohamiltoniano per Na	39
1.3	Pseudohamiltoniani per C	40
1.4	Derivate logaritmiche per C	43
1.5	Derivate logaritmiche per Na	44
1.6	Derivate logaritmiche per Si	45
1.7	Scatola sferica per la funzione d'onda	50
1.8	Scatola sferica per la derivata della funzione d'onda	51
1.9	Occupazione variabile, carica totale costante	53
1.10	Occupazione variabile, carica totale variabile	54
2.1	Tempo di correlazione per il VMC	94
2.2	Tempo di correlazione e autocorrelazione lineare per il DMC	96
2.3	Energie DMC per lo ione Si^{2+}	100
2.4	Confronto fra energie sperimentali e DMC	107
2.5	Confronto fra le energie dei sistemi a uno e due elettroni	108
2.6	Confronto fra energie full-core e di sola valenza in DFT-LDA	109
2.7	Confronto fra energie DMC e DFT-LDA	110
3.1	Energie DMC per lo ione Si^{2+} in funzione di Δt	135
3.2	Energie DMC per gli ioni Na^- , Mg , Al^+ , P^{3+} , S^{4+} e Cl^{5+}	138
3.3	Energie DMC per lo ione Ar^{6+}	139
3.4	Inverso dei tempi di calcolo in funzione del numero di processori	141
3.5	Parte iniziale del programma DMC parallelo (rete Ethernet)	148
3.6	Fase di equilibratura del programma DMC parallelo (rete Ethernet)	149
3.7	Termine della fase di equilibratura e generazione dei processi di media nel programma DMC parallelo (rete Ethernet)	149
3.8	Fase di media del programma DMC parallelo (rete Ethernet)	150

3.9	Parte finale del programma DMC parallelo (rete Ethernet)	150
3.10	Parte iniziale del programma DMC parallelo (rete FDDI)	151
3.11	Termine della fase di equilibratura e generazione dei processi di media nel programma DMC parallelo (rete FDDI)	151

Indice delle Tavole

1.1	Coefficienti numerici dello pseudohamiltoniano per l'atomo di Na	34
1.2	Coefficienti numerici dello pseudohamiltoniano per l'atomo di Si	35
1.3	Energie atomiche per Si	36
1.4	Energie atomiche per Na	36
1.5	Energie atomiche per C	36
1.6	Confronto fra costanti reticolari ed energie coesive per Na	57
1.7	Proprietà del cristallo di Si	57
1.8	Proprietà del cristallo di Na	58
2.1	Energie variazionali per gli atomi neutri (He e gruppo IIA)	97
2.2	Energie variazionali per gli ioni negativi (gruppo IA)	97
2.3	Energie variazionali per gli ioni positivi (gruppo IIIB)	97
2.4	Energie variazionali per gli ioni positivi (gruppo IVB)	98
2.5	Energie variazionali per gli ioni positivi (gruppo VB)	98
2.6	Energie variazionali per gli ioni positivi (gruppo VIB)	98
2.7	Energie variazionali per gli ioni positivi (gruppo VIIB)	98
2.8	Energie variazionali per gli ioni positivi (gruppo VIII)	99
2.9	Energie di stato fondamentale per gli atomi neutri (He e gruppo IIA)	100
2.10	Energie di stato fondamentale per gli ioni negativi (gruppo IA)	101
2.11	Energie di stato fondamentale per gli ioni positivi (gruppo IIIB)	101
2.12	Energie di stato fondamentale per gli ioni positivi (gruppo IVB)	101
2.13	Energie di stato fondamentale per gli ioni positivi (gruppo VB)	102
2.14	Energie di stato fondamentale per gli ioni positivi (gruppo VIB)	102
2.15	Energie di stato fondamentale per gli ioni positivi (gruppo VIIB)	102
2.16	Energie di stato fondamentale per gli ioni positivi (gruppo VIII)	102
2.17	Affinità elettroniche per H e i metalli alcalini	103
3.1	Tempi di trasmissione su diversi tipi di rete nel caso del PVM	120

3.2	Architetture per le quali è stato sviluppato PVM 3.3.	123
3.3	Energie variazionali di stato fondamentale	133
3.4	Tempi di calcolo per il programma VMC	133
3.5	Energie DMC di stato fondamentale	134
3.6	Tempi di calcolo per il programma DMC	136
3.7	Energie DMC di stato fondamentale (algoritmi di Reynolds e di Umrigar) . .	137
3.8	Tempi di esecuzione del programma VMC su cluster e su APE100	153
3.9	Tempi di esecuzione del programma DMC su cluster e su APE100	153

Introduzione

Le simulazioni Monte Carlo Quantistico e in particolare i metodi Monte Carlo Variazionale e di Diffusione costituiscono il filo conduttore che sta alla base di questo lavoro di ricerca. La ragione dell'interesse che queste particolari tecniche di indagine numerica rivestono nell'ambito della fisica della materia computazionale sta nel fatto che esse permettono di descrivere con grande accuratezza l'interazione elettrone-elettrone e dunque gli effetti in cui la correlazione elettronica gioca un ruolo fondamentale, superando in questo modo le limitazioni imposte dalla descrizione approssimata fornita dalle teorie di campo medio autoconsistente quali la Teoria del Funzionale Densità e il metodo di Hartree-Fock[1]. Per raggiungere tale accuratezza, tuttavia, è necessario un elevato sforzo computazionale[2, 3] e questo in passato ha limitato l'utilizzo delle tecniche Monte Carlo Quantistico a sistemi estremamente semplici quali atomi e molecole leggere[4]. Il nostro studio si rivolge in particolare ad alcune delle metodologie introdotte per superare le limitazioni appena accennate. Da un lato, l'interesse per le sole proprietà di valenza dei sistemi studiati porta in maniera naturale all'idea di utilizzare operatori di sola valenza già impiegati con successo in connessione con i metodi di particella singola. In questo caso il problema è duplice in quanto si presenta innanzitutto la necessità di verificare se le proprietà matematiche degli operatori di sola valenza esistenti siano compatibili con i metodi Monte Carlo Quantistico considerati e in caso contrario nasce l'esigenza di costruirne di nuovi; in secondo luogo resta da stabilire se la descrizione approssimata che questi operatori danno delle proprietà fisiche nel caso dei metodi di singola particella risulti altrettanto soddisfacente anche quando essi vengano utilizzati in connessione con le tecniche Monte Carlo Quantistico. D'altro canto, l'uso di operatori di sola valenza, che riduce in maniera considerevole le limitazioni espresse all'inizio, può essere affiancato dall'impiego di nuove metodologie computazionali quali il calcolo parallelo e dunque rendere competitivi i metodi Monte Carlo Quantistico rispetto ad altre teorie a molti corpi.

Solo recentemente è stata aperta la strada allo studio di sistemi più complessi quali i solidi[5, 6, 7], grazie all'uso di hamiltoniani approssimati di sola valenza. Nel primo capitolo presenteremo il lavoro da noi intrapreso in questa direzione volto all'effettiva costruzione

di hamiltoniani locali di sola valenza proposti recentemente[6] e particolarmente adatti alle simulazioni Monte Carlo Quantistico in quanto si tratta di operatori locali. I risultati di questa ricerca sono l'oggetto di due lavori di cui uno pubblicato[8] e l'altro sottomesso per la pubblicazione[9].

Uno dei problemi connessi con l'impiego di hamiltoniani di sola valenza riguarda la loro origine. Infatti, poiché al momento non sono disponibili operatori generati consistentemente in ambito Monte Carlo Quantistico[10], si ricorre ad hamiltoniani costruiti all'interno dei metodi di campo medio autoconsistente[5, 6, 7] che trattano in maniera approssimata l'interazione elettrone-elettrone, e dunque sorge il problema di quantificare l'accuratezza con cui essi descrivono le proprietà di valenza di interesse. A questo proposito, dopo aver trattato in un certo dettaglio le tecniche Monte Carlo Variazionale e di Diffusione, nel secondo capitolo descriveremo i risultati di uno studio effettuato per un sistema semplice, un atomo o ione con due elettroni di valenza, da cui, però, è possibile ottenere informazioni di carattere generale e che, data la sua semplicità, ci ha permesso di procedere in maniera sistematica per una frazione rilevante degli elementi della Tavola Periodica. Il materiale presentato fa parte di due lavori, uno pubblicato[11] e l'altro in corso di sottomissione[12].

Una soluzione alternativa al problema dell'elevato onere computazionale, solo parzialmente mitigato dall'introduzione di hamiltoniani di sola valenza, viene dalle nuove tecniche di calcolo parallelo a cui le simulazioni Monte Carlo Quantistico si prestano particolarmente bene. La realizzazione e la convenienza di questo tipo di approccio sono state affrontate nel terzo capitolo usando come sistema di prova e di riferimento lo stesso sistema di sola valenza a due elettroni considerato nel secondo capitolo. Ancora una volta la semplicità del sistema trattato, che, però, pone le stesse problematiche di sistemi più complessi, è importante in quanto facilita il compito della messa a punto della parte computazionale parallela secondo un modello del tutto generale e indipendente dal particolare sistema scelto. Il lavoro presentato, da poco concluso, è stato oggetto di due Tesi di laurea[13, 14] ed è stato presentato a una conferenza internazionale[15].

1 Operatore pseudohamiltoniano

1.1 Introduzione

Nelle tecniche Monte Carlo della funzione di Green (GFMC)[16], la forte dipendenza dal numero atomico Z del tempo di simulazione richiesto per ottenere errori statistici sufficientemente piccoli nel calcolo delle quantità fisiche rilevanti per un sistema atomico[2], pone un limite superiore molto piccolo al numero di elettroni che possono essere trattati contemporaneamente nelle simulazioni quantistiche di sistemi atomici, molecolari e di stato solido[3, 5, 7]. Osservando che molte proprietà di interesse non dipendono direttamente dai *core* atomici (che sono quasi identici in atomi, molecole e solidi), arriviamo in maniera naturale all'idea di un hamiltoniano di sola valenza che sostituisca l'hamiltoniano vero o *full-core* nella descrizione delle proprietà di valenza del sistema. Questa idea, sebbene non nuova[17], rappresenta un campo in continua evoluzione nella fisica dello stato solido[18], dove risultati importanti basati su diversi metodi di indagine e nuovi sviluppi teorici[19] continuano a fiorire. Sfortunatamente gli hamiltoniani di valenza oggi disponibili, che derivano dagli pseudopotenziali a norma conservata introdotti da Hamann, Schlüter, e Chiang nel 1979[20], non sono utilizzabili in connessione con le tecniche GFMC senza ulteriori approssimazioni (Sez. 1.2.1 e Appendice C)[7]. La necessità di un operatore di sola valenza specifico per il GFMC ha condotto Bachelet, Ceperley, e Chiochetti[6] allo sviluppo di uno pseudohamiltoniano locale (Sez. 1.2). Fino ad ora, comunque, sono stati costruiti solo pochi pseudohamiltoniani atomici. Inoltre, Foulkes e Schlüter[21] hanno dimostrato che solo gli elementi per i quali l'ordine degli autovalori è quello "naturale", $\epsilon_s \leq \epsilon_p \leq \epsilon_d \leq \dots$, possono essere descritti da uno pseudohamiltoniano *locale* (questo esclude i metalli di transizione). Nonostante queste limitazioni, però, tale operatore è stato per lungo tempo l'unica possibilità per lo studio di sistemi di stato solido mediante GFMC. Per esempio, è stato usato con successo da Li, Ceperley, e Martin[22] nella prima simulazione GFMC di un sistema esteso (il cristallo di Si), un calcolo molto al di là delle possibilità dei calcolatori dell'ultima generazione senza l'impiego di uno pseudohamiltoniano. Poiché gli pseudohamiltoniani sono, al momento, lo strumento più pratico per eseguire simulazioni GFMC per sistemi che con-

tengano atomi più pesanti del Neon, ci è sembrata una buona idea ottenere pseudohamiltoniani accurati per molti più elementi in quelle regioni della Tavola Periodica dove il metodo funziona bene, lasciando a metodi alternativi, che sono più elaborati e introducono ulteriori approssimazioni, il compito di trattare quegli elementi per cui gli pseudohamiltoniani sono impossibili o molto più difficili da costruire[7]. Per la costruzione degli pseudohamiltoniani (così come per gli pseudopotenziali non locali) ricorriamo alla teoria del funzionale densità (DFT-LDA)[23] e alle idee principali delineate nel Rif. [20], opportunamente estese per il nostro scopo (Sez. 1.2.3 e 1.3). Abbiamo generato vari pseudohamiltoniani per la seconda e la terza riga dei gruppi IIIB, IVB, VB, VIB, e VIIB, così come per i metalli alcalini Li e Na e un elemento della prima riga, il C (Sez. 1.4.2). Se si renderanno disponibili accurati orbitali naturali[10] per un numero sempre maggiore di elementi, allora il metodo presentato avrà un'immediata estensione all'atomo descritto da una vera teoria a molti corpi. Per gli atomi considerati in questo lavoro, comunque, gli ioni descritti mediante DFT-LDA dovrebbero costituire un'eccellente approssimazione[24]. Abbiamo verificato la trasferibilità dei nuovi operatori generati in diversi modi (sempre all'interno della DFT-LDA) facendo dei controlli più severi di quelli usualmente effettuati per gli pseudopotenziali a norma conservata (Sez. 1.4.3). I risultati (Sez. 1.4) sono estremamente positivi nella maggior parte dei casi; per il nostro atomo rappresentativo della prima riga, il carbonio, questi ultimi non sono della stessa qualità. I nostri risultati confermano inoltre l'osservazione, recentemente enfatizzata da Teter[25], che i controlli di routine sulle derivate logaritmiche non sempre forniscono informazioni affidabili sulla trasferibilità, e altri controlli, che coinvolgono la risposta auto-consistente del sistema, risultano più appropriati (Sez. 1.4.3). Per quanto riguarda la *portabilità* (e cioè il grado di accuratezza con cui gli pseudohamiltoniani possono essere impiegati come ioni "fisici" al di fuori della teoria DFT-LDA[24]), ci basiamo sugli incoraggianti risultati ottenuti nel Rif. [6] per atomi e molecole e nel Rif.[22] per il cristallo di Si, e lasciamo ulteriori controlli di portabilità per il futuro. Alla fine di questa introduzione, val la pena di sottolineare che, al di là dell'interesse che gli pseudohamiltoniani hanno nelle simulazioni GFMC, essi possono sostituire opportunamente gli pseudopotenziali non locali

in connessione con molti altri metodi di indagine[21], dato che la loro manipolazione è in generale più semplice di quella degli pseudopotenziali non locali (Appendice D): per esempio, essi sono stati usati recentemente per lo studio di *cluster* di metalli alcalini[26].

Le unità di misura usate ovunque in questo lavoro (a meno che non sia specificato diversamente) sono unità atomiche.

1.2 Teoria

1.2.1 Lo pseudohamiltoniano

La ragione per cui non è possibile un uso immediato degli pseudopotenziali non locali nei metodi GFMC è legata alla necessità di avere una funzione di Green positiva e alla cosiddetta approssimazione a nodi fissi[4]. Descriveremo più in dettaglio il GFMC e la sua approssimazione a nodi fissi nel capitolo successivo, ma ora abbiamo bisogno di richiamarne alcuni aspetti essenziali per lo sviluppo dello pseudohamiltoniano[6, 27]. Nel GFMC a nodi fissi abbiamo a che fare con il seguente problema alle condizioni al contorno (usiamo la notazione \hat{H}^{PS} per l'hamiltoniano elettronico dato che abbiamo sempre a che fare con operatori *pseudo* o di sola valenza)

$$\begin{cases} \hat{H}^{PS}\Psi(\mathbf{R}) = E^{FN}\Psi(\mathbf{R}) & \forall \mathbf{R} \in D \\ \Psi(\mathbf{R}) = 0 & \forall \mathbf{R} \in \partial D, \end{cases} \quad (1.1)$$

e cerchiamo l'autofunzione di energia minima $\Psi(\mathbf{R})$ che non cambia mai segno¹ nella regione D dello spazio delle configurazioni elettronico e che si annulla sulla sua frontiera ∂D . Il problema è ben definito se gli elementi di matrice $\langle \mathbf{R} | \hat{H}^{PS} | \mathbf{R}' \rangle$ si annullano per $\mathbf{R} \neq \mathbf{R}'$ o, in altre parole, se \hat{H}^{PS} è un operatore locale; altrimenti da

$$\int \langle \mathbf{R} | \hat{H}^{PS} | \mathbf{R}' \rangle \Psi(\mathbf{R}') d\mathbf{R}' = E^{FN} \Psi(\mathbf{R}) \quad (1.2)$$

segue che avremmo bisogno di specificare $\Psi(\mathbf{R})$ fuori dalla regione D per conoscere l'azione di \hat{H}^{PS} su $\Psi(\mathbf{R})$ al suo interno. L'energia corrispondente allo stato fondamentale del problema alle condizioni al contorno E_0^{FN} è un limite superiore per l'energia E_0 dello stato fermio-

¹Consideriamo solo operatori \hat{H}^{PS} reali e perciò possiamo scegliere sempre funzioni d'onda reali[28].

nico vero, non vincolato, dell'hamiltoniano \hat{H}^{PS} in tutto lo spazio; quando ∂D si avvicina alla superficie nodale esatta dello stato fondamentale fermionico, $E_0^{FN} - E_0$ si annulla quadraticamente[4]; questo significa che per un operatore locale \hat{H}^{PS} , l'approssimazione a nodi fissi ha l'importante proprietà di essere variazionale e, siccome ∂D è definita dai nodi di una funzione d'onda di prova approssimata $\Psi_T(\mathbf{R})$ tale che $\Psi_T(\mathbf{R}) > 0 \forall \mathbf{R} \in D - \partial D$, $\Psi_T(\mathbf{R}) = 0 \forall \mathbf{R} \in \partial D$, solo i nodi di $\Psi_T(\mathbf{R})$ sono importanti per E_0^{FN} , non il suo valore in qualunque altro punto. Un'altra ragione fondamentale per avere un hamiltoniano locale \hat{H}^{PS} viene da un'ipotesi basilare del GFMC: la funzione di Green

$$G(\mathbf{R}, \mathbf{R}', \tau) = \Psi_T(\mathbf{R}) \Psi_T^{-1}(\mathbf{R}') \langle \mathbf{R} | \exp(-\tau \hat{H}^{PS}) | \mathbf{R}' \rangle \quad (1.3)$$

deve essere interpretata come una densità di probabilità condizionale per ogni scelta di Ψ_T , se vogliamo far uso del risultato statistico che l'errore decresce in maniera inversamente proporzionale alla radice quadrata del numero di passi. Questo significa che $G(\mathbf{R}, \mathbf{R}', \tau)$ deve essere non negativa $\forall \mathbf{R}, \mathbf{R}' \in D$ e per ogni $\tau > 0$: questo è automatico per hamiltoniani “normali” formati dall'operatore energia cinetica standard più un potenziale locale[29], ma, sfortunatamente, non è così per gli pseudopotenziali non locali (vedere Appendice C); anche per operatori locali più generali la funzione di Green può non soddisfare questo requisito di positività. Sotto quali condizioni un operatore locale ha una funzione di Green positiva? Per rispondere a questa domanda richiamiamo un ragionamento noto[27]: innanzitutto espandiamo il generico hamiltoniano locale \hat{H}^{PS} in potenze dell'operatore momento coniugato a \mathbf{R} , $\hat{\mathbf{P}} = -i\nabla_{\mathbf{R}}$,

$$\hat{H}^{PS} = W(\mathbf{R}) + \sum_{\alpha=1}^{3N} K_{\alpha}(\mathbf{R}) \hat{P}_{\alpha} + \sum_{\alpha, \beta=1}^{3N} T_{\alpha\beta}(\mathbf{R}) \hat{P}_{\alpha} \hat{P}_{\beta} + \dots, \quad (1.4)$$

e poi consideriamo il tensore

$$c_{\alpha\beta}(\mathbf{R}_0, \tau) = \int (R - R_0)_{\alpha} (R - R_0)_{\beta} G(\mathbf{R}, \mathbf{R}_0, \tau) d\mathbf{R}. \quad (1.5)$$

Se G deve essere non negativa $\forall \mathbf{R}, \mathbf{R}' \in D$ e per ogni $\tau > 0$, allora $c_{\alpha\beta}$ deve essere definito positivo per ogni $\tau > 0$. Per $\tau = 0$ abbiamo

$$G(\mathbf{R}, \mathbf{R}_0, \tau = 0) = \Psi_T(\mathbf{R}) \Psi_T^{-1}(\mathbf{R}_0) \delta(\mathbf{R} - \mathbf{R}_0), \quad (1.6)$$

$$c_{\alpha\beta}(\mathbf{R}_0, \tau = 0) = 0. \quad (1.7)$$

Calcoliamo la derivata temporale di $c_{\alpha\beta}$ per $\tau = 0$

$$\begin{aligned} \left. \frac{\partial c_{\alpha\beta}}{\partial \tau} \right|_{\tau=0} &= \int (R - R_0)_\alpha (R - R_0)_\beta \left. \frac{\partial G}{\partial \tau}(\mathbf{R}, \mathbf{R}_0, \tau) \right|_{\tau=0} d\mathbf{R} = \\ &= -\Psi_T^{-1}(\mathbf{R}_0) \int (R - R_0)_\alpha (R - R_0)_\beta \Psi_T(\mathbf{R}) \langle \mathbf{R}_0 | \hat{H}^{PS} | \mathbf{R} \rangle d\mathbf{R} = \\ &= -\Psi_T^{-1}(\mathbf{R}_0) \int \delta(\mathbf{R} - \mathbf{R}_0) \hat{H}^{PS} [(R - R_0)_\alpha (R - R_0)_\beta \Psi_T(\mathbf{R})] d\mathbf{R}, \end{aligned} \quad (1.8)$$

dove abbiamo usato il fatto che \hat{H}^{PS} è hermitiano e reale e dunque possiamo sempre scegliere funzioni d'onda reali. Usando ripetutamente la relazione

$$[R_\alpha, f(\hat{\mathbf{P}})] = i \frac{\partial f}{\partial \hat{P}_\alpha}, \quad (1.9)$$

arriviamo a

$$\begin{aligned} \left. \frac{\partial c_{\alpha\beta}}{\partial \tau} \right|_{\tau=0} &= \Psi_T^{-1}(\mathbf{R}_0) \int \delta(\mathbf{R} - \mathbf{R}_0) \frac{\partial^2 \hat{H}^{PS}}{\partial \hat{P}_\alpha \partial \hat{P}_\beta} \Psi_T(\mathbf{R}) d\mathbf{R} = \\ &= \Psi_T^{-1}(\mathbf{R}_0) \left[\frac{\partial^2 \hat{H}^{PS}}{\partial \hat{P}_\alpha \partial \hat{P}_\beta} \Psi_T(\mathbf{R}) \right]_{\mathbf{R}=\mathbf{R}_0}. \end{aligned} \quad (1.10)$$

Questa espressione può essere scelta definita positiva per ogni funzione d'onda di prova Ψ_T solo se \hat{H}^{PS} è della forma

$$\hat{H}^{PS} = W(\mathbf{R}) + \sum_{\alpha=1}^{3N} K_\alpha(\mathbf{R}) \hat{P}_\alpha + \sum_{\alpha,\beta=1}^{3N} T_{\alpha\beta}(\mathbf{R}) \hat{P}_\alpha \hat{P}_\beta, \quad (1.11)$$

da cui

$$\left. \frac{\partial c_{\alpha\beta}}{\partial \tau} \right|_{\tau=0} = \Psi_T^{-1}(\mathbf{R}_0) T_{\alpha\beta}(\mathbf{R}_0) \Psi_T(\mathbf{R}_0) = T_{\alpha\beta}(\mathbf{R}_0). \quad (1.12)$$

In conclusione, l'hamiltoniano locale da usare nel GFMC deve avere la forma dell'Eq. (1.11), con il tensore Hessiano $T_{\alpha\beta}(\mathbf{R}_0)$ definito positivo per ogni $\mathbf{R}_0 \in D$. Questo garantisce che anche $\partial c_{\alpha\beta}(\mathbf{R}_0, \tau) / \partial \tau$ e $c_{\alpha\beta}(\mathbf{R}_0, \tau)$ sono definiti positivi $\forall \mathbf{R}_0 \in D$ in un intorno destro di $\tau = 0$, e dunque è condizione necessaria per avere $G(\mathbf{R}, \mathbf{R}', \tau)$ non negativa nello stesso intorno. Per quanto riguarda la prova della sufficienza della condizione appena enunciata

osserviamo che basta una dimostrazione valida per piccoli $\tau > 0$ in quanto da ciò segue la dimostrazione per ogni $\tau > 0$ dato che

$$G(\mathbf{R}, \mathbf{R}_0, \tau_1 + \tau_2) = \int G(\mathbf{R}, \mathbf{R}', \tau_2) G(\mathbf{R}', \mathbf{R}_0, \tau_1) d\mathbf{R}'. \quad (1.13)$$

Purtroppo, a tutt'oggi abbiamo una dimostrazione per il solo caso in cui $T_{\alpha\beta}$ sia indipendente da \mathbf{R} [29] mentre nel caso generale siamo in grado di dimostrare solo la positività dell'approssimazione a tempi corti per la funzione di Green esatta, $G(\mathbf{R}, \mathbf{R}', \tau)$, che in realtà è sufficiente per le applicazioni pratiche (vedere Appendice B). Per fare questo, però, ci limitiamo alla forma di \hat{H}^{PS} data nel Rif. [6], per la quale abbiamo bisogno di altre due specializzazioni. Innanzitutto richiediamo che \hat{H}^{PS} abbia la stessa struttura dell'hamiltoniano vero per un insieme di atomi, e cioè

$$\begin{aligned} \hat{H}^{PS} &= -\frac{1}{2} \sum_i \nabla_i^2 + \frac{1}{2} \sum_{i,j \neq i} \frac{1}{r_{ij}} + \sum_{i,I} \hat{v}_I^{PS}(\mathbf{r}_{iI}) = \\ &= \sum_i \hat{h}_i^{PS} + \frac{1}{2} \sum_{i,j \neq i} \frac{1}{r_{ij}}, \end{aligned} \quad (1.14)$$

dove i, j indicano gli elettroni di valenza, I gli ioni, $r_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$, $\mathbf{r}_{iI} = \mathbf{r}_i - \mathbf{s}_I$, $\{\mathbf{r}_i\}$ sono le coordinate elettroniche, e $\{\mathbf{s}_I\}$ quelle ioniche. \hat{v}_I^{PS} descrive l'interazione elettronica con l' I -esimo ione, e imponiamo che \hat{v}_I^{PS} abbia la stessa simmetria sferica dell'interazione vera intorno all' I -esimo ione; allora, se ci concentriamo solo su uno ione e un elettrone e omettiamo gli indici I e i , l'operatore \hat{h}^{PS} , Eq. (1.14), (usiamo lettere minuscole per enfatizzare la sua natura di operatore a una particella), scrivendo ogni termine in forma esplicitamente hermitiana, diventa

$$\begin{aligned} \hat{h}^{PS} &= \frac{1}{2} \hat{p}^2 + \hat{v}^{PS}(\mathbf{r}) = \\ &= w(\mathbf{r}) + \sum_{\alpha=1}^3 [k_{\alpha}(\mathbf{r}) \hat{p}_{\alpha} + \hat{p}_{\alpha} k_{\alpha}(\mathbf{r})] + \frac{1}{2} \sum_{\alpha,\beta=1}^3 \hat{p}_{\alpha} t_{\alpha\beta}(\mathbf{r}) \hat{p}_{\beta}, \end{aligned} \quad (1.15)$$

dove $\hat{\mathbf{p}} = -i\nabla_{\mathbf{r}}$ e $t_{\alpha\beta}$ è simmetrico e definito positivo. I termini lineari in $\hat{\mathbf{p}}$, Eq. (1.15), possono essere omessi se richiediamo che \hat{h}^{PS} sia invariante per inversione temporale[30], e cioè per la trasformazione $(\mathbf{r}, \hat{\mathbf{p}}) \rightarrow (\mathbf{r}, -\hat{\mathbf{p}})$, e otteniamo

$$\hat{h}^{PS} = w(\mathbf{r}) + \frac{1}{2} \sum_{\alpha,\beta=1}^3 \hat{p}_{\alpha} t_{\alpha\beta}(\mathbf{r}) \hat{p}_{\beta}. \quad (1.16)$$

Osserviamo che l'operatore

$$\frac{1}{2} \sum_{\alpha, \beta=1}^3 \hat{p}_\alpha t_{\alpha\beta}(\mathbf{r}) \hat{p}_\beta \quad (1.17)$$

è positivo e di conseguenza inferiormente limitato e il suo limite inferiore è non negativo[31].

Infatti, poiché il tensore $t_{\alpha\beta}(\mathbf{r})$ è definito positivo per tutti i valori di \mathbf{r} , per ogni vettore $\mathbf{v} \in \mathbf{R}^3$ di lunghezza $v > 0$ risulta

$$\sum_{\alpha, \beta=1}^3 v_\alpha t_{\alpha\beta}(\mathbf{r}) v_\beta > 0, \quad (1.18)$$

e presa una qualunque funzione $\phi(\mathbf{r})$ a norma finita si ha

$$\begin{aligned} & \int d\mathbf{r} \phi(\mathbf{r}) \sum_{\alpha, \beta=1}^3 \hat{p}_\alpha t_{\alpha\beta}(\mathbf{r}) \hat{p}_\beta \phi(\mathbf{r}) = \\ &= - \int d\mathbf{r} \phi(\mathbf{r}) \sum_{\alpha, \beta=1}^3 \nabla_\alpha t_{\alpha\beta}(\mathbf{r}) \nabla_\beta \phi(\mathbf{r}) = \\ &= \int d\mathbf{r} \sum_{\alpha, \beta=1}^3 \nabla_\alpha \phi(\mathbf{r}) t_{\alpha\beta}(\mathbf{r}) \nabla_\beta \phi(\mathbf{r}) \geq 0. \end{aligned} \quad (1.19)$$

Le espressioni a simmetria sferica più generali per w e $t_{\alpha\beta}$ sono

$$w(\mathbf{r}) = w(r), \quad (1.20)$$

$$t_{\alpha\beta}(\mathbf{r}) = t_1(r) \delta_{\alpha\beta} + \frac{t_2(r)}{r^2} r_\alpha r_\beta. \quad (1.21)$$

Per avere $t_{\alpha\beta}$ definito positivo, t_1 e t_2 sono soggette a dei vincoli. Il nostro $t_{\alpha\beta}$ è dato dall'Eq. (1.21), così nel nostro caso la condizione Eq. (1.18) diventa

$$\sum_{\alpha, \beta=1}^3 v_\alpha t_{\alpha\beta} v_\beta = t_1(r) v^2 + t_2(r) v^2 \left(\frac{\mathbf{r}\mathbf{v}}{rv} \right)^2,$$

da cui segue

$$\begin{aligned} & t_1(r) + \lambda t_2(r) > 0 \quad \forall \lambda \in [0, 1], \\ \Rightarrow & \begin{cases} t_1(r) > 0 \\ t_1(r) + t_2(r) > 0. \end{cases} \end{aligned} \quad (1.22)$$

Abbiamo

$$\hat{h}^{PS} = w(\mathbf{r}) + \frac{1}{2} \sum_{\alpha, \beta=1}^3 \hat{p}_\alpha \left[t_1(r) \delta_{\alpha\beta} + \frac{t_2(r)}{r^2} r_\alpha r_\beta \right] \hat{p}_\beta, \quad (1.23)$$

$$\Rightarrow \hat{h}^{PS} = w(\mathbf{r}) + \frac{1}{2} \hat{\mathbf{p}} t_1(r) \hat{\mathbf{p}} + \frac{1}{2} (\hat{\mathbf{p}} \mathbf{r}) \frac{t_2(r)}{r^2} (\mathbf{r} \hat{\mathbf{p}}). \quad (1.24)$$

Usando l'uguaglianza [28, 32]

$$(\hat{\mathbf{p}}\mathbf{r})f(r)(\mathbf{r}\hat{\mathbf{p}}) = \hat{\mathbf{p}}r^2f(r)\hat{\mathbf{p}} - \hat{\mathbf{L}}f(r)\hat{\mathbf{L}}, \quad (1.25)$$

dove $\hat{\mathbf{L}} = \mathbf{r} \wedge \hat{\mathbf{p}}$, possiamo scrivere

$$\hat{h}^{PS} = w(\mathbf{r}) + \frac{1}{2}\hat{\mathbf{p}}[t_1(r) + t_2(r)]\hat{\mathbf{p}} - \frac{1}{2}\frac{t_2(r)}{r^2}\hat{L}^2, \quad (1.26)$$

dove abbiamo usato il fatto che una funzione puramente radiale ha simmetria sferica e così commuta con ogni componente di $\hat{\mathbf{L}}$. Questa è l'espressione per \hat{h}^{PS} data nel Rif. [6], posto di definire

$$A(r) = t_1(r) + t_2(r) - 1, \quad (1.27)$$

$$B(r) = -t_2(r), \quad (1.28)$$

$$v_{ion}(r) = w(r), \quad (1.29)$$

$$\Rightarrow \hat{h}^{PS} = -\frac{1}{2}\nabla[A(r) + 1]\nabla + \frac{B(r)}{2r^2}\hat{L}^2 + v_{ion}(r), \quad (1.30)$$

con le condizioni

$$\begin{cases} A(r) + 1 > 0 \\ A(r) + B(r) + 1 > 0. \end{cases} \quad (1.31)$$

A causa dell'Eq. (1.31), se la funzione $v_{ion}(r)$ è finita per ogni valore di $r > 0$ l'operatore \hat{h}^{PS} è inferiormente limitato. Tornando alle coordinate cartesiane con le derivate tutte a destra, abbiamo

$$\begin{aligned} \hat{h}^{PS} = & -\frac{1}{2}\sum_{\alpha,\beta=1}^3 \left\{ [A(r) + B(r) + 1]\delta_{\alpha\beta} - \frac{B(r)}{r^2}r_\alpha r_\beta \right\} \frac{\partial}{\partial r_\alpha} \frac{\partial}{\partial r_\beta} + \\ & + \sum_{\alpha=1}^3 \left[-\frac{A'(r)}{2} + \frac{B(r)}{r} \right] \frac{r_\alpha}{r} \frac{\partial}{\partial r_\alpha} + v_{ion}(r). \end{aligned} \quad (1.32)$$

La funzione di Green per lo pseudohamiltoniano (nell'approssimazione a tempi corti) è considerata in Appendice B e osserviamo che essa è positiva come discusso precedentemente.

1.2.2 Le equazioni di Kohn-Sham

Scriviamo ora le equazioni di Kohn-Sham nella DFT-LDA sia per l'hamiltoniano full-core che per lo pseudohamiltoniano in approssimazione sferica (la loro soluzione è trattata in dettaglio

in Appendice A). Siano $\epsilon_{n\ell}$ e $\chi_{n\ell}(r)$ gli autovalori a singola particella e le autofunzioni radiali di momento angolare ℓ e di numero quantico principale n (nel seguito n verrà ommesso dato che per ogni valore del momento angolare ℓ , ci concentriamo su un solo stato di valenza), e indichiamo le quantità full-core con l'indice FC (questa notazione verrà usata ovunque in questo capitolo). Abbiamo allora

$$-\frac{1}{2} \frac{d^2 \chi_{n\ell}^{FC}(r)}{dr^2} + \left[\frac{\ell(\ell+1)}{2r^2} - \frac{Z}{r} + v_{HXC}^{FC}(r) \right] \chi_{n\ell}^{FC}(r) = \epsilon_{n\ell}^{FC} \chi_{n\ell}^{FC}(r), \quad (1.33)$$

$$\begin{aligned} -\frac{1}{2} [A(r) + 1] \frac{d^2 \chi_{n\ell}(r)}{dr^2} + \left\{ [A(r) + B(r) + 1] \frac{\ell(\ell+1)}{2r^2} - \frac{1}{2} \frac{dA}{dr} \left[\frac{d}{dr} - \frac{1}{r} \right] + \right. \\ \left. + v_{ion}(r) + v_{HXC}(r) \right\} \chi_{n\ell}(r) = \epsilon_{n\ell} \chi_{n\ell}(r). \end{aligned} \quad (1.34)$$

Quando gli effetti relativistici sono importanti, sostituiamo l'Eq. (1.33) con un'equazione di Dirac scalar relativistica[33]. $v_{HXC}(r)$ è il potenziale di Hartree, scambio e correlazione, definito come

$$v_{HXC}(r) = \int \frac{n(r')}{|\mathbf{r} - \mathbf{r}'|} d^3 \mathbf{r}' + \frac{\delta E_{XC}^{LDA}[n]}{\delta n(r)}, \quad (1.35)$$

$$n(r) \equiv \frac{1}{4\pi r^2} \sum_{n\ell} f_{n\ell} |\chi_{n\ell}(r)|^2. \quad (1.36)$$

$E_{XC}^{LDA}[n]$ è il funzionale energia di scambio e correlazione della DFT-LDA[34, 35], $n(r)$ la densità di carica e $f_{n\ell}$ i numeri di occupazione degli orbitali di Kohn-Sham radiali $\chi_{n\ell}(r)$ [23]. Per semplificare la notazione, introduciamo le funzioni $a(r)$, $c(r)$ e $v(r)$, definite come segue

$$a(r) = A(r) + 1, \quad (1.37)$$

$$c(r) = A(r) + B(r) + 1, \quad (1.38)$$

$$v(r) = v_{ion}(r) + v_{HXC}(r), \quad (1.39)$$

$$v^{FC}(r) = -\frac{Z}{r} + v_{HXC}^{FC}(r). \quad (1.40)$$

Allora l'Eq. (1.34) diventa:

$$-\frac{1}{2} a(r) \frac{d^2 \chi_{\ell}(r)}{dr^2} + \left\{ c(r) \frac{\ell(\ell+1)}{2r^2} - \frac{1}{2} \frac{da}{dr} \left[\frac{d}{dr} - \frac{1}{r} \right] + v(r) \right\} \chi_{\ell}(r) = \epsilon_{\ell} \chi_{\ell}(r). \quad (1.41)$$

Come menzionato brevemente nell'Introduzione, Foulkes e Schlüter hanno dimostrato che il nostro operatore da origine a un ordine “naturale” $\epsilon_s \leq \epsilon_p \leq \epsilon_d \leq \dots$ nelle energie degli autovalori[21]. Può essere utile, basandoci sull'Eq. (1.41), richiamare il loro ragionamento in maniera dettagliata. Se moltiplichiamo l'Eq. (1.41) per $\chi_{\ell'}(r)$ a sinistra, la sottraiamo a se stessa dopo lo scambio di ℓ con ℓ' e integriamo il risultato da $r = 0$ a $+\infty$, otteniamo:

$$\begin{aligned} & \frac{1}{2} \int_0^{+\infty} \chi_{\ell}(r) \frac{d}{dr} \left[a(r) \frac{d\chi_{\ell'}}{dr}(r) \right] dr - \frac{1}{2} \int_0^{+\infty} \chi_{\ell'}(r) \frac{d}{dr} \left[a(r) \frac{d\chi_{\ell}}{dr}(r) \right] dr + \\ & + \frac{1}{2} [\ell(\ell+1) - \ell'(\ell'+1)] \int_0^{+\infty} c(r) \frac{\chi_{\ell}(r) \chi_{\ell'}(r)}{r^2} dr = \\ & = (\epsilon_{\ell} - \epsilon_{\ell'}) \int_0^{+\infty} \chi_{\ell}(r) \chi_{\ell'}(r) dr. \end{aligned} \quad (1.42)$$

Riarrangiando il primo termine a sinistra

$$\begin{aligned} & \int_0^{+\infty} \chi_{\ell}(r) \frac{d}{dr} \left[a(r) \frac{d\chi_{\ell'}}{dr}(r) \right] dr = \\ & = \left[\chi_{\ell}(r) a(r) \frac{d\chi_{\ell'}}{dr}(r) \right]_0^{+\infty} - \int_0^{+\infty} \frac{d\chi_{\ell}}{dr}(r) a(r) \frac{d\chi_{\ell'}}{dr}(r) dr = \\ & = - \left[\frac{d\chi_{\ell}}{dr}(r) a(r) \chi_{\ell'}(r) \right]_0^{+\infty} + \int_0^{+\infty} \chi_{\ell'}(r) \frac{d}{dr} \left[a(r) \frac{d\chi_{\ell}}{dr}(r) \right] dr = \\ & = \int_0^{+\infty} \chi_{\ell'}(r) \frac{d}{dr} \left[a(r) \frac{d\chi_{\ell}}{dr}(r) \right] dr, \end{aligned} \quad (1.43)$$

dove abbiamo usato il fatto che $\chi_{\ell}(0) = 0$ e $\chi_{\ell}(+\infty) = 0$ (vedere Appendice A), l'Eq. (1.42) diventa

$$(\epsilon_{\ell} - \epsilon_{\ell'}) \int_0^{+\infty} \chi_{\ell}(r) \chi_{\ell'}(r) dr = \frac{1}{2} [\ell(\ell+1) - \ell'(\ell'+1)] \int_0^{+\infty} c(r) \frac{\chi_{\ell}(r) \chi_{\ell'}(r)}{r^2} dr. \quad (1.44)$$

$c(r)$ è positiva per ogni $r > 0$, e se prendiamo per χ_{ℓ} e $\chi_{\ell'}$ gli stati di energia più bassa con momento angolare ℓ e ℓ' , che sono senza nodi e possono essere scelti reali e positivi $\forall r > 0$, ne segue

$$\epsilon_{\ell} > \epsilon_{\ell'} \Leftrightarrow \ell > \ell' \quad \forall \ell, \ell', \quad (1.45)$$

che da la regola di ordinamento[21].

1.2.3 La conservazione della norma

Siamo ora pronti per generalizzare il metodo del Rif. [20] per la costruzione degli pseudo-hamiltoniani. Partiamo da uno stato di riferimento atomico che contiene tutti gli orbitali il

cui momento angolare è rilevante per le proprietà di valenza dei sistemi di interesse. Questa è una delle differenze principali rispetto agli pseudopotenziali non locali: abbiamo bisogno di tutti i momenti angolari contemporaneamente perché lo pseudohamiltoniano, essendo locale, agisce su tutti i momenti angolari allo stesso tempo, piuttosto di selezionarli uno per volta come gli pseudopotenziali non locali del tipo Hamann-Schlüter-Chiang attraverso gli operatori di proiezione. In pratica $\ell = 0, 1$ (onde s e p) o $\ell = 0, 1, 2$ (onde s , p e d) sono sufficienti per caratterizzare la maggior parte degli elementi interessanti della Tavola Periodica. Scegliamo un unico raggio di core r_c , lo stesso per ognuna delle tre funzioni $a(r)$, $c(r)$ e $v(r)$, e lo usiamo in un modo molto più simile a quello di Kerker[36] che non a quello di Hamann, Schlüter, e Chiang. Dato lo stato di riferimento e il raggio di core possiamo scrivere le quattro richieste principali per il nostro pseudohamiltoniano; tre di queste sono le stesse della procedura delineata nel Rif. [20], la quarta è specifica per lo pseudohamiltoniano:

1. autovalori di valenza (di singola particella) pseudo e full-core identici (P1);
2. orbitali di valenza (di singola particella) pseudo correttamente normalizzati che siano identici a quelli full-core fuori dal core (conservazione della norma) (P2);
3. orbitali di valenza senza nodi (P3);
4. condizioni di “positività”: $A(r) + 1 > 0$ e $A(r) + B(r) + 1 > 0$ (P4).

Una conseguenza immediata della proprietà (P3) è che per ogni momento angolare ℓ il nostro orbitale di valenza è l'autostato di più bassa energia dell'Eq. (1.41). Ciò dimostra anche che non è possibile fare nulla per sovvertire la regola di ordinamento, Eq. (1.45): le uniche possibilità sono di rinunciare o alla proprietà (P3), che introdurrebbe stati di core (pseudo) non fisici, o alla condizione (P4) ($c(r) > 0 \forall r > 0$), che, invece, violerebbe le richieste dell'approssimazione a nodi fissi nel GFMC e renderebbe lo spettro di \hat{h}^{PS} non limitato inferiormente. Quindi, la regola di ordinamento per la prima volta dimostrata da Foulkes e Schlüter[21] pone delle severe limitazioni per la descrizione degli elementi di transizione per i quali l'ordinamento energetico degli stati di valenza segue raramente l'Eq. (1.45): in questo

caso l'atomo di sola valenza non può essere riprodotto da nessuno pseudohamiltoniano che soddisfi le proprietà da (P1) a (P4).

Le proprietà (P1) e (P2) implicano il seguente andamento asintotico per $a(r)$, $c(r)$ e $v(r)$:

$$\begin{cases} a(r) = 1 \\ c(r) = 1 \\ v(r) = v^{FC}(r). \end{cases} \quad \forall r \geq r_c \quad (1.46)$$

In analogia con il metodo di Hamann, Schlüter, e Chiang, si può dimostrare che le proprietà (P1) e (P2) implicano l'uguaglianza fino al primo ordine in energia delle derivate logaritmiche pseudo e full-core delle funzioni d'onda di singola particella. Se procediamo in modo simile a quanto descritto nel Rif. [20], ma usando l'Eq. (1.41) per lo pseudohamiltoniano, otteniamo

$$\left. \frac{\partial x_\ell}{\partial \epsilon}(\epsilon, r) \right|_{r=R} = -\frac{2}{a(R)\chi_\ell^2(\epsilon, R)} \int_0^R \chi_\ell^2(\epsilon, r) dr, \quad (1.47)$$

$$x_\ell(\epsilon, r) \equiv \frac{\chi'_\ell(\epsilon, r)}{\chi_\ell(\epsilon, r)}, \quad (1.48)$$

dove $\chi_\ell(\epsilon, r)$ è la soluzione dell'equazione di Schrödinger radiale, Eq. (1.41), regolare in $r = 0$ e corrispondente all'energia di singola particella ϵ , in qualche stato di riferimento (di solito lo stato fondamentale atomico), $\chi'_\ell(\epsilon, r)$ la sua derivata prima rispetto a r e $x_\ell(\epsilon, r)$ la sua derivata logaritmica (vedere Appendice A). Per $R > r_c$, dato che $a(R) = 1$, l'Eq. (1.47) si riduce al ben noto risultato[20, 37]

$$\left. \frac{\partial x_\ell}{\partial \epsilon}(\epsilon, r) \right|_{r=R} = -\frac{2}{\chi_\ell^2(\epsilon, R)} \int_0^R \chi_\ell^2(\epsilon, r) dr, \quad (1.49)$$

e quindi, come mostrato originariamente da Hamann, Schlüter, e Chiang, la “conservazione della norma” [proprietà (P1) e (P2)] è di fondamentale importanza per la trasferibilità ionica. Mentre l'Eq. (1.49), per $R > r_c$, vale sia per lo pseudohamiltoniano che per gli pseudopotenziali non locali, lo stesso non è vero per le derivate in energia di ordine superiore delle derivate logaritmiche. Shirley *et al.*[37] hanno ottenuto per gli pseudopotenziali non locali

$$\begin{aligned} \left(\frac{\partial}{\partial \epsilon} \right)^n x_\ell(\epsilon, r) \Big|_{r=R} &= -\frac{1}{\chi_\ell^2(\epsilon, R)} \sum_{i=1}^{n-1} \binom{n}{i} \times \\ &\times \int_0^R \chi_\ell^2(\epsilon, r) \left(\frac{\partial}{\partial \epsilon} \right)^i x_\ell(\epsilon, r) \left(\frac{\partial}{\partial \epsilon} \right)^{n-i} x_\ell(\epsilon, r) dr, \end{aligned} \quad (1.50)$$

mentre Lastris[38] ha mostrato che per lo pseudohamiltoniano queste quantità sono uguali a

$$\left(\frac{\partial}{\partial \epsilon} \right)^n x_\ell(\epsilon, r) \Big|_{r=R} = -\frac{1}{\chi_\ell^2(\epsilon, R)} \sum_{i=1}^{n-1} \binom{n}{i} \times \\ \times \int_0^R a(r) \chi_\ell^2(\epsilon, r) \left(\frac{\partial}{\partial \epsilon} \right)^i x_\ell(\epsilon, r) \left(\frac{\partial}{\partial \epsilon} \right)^{n-i} x_\ell(\epsilon, r) dr. \quad (1.51)$$

L'Eq. (1.51) differisce dall'Eq. (1.50) per la presenza di $a(r)$ sotto il segno di integrale; questo è un modo di vedere perché, anche se pseudohamiltoniano e pseudopotenziali non locali hanno funzioni d'onda pseudo identiche a qualche energia di riferimento ϵ_ℓ , le loro derivate logaritmiche, per energie distanti dallo stato di riferimento, prima o poi diventeranno diverse².

1.2.4 Pseudohamiltoniano e pseudopotenziali non locali

In questa sezione consideriamo la relazione fra lo pseudohamiltoniano e gli pseudopotenziali non locali in ambito DFT-LDA. Se confrontiamo le equazioni di Kohn-Sham, Eq. (1.34), per lo pseudohamiltoniano con le corrispondenti equazioni per gli pseudopotenziali non locali

$$-\frac{1}{2} \frac{d^2 \chi_\ell(r)}{dr^2} + \left[\frac{\ell(\ell+1)}{2r^2} + v_\ell^{PS}(r) + v_{HXC}(r) \right] \chi_\ell(r) = \epsilon_\ell \chi_\ell(r), \quad (1.52)$$

e richiediamo che entrambe abbiano gli stessi autovalori ϵ_ℓ e autovettori $\chi_\ell(r)$ per $\ell = 0, 1, 2$, per sottrazione otteniamo la seguente equazione che lega $A(r)$, $B(r)$, $v_{ion}(r)$ con $v_\ell^{PS}(r)$:

$$A(r) \frac{1}{\chi_\ell(r)} \frac{d^2 \chi_\ell(r)}{dr^2} - [A(r) + B(r)] \frac{\ell(\ell+1)}{2r^2} + \\ + \frac{dA}{dr} \left[\frac{1}{\chi_\ell(r)} \frac{d\chi_\ell(r)}{dr} - \frac{1}{r} \right] + 2[v_\ell^{PS}(r) - v_{ion}(r)] = 0. \quad (1.53)$$

²Quando per un dato pseudoione due operatori diversi - gli ordinari pseudopotenziali non locali e lo pseudohamiltoniano che descriviamo in questo capitolo - hanno uno stesso stato di riferimento (gli stessi orbitali di singola particella occupati, gli stessi autovalori di singola particella, la stessa densità elettronica, e così lo stesso valore per il funzionale energia di Kohn-Sham) i limiti e le virtù di quei criteri di trasferibilità che si basano tutti sulle proprietà differenziali dello pseudoione calcolate in qualche stato di riferimento (per esempio la conservazione della norma o la *chemical hardness* proposta recentemente da Teter[25]) diventano più evidenti. La norma all'interno del core è la stessa e così i due ioni si comportano allo stesso modo fino al primo ordine in energia, ma, proprio perché gli orbitali di singola particella sono gli stessi, le derivate superiori in energia delle derivate logaritmiche dei loro orbitali di singola particella saranno più o meno diverse, come discusso qui; la *chemical hardness* di Teter è la stessa se lo sono gli orbitali di valenza nello stato di riferimento, ma gli ioni, in effetti, non avranno lo stesso grado di trasferibilità a causa dei termini dovuti all'auto-consistenza che sono stati trascurati nella sua definizione, come discusso altrove[39]. Sembra ragionevole concludere che, siccome i controlli "ultimi" di trasferibilità (l'effettivo confronto fra calcoli pseudo e full-core per sistemi molecolari o di stato solido) sono difficilmente accessibili e concettualmente non molto soddisfacenti, un buon criterio per la trasferibilità ionica è quello di considerare sia le derivate logaritmiche che la *chemical hardness* per due o tre diversi stati di riferimento (chimicamente significativi).

Notiamo che il ruolo dei sei potenziali non è simmetrico: dati $A(r)$, $B(r)$, $v_{ion}(r)$ non ci sono difficoltà nel calcolo dei corrispondenti $v_\ell^{PS}(r)$, mentre dati questi ultimi dobbiamo risolvere un'equazione differenziale del primo ordine per ottenere $A(r)$ [6, 21]

$$G(r)A' + F(r)A + 2V(r) = 0, \quad (1.54)$$

dove $G(r)$, $F(r)$ e $V(r)$ sono funzioni che contengono $v_\ell^{PS}(r)$, $\chi_\ell(r)$, ϵ_ℓ . Questa equazione deve essere risolta con la condizione iniziale $A(r_c) = 0$. Sfortunatamente l'esperienza ci dice che emergono due importanti difficoltà quando cerchiamo di calcolare $A(r)$ usando gli pseudopotenziali non locali standard[40]: o $G(r)$ ha degli zeri in $[0, r_c]$ e $A'(r)$ diverge in tali punti, o la proprietà (P4) non è soddisfatta. Poiché gli pseudopotenziali non locali non sono unici all'interno del core, si era inizialmente sperato che piccole distorsioni degli pseudopotenziali esistenti, tali da conservare la norma, potessero rimuovere questi due ostacoli e rendere le Eq. (1.53,1.54) una strada percorribile per la costruzione degli pseudohamiltoniani a partire dagli pseudopotenziali non locali. Ancora una volta l'esperienza ha subito mostrato che la speranza non era realistica: “in vicinanza” degli pseudopotenziali non locali esistenti non ci sono soluzioni delle Eq. (1.53,1.54) che soddisfano tutti i vincoli richiesti, e così nessuno pseudohamiltoniano accettabile[6]. Il modo in cui Bachelet, Ceperley, e Chiochetti[6] superarono il problema fu di evitare l'Eq. (1.54) eliminando i vincoli imposti allo stato con $\ell = 2$, e limitandosi a $\ell = 0, 1$. Naturalmente uno pseudoione sp è meno accurato di uno pseudoione spd , ma i metalli di transizione sono in ogni caso fuori portata, e per il resto della Tavola Periodica (elementi sp) la mancanza dell'onda d non è così seria. Questo lascia $A(r)$ indeterminata e possiamo parametrizzarla per mezzo di A_0 , r_0 e k come

$$A(r) = A_0 \exp \left[- \left(\frac{r}{r_0} \right)^k \right] \quad r_0, k > 0; \quad (1.55)$$

usando tale forma per $A(r)$ deriviamo esplicitamente $B(r)$ $v_{ion}(r)$ dalle funzioni d'onda pseudo e dagli pseudopotenziali con $\ell = 0$ e $\ell = 1$ per mezzo dell'Eq. (1.53), e alla fine proviamo a fissare i tre parametri A_0 , r_0 e k in modo tale da soddisfare la proprietà (P4). Questo metodo funziona per un certo numero di semplici elementi sp [6]. Naturalmente

l'onda d è descritta in modo poco accurato e, anche nei casi in cui solo le onde s e p sono rilevanti, uno stato d troppo legato può portare a effetti non fisici.

L'idea di concentrarsi su due funzioni d'onda χ_0 , χ_1 e lavorare su $A(r)$ è stata generalizzata e perseguita da Foulkes e Schlüter[21]. Essi non partono dagli pseudopotenziali non locali, ma parametrizzano direttamente in modo opportuno ed efficace queste tre funzioni, e ottimizzano i parametri in maniera tale da soddisfare la proprietà (P4) e le proprietà (P1) e (P2) per χ_2 e ϵ_2 ; ϵ_0 e ϵ_1 soddisfano la proprietà (P1) e vengono tenuti fissi.

1.3 Tecnica di costruzione

1.3.1 Aspetti fondamentali

Discuteremo ora in dettaglio tutte le proprietà richieste allo pseudohamiltoniano che abbiamo considerato nella Sez. 1.2.3. Possiamo raggruppare queste proprietà secondo due diversi criteri. Una prima distinzione può essere fra richieste principali e secondarie. Le prime sono quelle condizioni che devono essere soddisfatte esattamente perché lo pseudohamiltoniano abbia significato fisico come operatore di sola valenza a norma conservata. Le ultime sono condizioni addizionali e possono essere soddisfatte in modo approssimato: il grado di approssimazione misura la qualità dello pseudohamiltoniano (dal punto di vista del suo possibile uso nel maggior numero possibile di sistemi fisici differenti). Una seconda distinzione è possibile considerando i dettagli della nostra costruzione: alcune richieste saranno soddisfatte per costruzione o limitando il dominio nello spazio degli operatori hermitiani in cui effettuiamo la nostra ricerca, mentre altre saranno soddisfatte per mezzo di uno schema numerico di ottimizzazione.

Usando la stessa notazione della Sez. 1.2, possiamo scrivere le proprietà da (P1) a (P4) enunciate in Sez. 1.2.3 come

$$\epsilon_\ell = \epsilon_\ell^{FC} \quad \ell = 0, 1, 2, \quad (1.56)$$

$$\chi_\ell(r) = \chi_\ell^{FC}(r) \quad \forall r \geq r_c, \quad (1.57)$$

$$\int_0^{r_c} |\chi_\ell(r)|^2 dr = \int_0^{r_c} |\chi_\ell^{FC}(r)|^2 dr, \quad (1.58)$$

$$\chi_\ell(r) \neq 0 \quad \forall r > 0, \quad (1.59)$$

$$\begin{cases} a(r) > 0 \\ c(r) > 0. \end{cases} \quad (1.60)$$

Inoltre, per garantire che l'operatore \hat{h}^{PS} nell'Eq. (1.32) sia definito in $r = 0$, dobbiamo avere

$$\lim_{r \rightarrow 0} B(r) = 0, \quad (1.61)$$

$$\lim_{r \rightarrow 0} \left[-\frac{A'(r)}{2} + \frac{B(r)}{r} \right] = 0, \quad (1.62)$$

che sono soddisfatte se scegliamo:

$$A'(r) \stackrel{r \rightarrow 0}{\equiv} O(r), \quad (1.63)$$

$$\frac{B(r)}{r} \stackrel{r \rightarrow 0}{\equiv} O(r). \quad (1.64)$$

Di conseguenza [38, 41]

$$\begin{cases} a(r) = a_0 + a_1 r + O(r^2) \\ c(r) = c_0 + c_1 r + O(r^2) \\ a_0 = c_0 \\ a_1 = c_1 = 0. \end{cases} \quad (1.65)$$

La ragione per cui nelle Eq. (1.63,1.64) prendiamo $O(r)$ e non una potenza frazionaria di r è sta nel fatto che $r = 0$ è un punto singolare per l'Eq. (1.41); la nostra scelta garantisce che questo punto singolare è “regolare” [42] e l'equazione ammette due soluzioni linearmente indipendenti, una delle quali è finita in $r = 0$ (vedere Appendice A). Se $v(r)$ è finito in $r = 0$, è facile dimostrare che $a_0 = c_0$ e $a_1 = c_1 = 0$ se e solo se [41]

$$\chi_\ell(r) \stackrel{r \rightarrow 0}{\equiv} O(r^{\ell+1}) + O(r^{\ell+3}), \quad (1.66)$$

dove $O(r^{\ell+1})$ è anche il primo termine nell'espansione vicino a $r = 0$ delle funzioni d'onda DFT-LDA dell'atomo full-core e degli pseudopotenziali non locali; il termine successivo nell'espansione della funzione d'onda full-core è $O(r^{\ell+2})$ a causa della singolarità coulombiana presente nel potenziale nucleare, mentre per gli pseudopotenziali non locali abbiamo ancora $O(r^{\ell+3})$.

Le Eq. (1.56-1.60) e l'Eq. (1.65) sono le richieste principali. Nel nostro metodo, a causa della difficoltà di imporre i vincoli espressi dalle disuguaglianze, l'Eq. (1.60) è trattata come una condizione soddisfatta per costruzione, assieme all'Eq. (1.65) [vedere oltre l'Eq. (1.69)] e questo vale anche per l'Eq. (1.57) (l'esatta continuità con l'atomo full-core fuori dal core è pure imposta per costruzione); le Eq. (1.56,1.58,1.59) sono invece imposte per mezzo di uno schema numerico di ottimizzazione. Le richieste secondarie nascono da ragioni di carattere pratico³. Queste ci portano a considerare solo quelle funzioni radiali $a(r)$, $c(r)$ e $v(r)$ che sono limitate e continue con le loro derivate prime $\forall r \geq 0$. Per quanto riguarda le proprietà di continuità di $a(r)$, $c(r)$ e $v(r)$ con le espressioni note fuori dal raggio di core r_c , noi richiediamo esplicitamente la continuità delle funzioni e delle loro derivate prime in r_c (per costruzione, vedere la Sez. 1.3.3)

$$\begin{aligned} a(r_c) &= 1, \\ a'(r_c) &= 0, \\ c(r_c) &= 1, \\ c'(r_c) &= 0, \\ v(r_c) &= v^{FC}(r_c), \\ v'(r_c) &= v'^{FC}(r_c), \end{aligned} \tag{1.67}$$

dove $v^{FC}(r)$ è il potenziale full-core auto-consistente.

Mentre l'Eq. (1.60) fissa solo un limite inferiore su $a(r)$ e $c(r)$, varie ragioni suggeriscono di imporre dei limiti più stretti. Variazioni troppo ampie di queste funzioni nell'intervallo $[0, r_c]$ possono dare problemi numerici connessi con l'uso dello pseudohamiltoniano nei calcoli reali; valori troppo grandi o troppo piccoli di $a(r)$ e $c(r)$ in una frazione rilevante di $[0, r_c]$ possono alterare gli effetti dell'operatore energia cinetica e del potenziale centrifugo sulle funzioni d'onda DFT-LDA di singola particella in modo tale da cambiare drasticamente la

³Specialmente in previsione del successivo utilizzo dello pseudohamiltoniano.

posizione degli stati eccitati, deteriorando così la trasferibilità dello pseudohamiltoniano.

Scegliamo allora

$$\begin{aligned} 0 < a_{lo} \leq a(r) \leq a_{hi}, \\ 0 < c_{lo} \leq c(r) \leq c_{hi}. \end{aligned} \quad (1.68)$$

I limiti inferiori, che implicano l'Eq. (1.60), saranno imposti per costruzione, quelli superiori mediante ottimizzazione numerica.

Il gruppo delle Eq. (1.67,1.68) implica un raccordo “morbido” in r_c ed evita grosse variazioni di $a(r)$, $c(r)$ e $v(r)$, ma non ci dice nulla sulla loro “morbidezza”, che è una proprietà auspicabile per la maggior parte delle applicazioni pratiche. Per raggiungere questo scopo abbiamo bisogno di una procedura addizionale, trattata in dettaglio nella Sez. 1.3.3. In poche parole, costruiamo una funzione “morbida” nell'intervallo $[0, r_c]$ come combinazione lineare di certe funzioni speciali $\{R_i\}$ con coefficienti $\{\lambda_i\}$. In questo modo, tenendo presenti sia le Eq. (1.60) che l'Eq. (1.68), il nostro criterio di “morbidezza” può essere imposto per costruzione mediante le condizioni

$$\begin{cases} a(r) = a_{lo} + [\sum_i \lambda_i^a R_i(r)]^2 \\ c(r) = c_{lo} + [\sum_i \lambda_i^c R_i(r)]^2 \\ v(r) = \sum_i \lambda_i^v R_i(r). \end{cases} \quad r \in [0, r_c] \quad (1.69)$$

Le Eq. (1.65,1.67), assieme alle espressioni precedenti, introducono delle relazioni lineari fra i coefficienti $\{\lambda_i^a\}$, $\{\lambda_i^c\}$ e $\{\lambda_i^v\}$ nelle Eq. (1.69) (Sez. 1.3.3).

Ulteriori condizioni possono essere pensate e imposte⁴. Per esempio, in alcuni casi abbiamo aggiunto la richiesta di migliori derivate logaritmiche, oltre alla normale conservazione della norma⁵. Questo si può ottenere scegliendo per ogni stato di momento angolare ℓ una seconda energia ϵ'_ℓ sopra al corrispondente autovalore ϵ_ℓ ($\epsilon'_\ell > \epsilon_\ell$) e, come condizione secondaria (da soddisfare mediante ottimizzazione numerica), richiedere l'uguaglianza delle derivate logaritmiche pseudo e full-core anche per questa seconda energia ϵ'_ℓ , per esempio imponendo la seguente condizione per $R = r_c$,

$$\frac{\chi'_\ell(\epsilon'_\ell, R)}{\chi_\ell(\epsilon'_\ell, R)} = \frac{\chi'^{FC}_\ell(\epsilon'_\ell, R)}{\chi^{FC}_\ell(\epsilon'_\ell, R)}, \quad (1.70)$$

⁴Naturalmente, aggiungere un numero sempre maggiore di vincoli può rendere più difficile o addirittura impossibile la costruzione dello pseudohamiltoniano voluto.

⁵Come discuteremo nella Sez. 1.4.3, questa richiesta aggiuntiva a volte (anche se non sempre, vedere i Rif.[25, 37]) migliora la trasferibilità degli pseudoioni.

dove $\chi_\ell^{FC}(\epsilon, r)$ e $\chi_\ell(\epsilon, r)$ sono le soluzioni delle equazioni di Schrödinger radiali (1.33) e (1.41) che corrispondono all'energia ϵ , sono regolari nell'origine $r = 0$, e sono, in generale, non normalizzabili[38]. Il potenziale auto-consistente, Eq. (1.39,1.40), è quello dello stato di riferimento (Appendice A).

1.3.2 Il funzionale di costo e la sua minimizzazione

La nostra distinzione fra condizioni soddisfatte per costruzione e mediante ottimizzazione numerica non ha rilevanza fisica, ne è in alcun modo unica. Semplicemente semplifica il problema della costruzione di un opportuno pseudohamiltoniano in termini dell'algoritmo da realizzare su un calcolatore. Il problema di determinare $a(r)$, $c(r)$ e $v(r)$ è altamente non lineare dato che queste tre funzioni sono legate alle proprietà dell'atomo full-core originale per mezzo dell'equazione differenziale agli autovalori del secondo ordine, Eq. (1.41), e delle condizioni che compaiono nelle Eq. (1.56,1.58,1.59) [più, eventualmente, l'Eq. (1.70)]. Una difficoltà addizionale viene dall'Eq. (1.60), il vincolo più forte che ha impedito[6] l'utilizzo diretto degli pseudopotenziali non locali[40] per generare lo pseudohamiltoniano mediante inversione dell'Eq. (1.53). Proprio sulla base di questa esperienza abbiamo deciso che la strategia migliore è quella di soddisfare questa condizione automaticamente fin dall'inizio, e cioè includerla fra quelle soddisfatte per costruzione. Le condizioni da soddisfare per costruzione sono imposte analiticamente in un modo molto semplice descritto dettagliatamente nel seguito. Qui, invece, vogliamo illustrare come sia possibile trattare mediante una procedura di ottimizzazione numerica, quelle uguaglianze o disuguaglianze non soddisfatte per costruzione. Fortunatamente, anche per questo secondo scopo esiste una maniera naturale e abbastanza semplice: è necessario solo quantificare in maniera ragionevole il grado di approssimazione con cui si vuole soddisfare una uguaglianza o una disuguaglianza. Se l'equazione $x = y$ vale solo in modo approssimato, allora la “distanza”

$$d(x, y) = (x - y)^2 \geq 0 \quad (1.71)$$

può essere una misura pratica della deviazione dall'uguaglianza [soddisfatta esattamente se e solo se $d(x, y) = 0$] e può essere generalizzata per la disuguaglianza $x \geq y$ come

$$D(x, y) = (x - y)^2 \theta(y - x) \geq 0, \quad (1.72)$$

o, se x e y sono funzioni definite nell'intervallo $[a, b]$,

$$D(x, y) = \int_a^b [x(r) - y(r)]^2 \theta[y(r) - x(r)] dr \geq 0. \quad (1.73)$$

Se sommiamo tutte le “distanze” così definite per le Eq. (1.56, 1.58, 1.59, 1.68) e per l'Eq. (1.70), otteniamo un funzionale di $a(r)$, $c(r)$ e $v(r)$ che è non negativo, ed è uguale a zero se e solo se le equazioni sono soddisfatte esattamente, e cioè un funzionale il cui minimo assoluto, uguale a zero, corrisponde al nostro pseudohamiltoniano ideale. Anche se il funzionale non assume mai il valore zero (o non siamo in grado di trovarlo), rimane vero che, più è basso il valore che riusciamo a trovare, migliore sarà la nostra approssimazione alla soluzione esatta. In realtà, una semplice somma delle “distanze” nella definizione del funzionale, come suggerito dalle Eq. (1.71-1.73), può non essere la cosa migliore. Una somma pesata di queste distanze terrà conto del loro valore assoluto e della loro importanza (richieste principali e secondarie) in modo controllato, e dunque abbiamo preferito questa scelta. Abbiamo quindi definito il nostro funzionale di costo come segue

$$\begin{aligned} E_{cost}[a, c, v] &= E_{eig}[a, c, v] + E_{norm}[a, c, v] + \\ &+ E_{node}[a, c, v] + E_{bound}[a, c, v] + \\ &+ E_{ld}[a, c, v], \end{aligned} \quad (1.74)$$

$$E_{eig}[a, c, v] = \sum_{\ell=0}^2 \beta_{\ell} d[x_{\ell}(\epsilon_{\ell}^{FC}, r_c), x_{\ell}^{FC}(\epsilon_{\ell}^{FC}, r_c)], \quad (1.75)$$

$$E_{norm}[a, c, v] = \sum_{\ell=0}^2 \gamma_{\ell} d(N_{\ell}, N_{\ell}^{FC}), \quad (1.76)$$

$$E_{node}[a, c, v] = \begin{cases} 0 & \text{if } \chi_{\ell}(r) \neq 0 \quad \forall r > 0 \\ +\infty & \text{otherwise,} \end{cases} \quad (1.77)$$

$$E_{bound}[a, c, v] = \eta D(a_{hi}, a) + \mu D(c_{hi}, c), \quad (1.78)$$

$$E_{ld}[a, c, v] = \sum_{\ell=0}^2 \delta_{\ell} d[x_{\ell}(\epsilon'_{\ell}, r_c), x_{\ell}^{FC}(\epsilon'_{\ell}, r_c)], \quad (1.79)$$

dove

$$N_{\ell} = \int_0^{r_c} |\chi_{\ell}(\epsilon_{\ell}^{FC}, r)|^2 dr, \quad (1.80)$$

e β_{ℓ} , γ_{ℓ} , δ_{ℓ} , η e μ pesano l'importanza dei termini corrispondenti. I vari contributi al funzionale E_{cost} corrispondono, e sono riportati nello stesso ordine, alle Eq. (1.56, 1.58, 1.59, 1.68) e all'Eq. (1.70). $d[x_{\ell}(\epsilon_{\ell}^{FC}, r_c), x_{\ell}^{FC}(\epsilon_{\ell}^{FC}, r_c)]$ è equivalente a $d(\epsilon_{\ell}, \epsilon_{\ell}^{FC})$ perché $x_{\ell}(\epsilon_{\ell}^{FC}, r_c) = x_{\ell}^{FC}(\epsilon_{\ell}^{FC}, r_c) \Leftrightarrow \epsilon_{\ell} = \epsilon_{\ell}^{FC}$, ma determinare l'autovalore ϵ_{ℓ} richiede molto più lavoro che non ottenere $x_{\ell}(\epsilon_{\ell}^{FC}, r_c)$ [quest'ultimo compito richiede solo un'integrazione in più dell'Eq. (1.41) da $r = 0$ a $r = r_c$, vedere Appendice A]. L'Eq. (1.59) è stata tenuta in considerazione tramite il termine “accetto-rigetto” E_{node} : solo a , c e v tali che $\chi_{\ell}(r) \neq 0 \forall r > 0$ vengono accettati.

Poiché, attraverso le Eq. (1.69), a , c e v sono ora funzioni di $\{\lambda_i^a\}$, $\{\lambda_i^c\}$, e $\{\lambda_i^v\}$ (la prossima sezione contiene tutti i dettagli sul nostro insieme di funzioni di base), il funzionale E_{cost} diventa anch'esso una funzione di $\{\lambda_i^a\}$, $\{\lambda_i^c\}$, e $\{\lambda_i^v\}$, e come tale è usato in pratica: si cerca il minimo di E_{cost} nello spazio multidimensionale di $\{\lambda_i^a\}$, $\{\lambda_i^c\}$ e $\{\lambda_i^v\}$. Ne ragioni teoriche ne l'esperienza suggeriscono che questa funzione di costo sia priva di minimi locali. Per una generica funzione di molte variabili reali, a meno di non conoscere con precisione sufficiente la posizione del minimo assoluto, dobbiamo affrontare il problema di evitare i minimi locali durante la ricerca del minimo globale. Esistono molti algoritmi deterministici che, dato un certo punto di partenza, sono in grado di localizzare rapidamente il minimo locale più vicino della funzione in analisi[43], ma il nostro problema è la mancanza di un buon punto di partenza. Se dovessimo basarci su questi algoritmi deterministici, dovremmo provare moltissimi (in linea di principio infiniti) diversi punti di partenza scelti a caso, ottenere mediante la ricerca deterministica molti (tutti) minimi locali, e alla fine selezionare

il migliore (in linea di principio quello assoluto). Questa strada è però impraticabile se il numero di variabili è elevato. Una strategia molto più efficiente è l'*annealing* simulato con campionamento Metropolis Monte Carlo[44, 45, 46], in cui un cammino casuale nel dominio di ricerca è affiancato da un criterio di campionamento che guida il cammino solo verso le regioni più interessanti e in modo tale da evitare l'intrappolamento nei vari minimi locali. L'idea di usare l'*annealing* simulato per generare gli pseudohamiltoniani è dovuta a Mitáš *et al.*[27, 47], che ha anche dimostrato la percorribilità di questa strada. Siano $a_0(r)$, $c_0(r)$ e $v_0(r)$ le nostre funzioni a un certo punto; allora il passo successivo del cammino casuale genera tre nuove funzioni $a(r)$, $c(r)$ e $v(r)$; queste funzioni sono accettate (selezionate) con la probabilità

$$P = \begin{cases} 1 & \text{if } E \leq E_0 \\ \exp\left(-\frac{E-E_0}{T}\right) & \text{if } E > E_0 \end{cases} \quad (1.81)$$

data dal criterio di campionamento alla Boltzmann. Qui abbiamo posto $E = E_{cost}[a, c, v]$, $E_0 = E_{cost}[a_0, c_0, v_0]$, e il ruolo della temperatura finita $T > 0$ è quello di garantire una probabilità non nulla che permetta alla funzione di costo di crescere fra un passo e l'altro. In altre parole, durante il cammino Monte Carlo, alla funzione E_{cost} è permesso (a volte) crescere, in modo tale da evitare l'intrappolamento nei minimi locali⁶. Se la mossa è accettata, il passo successivo avrà il nuovo insieme $a(r)$, $c(r)$ e $v(r)$ come punto di partenza. Altrimenti il passo è ignorato, le funzioni $a(r)$, $c(r)$ e $v(r)$ scartate, e il nuovo passo partirà ancora da $a_0(r)$, $c_0(r)$ e $v_0(r)$. La scelta più semplice per la distribuzione che regola il cammino casuale è una probabilità uniforme nell'intervallo $[-\sqrt{3}Q, \sqrt{3}Q]$ per ognuna delle coordinate⁷. Questa scelta da origine a una distribuzione isotropa in cui la lunghezza del passo quadratico medio è $Q\sqrt{n}$ (n è la dimensione del vettore posizione) e una matrice covarianza diagonale $S_{ij} = Q^2\delta_{ij}$ [48]. Anche se questa scelta è attraente per la sua semplicità, non abbiamo ragione di ritenere che la nostra funzione di costo sia isotropa, e potremmo

⁶In pratica la migliore strategia per “congelare” gradualmente il sistema nel suo minimo globale è iniziare il cammino casuale con una temperatura relativamente alta e quindi ridurla con il procedere del cammino.

⁷ Q è scelto in modo tale da avere la massima efficienza o equivalentemente il massimo guadagno di informazione, e cioè quando approssimativamente metà dei passi generati dal cammino casuale vengono accettati.

perdere molto tempo nell'esplorazione di direzioni di ricerca infruttuose. Per superare questo problema abbiamo bisogno di un cammino casuale la cui matrice covarianza si adatti alla topografia locale della funzione di costo, e abbiamo adottato la strategia proposta da Vanderbilt e Louie[48, 49]. Il minimo globale è raggiunto con probabilità unitaria solo per un cammino infinito[46] ma, se fissiamo una soglia (sopra il valore ideale che è zero), che corrisponda a uno pseudohamiltoniano accettabile, raggiungiamo spesso una regione dove la funzione di costo ha un valore inferiore all'accuratezza richiesta in un tempo (o lunghezza del cammino) ragionevole. In pratica la procedura di annealing simulato viene interrotta non appena il valore della funzione di costo scende al di sotto della soglia; è risultato conveniente usare il metodo dei semplici non lineare[43] per migliorare il minimo trovato come punto finale dell'annealing simulato e ottenere così la soluzione finale ottimale.

1.3.3 Rappresentazione di $a(r)$, $c(r)$ e $v(r)$

Come abbiamo anticipato nella sezione precedente, un trattamento numerico è possibile se possiamo rappresentare in forma conveniente e finita le tre funzioni continue $a(r)$, $c(r)$ e $v(r)$ e così trasformare il funzionale di costo in una funzione a un numero finito di variabili. Un'ovvia possibilità è quella di espandere le tre funzioni a , c , v in termini di una qualche opportuna base finita di funzioni, e quindi lavorare con i coefficienti $\{\lambda_i^a\}$, $\{\lambda_i^c\}$ e $\{\lambda_i^v\}$ di tale espansione. Esistono molte basi analitiche di funzioni; fra queste abbiamo scelto una famiglia non ortogonale $\{R_i\}$ la cui virtù principale, dal nostro punto di vista, è il criterio di "morbidezza" che soddisfa[50, 51]. Sia \mathcal{W} lo spazio delle funzioni analitiche di un variabile reale nell'intervallo $(0, r_c)$ e definiamo la seminorma "morbida" $\|f\|$ della funzione $f \in \mathcal{W}$

$$\|f\|^2 = \sum_{n=1}^{+\infty} D^{2n} \int_0^{r_c} \left(\frac{\partial^n f}{\partial x^n} \right)^2 dx, \quad (1.82)$$

dove $D > 0$ è detta tensione. È possibile dimostrare[50, 51] che la minimizzazione di $\|f\|$ nel dominio $\{f \in \mathcal{W} : \|f\| < +\infty\}$, con il vincolo $f(x_i) = f_i$, $f'(x_1) = f'_1$ e $f'(x_N) = f'_N$, dove $\{x_i\}_{i=1}^N$ è una partizione (uniforme) N -dimensionale di $[0, r_c]$, dà come risultato

$$f(x) = \lambda_0 + \sum_{i=1}^N \lambda_i R(x, x_i) + \lambda_{N+1} \frac{\partial}{\partial y} R(x, y) \Big|_{y=x_N} +$$

$$+ \lambda_{N+2} \frac{\partial}{\partial y} R(x, y) \Big|_{y=x_1}, \quad (1.83)$$

dove

$$R(x, y) = -\frac{|x - y|}{D^2} \coth \left(\frac{\pi |x - y|}{D} \right). \quad (1.84)$$

I coefficienti $\{\lambda_i\}_{i=0}^{N+2}$ sono le soluzioni del sistema lineare $(N+3) \times (N+3)$

$$\begin{aligned} f_k = \lambda_0 + \sum_{i=1}^N \lambda_i R(x_k, x_i) + \lambda_{N+1} \frac{\partial}{\partial y} R(x_k, y) \Big|_{y=x_N} + \\ + \lambda_{N+2} \frac{\partial}{\partial y} R(x_k, y) \Big|_{y=x_1} \quad k = 1, \dots, N \end{aligned} \quad (1.85)$$

$$\begin{aligned} f'_j = \sum_{i=1}^N \lambda_i \frac{\partial}{\partial x} R(x, y) \Big|_{\substack{x=x_j \\ y=x_i}} + \lambda_{N+1} \frac{\partial}{\partial x} \frac{\partial}{\partial y} R(x, y) \Big|_{\substack{x=x_j \\ y=x_N}} + \\ + \lambda_{N+2} \frac{\partial}{\partial x} \frac{\partial}{\partial y} R(x, y) \Big|_{\substack{x=x_j \\ y=x_1}} \quad j = 1, N \end{aligned} \quad (1.86)$$

$$\sum_{i=1}^N \lambda_i = 0 \quad (1.87)$$

e $f(x)$ è detta *spline* regolarizzato con tensione (RST)[50, 51]. Notiamo che la tensione D ci permette di selezionare quali derivate pesano di più nella definizione della seminorma, Eq. (1.82), e così permette di passare da un'interpolazione lineare a tratti a una curva sempre più smussata passante per i vari punti.

Per soddisfare i vincoli imposti dall'Eq. (1.68) prendiamo il quadrato del membro di destra nell'Eq. (1.83) e sommiamo una costante positiva; alla fine otteniamo l'Eq. (1.69) riportata nella Sez. 1.3.1, dove abbiamo posto:

$$\left\{ \begin{array}{l} R_0(x) = 1 \\ R_i(x) = R(x, x_i) \quad i = 1, \dots, N \\ R_{N+1}(x) = \frac{\partial}{\partial y} R(x, y) \Big|_{y=x_N} \\ R_{N+2}(x) = \frac{\partial}{\partial y} R(x, y) \Big|_{y=x_1} \end{array} \right. \quad (1.88)$$

Nel nostro caso una scelta ragionevole è risultata essere $N \approx 20$. Usando la rappresentazione in termini di RST, le Eq. (1.65,1.67) possono essere scritte in una forma più utile. Sotto ripetiamo a sinistra le equazioni originali e a destra le loro conseguenze sugli RST che compaiono nelle Eq. (1.69,1.85-1.87) e le loro derivate prime

$$\begin{aligned}
 \begin{cases} a(r_c) = 1 \\ a'(0) = 0 \\ a'(r_c) = 0 \end{cases} & \Rightarrow \begin{cases} f_N = \sqrt{1 - a_{lo}} \\ f'_1 = 0 \\ f'_N = 0 \end{cases} \\
 \begin{cases} c(0) = a(0) \\ c(r_c) = 1 \\ c'(0) = 0 \\ c'(r_c) = 0 \end{cases} & \Rightarrow \begin{cases} f_1 = \sqrt{a(0) - c_{lo}} \\ f_N = \sqrt{1 - c_{lo}} \\ f'_1 = 0 \\ f'_N = 0 \end{cases} \quad (1.89) \\
 \begin{cases} v(r_c) = v^{FC}(r_c) \\ v'(r_c) = v'^{FC}(r_c) \end{cases} & \Rightarrow \begin{cases} f_N = v^{FC}(r_c) \\ f'_N = v'^{FC}(r_c) \end{cases}
 \end{aligned}$$

Dalle condizioni appena scritte seguono delle relazioni lineari fra i coefficienti $\{\lambda_i^a\}$, $\{\lambda_i^c\}$ e $\{\lambda_i^v\}$. Un'altra relazione viene dall'Eq. (1.87) e abbiamo in tutto 4 vincoli lineari fra i $\{\lambda_i^a\}_{i=0}^{N+2}$, 5 fra i $\{\lambda_i^c\}_{i=0}^{N+2}$ e 3 fra i $\{\lambda_i^v\}_{i=0}^{N+1}$. Noi prendiamo $\{\lambda_i^a\}_{i=1}^{N-1}$, $\{\lambda_i^c\}_{i=1}^{N-2}$ e $\{\lambda_i^v\}_{i=1}^{N-1}$ come coefficienti indipendenti, e determiniamo i restanti dalle Eq. (1.85-1.87,1.89).

Una volta che le tre funzioni sono state determinate, usiamo le Eq. (1.35), (1.36), (1.39) e (1.41) per calcolare il potenziale ionico $v_{ion}(r)$. Le Eq. (1.41) vengono integrate usando una griglia numerica finita per trovare la densità di carica auto-consistente $n(r)$ (vedere Appendice A). Il potenziale $v_{HXC}(r)$ ottenuto da questa densità è interpolato con uno RST in $[0, r_c]$ per avere $v_{ion}(r)$ nella stessa forma. Fuori dal core $A(r) = B(r) = 0$ e, dall'Eq. (1.46),

$$v_{ion}(r) = -\frac{Z}{r} + v_{HXC}^{FC}(r) - v_{HXC}(r). \quad (1.90)$$

Per grandi valori di r , $v_{ion}(r)$ tende a $-Z_v/r$ ma subito fuori dal core può differire leggermente da questo andamento asintotico, e dunque fuori dal core abbiamo trovato conveniente rappresentarlo mediante la sovrapposizione di alcune funzioni d'errore moltiplicate per la coda coulombiana

$$v_{ion}^{FIT}(r) = -\frac{Z_v}{r} \sum_i c_i \text{erf}(\sqrt{\alpha_i} r) \quad r \geq r_c. \quad (1.91)$$

In particolare utilizziamo 3 funzioni d'errore, soggette alle condizioni

$$\lim_{r \rightarrow \infty} v_{ion}^{FIT}(r) = -\frac{Z_v}{r}, \quad (1.92)$$

$$v_{ion}^{FIT}(r_c) = v_{ion}(r_c), \quad (1.93)$$

$$\frac{dv_{ion}^{FIT}}{dr}(r_c) = \frac{dv_{ion}}{dr}(r_c), \quad (1.94)$$

che possono essere riscritte come

$$\sum_{i=1}^3 c_i = 1, \quad (1.95)$$

$$-\frac{Z_v}{r_c} \sum_{i=1}^3 c_i \operatorname{erf}(\sqrt{\alpha_i} r_c) = v_{ion}(r_c), \quad (1.96)$$

$$-\frac{2Z_v}{r_c} \sum_{i=1}^3 c_i \sqrt{\frac{\alpha_i}{\pi}} \exp(-\alpha_i r_c^2) = \frac{dv_{ion}}{dr}(r_c) + \frac{v_{ion}(r_c)}{r_c}. \quad (1.97)$$

Dato un qualunque insieme di costanti di decadimento $\{\alpha_i\}$, i tre coefficienti lineari $\{c_i\}$ vengono determinati dalle tre equazioni lineari (1.95-1.97). I parametri non lineari $\{\alpha_i\}$, invece, vengono scelti in modo tale da interpolare il meglio possibile il potenziale ionico fuori dal core; dato che, per grandi valori di r , i due andamenti sono gli stessi per costruzione, questo equivale alla richiesta che l'interpolazione sia la migliore possibile immediatamente fuori dal core. Le tre costanti di decadimento $\{\alpha_i\}$ devono dunque essere tali da minimizzare la funzione di costo:

$$\max_r \{|v_{ion}^{FIT}(r) - v_{ion}(r)| : r \in [r_c, +\infty]\} \quad (1.98)$$

e l'algoritmo dei semplici[43] è stato usato a tal fine.

1.4 Risultati e discussione

1.4.1 Considerazioni generali sull'annealing simulato

Il punto di partenza della nostra procedura di annealing simulato è di solito lo pseudohamiltoniano del Rif. [6], che riproduce esattamente solo gli stati di momento angolare s e p . L'intera procedura è portata avanti finchè non viene raggiunto un minimo della funzione di costo, tale che le richieste principali siano soddisfatte entro l'accuratezza richiesta: valori tipici per gli errori relativi sugli autovalori DFT-LDA sono dell'ordine di 10^{-4} ; sulle

norme di 10^{-3} [8]. Poiché le richieste secondarie hanno pesi più piccoli nella funzione di costo (Sez. 1.3.2), queste sono soddisfatte solo in modo approssimato in corrispondenza del minimo trovato. A seconda del grado di approssimazione decidiamo se fermarci davvero e prendere le funzioni trovate $a(r)$, $c(r)$ e $v(r)$ come soluzione finale o se, invece, è il caso di ripartire. In effetti, oltre a considerare la funzione di costo e trovare un minimo sotto la soglia stabilita, come suggerito al termine della Sez. 1.3.2, alla fine consideriamo anche i singoli termini della funzione di costo e operiamo un aggiustamento fine che corrisponde a imporre delle soglie di tolleranza ai singoli contributi.

La procedura di annealing simulato (che inevitabilmente comporta un certo grado di arbitrarietà), è costituita da molti cicli in cui si procede per tentativi. Questo è dovuto alla scarsa conoscenza e alla grande complessità della topografia della funzione di costo, e al fatto che l'algoritmo converge alla soluzione ottimale con probabilità unitaria solo in linea di principio, dato che non possiamo eseguire cammini casuali infinitamente lunghi. Una ricetta generale sul modo più efficiente possibile di portare avanti l'annealing simulato è così impossibile. Ci sono comunque due linee guida: un cammino troppo corto implica un'alta probabilità di mancare le regioni importanti dello spazio dei parametri; un'abbassamento troppo veloce della temperatura T implica un'alta probabilità di rimanere "congelati" in un minimo locale che non è accettabile. Naturalmente "troppo corto" e "troppo veloce" dipendono dal problema e possiamo solo ricorrere alla strategia per tentativi. Nella nostra esperienza (con la nostra particolare funzione di costo a ~ 60 parametri), se il punto di partenza del cammino casuale è abbastanza distante da un minimo locale, o in altre parole se la temperatura T non è troppo bassa, anche due procedure di annealing simulato che partono dallo stesso punto nello spazio dei parametri, terminano con due minimi diversi, se le due procedure utilizzano strategie lievemente diverse. Questo è naturalmente dovuto alla complessa struttura della funzione di costo, e sarebbe scoraggiante se lo scopo dell'annealing simulato fosse quello di individuare l'esatta posizione del minimo assoluto della funzione di costo. Fortunatamente, comunque, una volta che l'annealing simulato trova un minimo locale che abbia l'accuratezza richiesta, allora tutti i minimi locali (in linea di principio

diversi), che raggiungiamo con procedure indipendenti ma che partono dallo stesso punto iniziale, sono di accuratezza comparabile. Dunque, per il nostro scopo, la strategia seguita è perfettamente adeguata.

1.4.2 Alcuni esempi

Abbiamo ottenuto un certo numero di pseudohamiltoniani per mezzo del potente metodo dell'annealing simulato. In particolare, per i metalli alcalini da Li fino a Cs, partendo dallo pseudohamiltoniano del Rif. [6], è possibile ottenere uno pseudohamiltoniani più trasferibili, come vedremo nel dettaglio per Na. Un'altra regione della Tavola Periodica che è stata studiata è la seconda riga per la quale, oltre a Na, abbiamo costruito pseudohamiltoniani per Al, Si, P e Cl. Fra questi il caso di Si verrà considerato in dettaglio nel seguito. Anche la terza riga è stata affrontata con risultati positivi per Ge e As, mentre abbiamo delle difficoltà per gli elementi della prima riga: presenteremo dei risultati per C tutt'altro che soddisfacenti, ma che mostrano come sia possibile fare qualcosa e anche chiarire le ragioni per cui gli pseudohamiltoniani per la prima riga siano intrinsecamente meno trasferibili. Come abbiamo visto, ragioni teoriche ci impediscono di descrivere i metalli di transizione per mezzo di pseudohamiltoniani standard [e cioè che soddisfano le proprietà da (P1) a (P4)] e non li abbiamo considerati. Altre regioni, quali la seconda colonna della Tavola Periodica, non dovrebbero presentare particolari problemi e verranno considerate in futuro.

I coefficienti numerici tramite i quali sono espressi gli pseudohamiltoniani generati per Na e Si, Eq. (1.69) e Sez. 1.3.3, sono riportati nelle Tab. 1.1, 1.2.

Per ogni pseudohamiltoniano ottenuta dalla procedura di annealing simulato, abbiamo eseguito un calcolo DFT-LDA completo. Ciò è stato fatto per controllare gli errori numerici sugli autovalori della configurazione di riferimento rispetto ai corrispondenti atomi full-core, dato che questi sono solo stimati e mai calcolati esattamente (vedere Appendice A) durante l'annealing simulato. La configurazione di riferimento scelta per Si è $s^{0.8}p^{2.4}d^{0.2}$ ($\ell = 0, 1, 2$) con raggio di core $r_c = 1.91$ au, per Na $s^{0.6}p^{0.1}$ ($\ell = 0, 1$) con $r_c = 2.40$ au e per C è lo stato fondamentale s^2p^2 ($\ell = 0, 1$) con $r_c = 1.31$ au. Le Tab. 1.3, 1.4 e 1.5 mostrano i risultati per

λ_i^a	λ_i^c	λ_i^v
5.76518989e-01	1.70979193e+00	1.20130048e+00
6.65210233e-02	-1.72351870e-01	2.84590567e-02
5.77566455e-02	6.86826383e-01	1.03877144e-01
-2.09352277e-02	-1.53759380e+00	4.28380526e-02
-3.20963738e-03	2.68266646e+00	1.54409915e-02
-5.86757432e-02	-3.84102636e+00	-6.63699217e-02
-1.05513453e-01	5.91776382e+00	3.10125715e-01
-1.32814774e-01	-8.87619573e+00	-6.67581730e-01
-1.32583064e-01	1.28114576e+01	9.97168797e-01
2.83884637e-04	-1.78265946e+01	-8.47770603e-01
1.98865341e-02	2.33384671e+01	8.20777297e-01
-5.41527681e-03	-2.68059186e+01	-7.23437142e-01
2.30146630e-01	2.56523750e+01	6.60790352e-01
6.47655429e-03	-2.49046589e+01	-6.53277213e-01
1.92650468e-01	2.70519437e+01	6.80532779e-01
-3.57401854e-02	-3.35346855e+01	-1.21794425e+00
6.18658109e-03	4.43420136e+01	1.53108557e+00
1.14996058e-02	-6.02924420e+01	-2.46882437e+00
1.64896499e-02	8.13617535e+01	3.26615539e+00
5.69463531e-02	-1.05242990e+02	-4.53372782e+00
2.05776679e-02	1.42281347e+02	6.15069776e+00
-1.90534236e-01	-8.30921569e+01	-3.42901586e+00
3.94510582e-02	6.14666699e+00	2.46550693e-01
2.14548751e-02	-1.11733006e-03	
$Z = 11$	$Z_v = 1$	$N = 21$
$r_c = 2.40427915e+00$	$D = 5.00000000e-01$	
$a_{lo} = 5.00000000e-02$	$c_{lo} = 1.00000000e-02$	
$c_1 = -5.62579846e-01$	$c_2 = -7.60727821e+00$	$c_3 = 9.16985806e+00$
$\alpha_1 = 3.00651160e-01$	$\alpha_2 = 6.34776801e-01$	$\alpha_3 = 9.84380454e-01$

Tavola 1.1: Coefficienti numerici dello pseudohamiltoniano per l'atomo di Na (vedere Sez. 1.3.3).

λ_i^a	λ_i^c	λ_i^v
1.57773662e+00	7.17813275e-01	-1.22586351e+00
-8.85236760e-01	2.16485127e+02	4.80429388e-02
1.80143777e+00	-4.97923120e+02	-7.73525376e-01
-1.72675285e+00	6.83268691e+02	3.82821268e+00
1.75841796e+00	-9.63053012e+02	-9.73799267e+00
-2.80132889e+00	1.32586946e+03	1.83722202e+01
3.69441735e+00	-1.77981302e+03	-2.74939378e+01
-5.16169983e+00	2.32586387e+03	4.00928597e+01
7.34859550e+00	-2.93498646e+03	-5.54510496e+01
-8.97216383e+00	3.53117054e+03	7.31762428e+01
1.17584506e+01	-3.98046094e+03	-9.75429253e+01
-1.54935995e+01	4.16770026e+03	1.27989123e+02
2.03687067e+01	-4.11444787e+03	-1.65142887e+02
-2.67627715e+01	3.91064877e+03	2.12550896e+02
3.36901960e+01	-3.57796012e+03	-2.70134840e+02
-4.14709224e+01	3.10003116e+03	3.35502611e+02
5.15386069e+01	-2.52395610e+03	-4.06201324e+02
-6.16352611e+01	1.93796994e+03	4.65263656e+02
7.21171341e+01	-1.40818063e+03	-4.94131168e+02
-8.07012089e+01	9.67736199e+02	4.81713922e+02
8.62369408e+01	-6.57305731e+02	-4.46214160e+02
-4.47019581e+01	2.71342986e+02	2.14286023e+02
2.33283412e+00	-1.15903468e+01	-1.01743412e+01
-4.45098506e-02	9.75030914e+00	
$Z = 14$	$Z_v = 4$	$N = 21$
$r_c = 1.90801447e+00$	$D = 5.00000000e-01$	
$a_{lo} = 1.00000000e-01$	$c_{lo} = 1.00000000e-03$	
$c_1 = 7.01006204e+00$	$c_2 = -5.85550349e+00$	$c_3 = -1.54558553e-01$
$\alpha_1 = 2.37847912e+00$	$\alpha_2 = 2.15978820e+00$	$\alpha_3 = 1.04377440e+00$

Tavola 1.2: Coefficienti numerici dello pseudohamiltoniano per l'atomo di Si (vedere Sez. 1.3.3).

	configurazione	ϵ_s	ϵ_p	ϵ_d	E
PH	$s^{0.8}p^{2.4}d^{0.2}$	-0.62778	-0.35861	-0.07714	-3.24934
	s^2p^2	-0.39899	-0.15268	-	-3.76002
Full-core	$s^{0.8}p^{2.4}d^{0.2}$	-0.62788	-0.35866	-0.07714	-288.25400
	s^2p^2	-0.39989	-0.15322	-	-288.76510
PP	$s^{0.8}p^{2.4}d^{0.2}$	-0.62588	-0.35752	-0.07816	-3.24021
	s^2p^2	-0.39990	-0.15327	-	-3.75042

Tavola 1.3: Energie atomiche per Si nella configurazione di riferimento $s^{0.8}p^{2.4}d^{0.2}$ e nello stato fondamentale s^2p^2 . Gli autovalori ϵ_ℓ e le energie totali E sono mostrati per lo pseudohamiltoniano (PH), l'atomo full-core (Full-core) e gli pseudopotenziali non locali (PP) del Rif. [40] (vedere testo).

	configurazione	ϵ_s	ϵ_p	ϵ_d	E
PH	$s^{0.6}p^{0.1}d^0$	-0.15928	-0.07313	-0.01218	-0.13843
	s^1p^0	-0.10466	-0.02824	-	-0.18617
Full-core	$s^{0.6}p^{0.1}d^0$	-0.15931	-0.07314	-0.01271	-161.56832
	s^1p^0	-0.10371	-0.02847	-	-161.61580
PP	$s^{0.6}p^{0.1}d^0$	-0.15724	-0.07738	-0.01259	-0.13767
	s^1p^0	-0.10357	-0.03205	-	-0.18437

Tavola 1.4: Energie atomiche per Na nella configurazione di riferimento $s^{0.6}p^{0.1}$ e nello stato fondamentale s^1 . Gli autovalori ϵ_ℓ e le energie totali E sono mostrati per lo pseudohamiltoniano (PH), l'atomo full-core (Full-core) e gli pseudopotenziali non locali (PP) del Rif. [40] (vedere testo).

	configurazione	ϵ_s	ϵ_p	E
PH	s^2p^2	-0.50128	-0.19931	-5.34973
Full-core	s^2p^2	-0.50120	-0.19925	-37.42832
PP	s^2p^2	-0.50139	-0.19857	-5.34505

Tavola 1.5: Energie atomiche per C nello stato di riferimento s^2p^2 che è anche lo stato fondamentale. Gli autovalori ϵ_ℓ e le energie totali E sono mostrati per lo pseudohamiltoniano (PH), l'atomo full-core (Full-core) e gli pseudopotenziali non locali (PP) del Rif. [40] (vedere testo).

Si, Na e C confrontati con i calcoli full-core e con gli pseudopotenziali non locali standard[40]; abbiamo considerato anche altre configurazioni interessanti, quali lo stato fondamentale.

Le funzioni $A(r)$, $B(r)$ e $v_{ion}(r)$, assieme alle corrispondenti funzioni d'onda radiali $\chi_\ell(r)$

per la configurazione di riferimento, sono mostrate in Fig. 1.1 per Si e Fig. 1.2 per Na. Le funzioni d'onda full-core sono mostrate per confronto: notiamo che ne l'errore sull'autovalore ne quello sulla norma possono essere risolti; in altre parole, le code delle funzioni sono perfettamente sovrapposte. Questo risultato, ottenuto tramite ottimizzazione numerica (e non mediante inversione numerica come nel metodo di Hamann, Schlüter, e Chiang del Rif. [20] e altri metodi simili), è un successo rilevante della nuova procedura presentata qui. Anche nel caso di Na consideriamo autovalore e autovettore d (con occupazione zero nella configurazione di riferimento $s^{0.6}p^{0.1}d^0$), anche se per questo elemento non abbiamo incluso l'onda d nella costruzione dello pseudohamiltoniano; per l'onda d noi troviamo (Tab. 1.4) $\epsilon_d \geq \epsilon_d^{FC}$, come deve essere (vedere la fine della Sez. 1.2.4); l'autovalore ϵ_d è molto vicino all'autovalore full-core (solo qualche punto percentuale diverso da ϵ_d^{FC}), ma vediamo dalla Fig. 1.2(d) che la conservazione della norma di $\chi_d(r)$ non è, invece, così buona: le code delle funzioni d'onda non si sovrappongono. Questo dimostra che, a meno di non includere esplicitamente tutte e tre le onde spd nella procedura di ottimizzazione, lo pseudohamiltoniano risultante è inevitabilmente meno accurato.

1.4.3 Il problema della trasferibilità

Per mettere a confronto lo pseudohamiltoniano con lo ione vero full-core, abbiamo effettuato un certo numero di calcoli atomici. Incominciamo con l'esaminare le derivate logaritmiche in funzione dell'energia, mostrate in Fig. 1.4 per C, Fig. 1.5 per Na, e Fig. 1.6 per Si. Le derivate logaritmiche sono calcolate al raggio covalente; nelle figure mostriamo $R'_\ell(\epsilon, r)/R_\ell(\epsilon, r)$ invece $\text{dix}_\ell(\epsilon, r)$ definito nell'Eq. (1.48); essi sono legati dalla relazione

$$R_\ell(\epsilon, r) = \frac{\chi_\ell(\epsilon, r)}{r}, \quad (1.99)$$

$$\frac{R'_\ell(\epsilon, r)}{R_\ell(\epsilon, r)} = rx_\ell(\epsilon, r) - 1. \quad (1.100)$$

Il caso del carbonio illustra una differenza importante fra pseudohamiltoniani e pseudopotenziali non locali. La Fig. 1.4 mostra le derivate logaritmiche per l'onda s [(a) e (c), riquadri di sinistra] e per l'onda p [(b) e (d), riquadri di destra] in funzione dell'energia

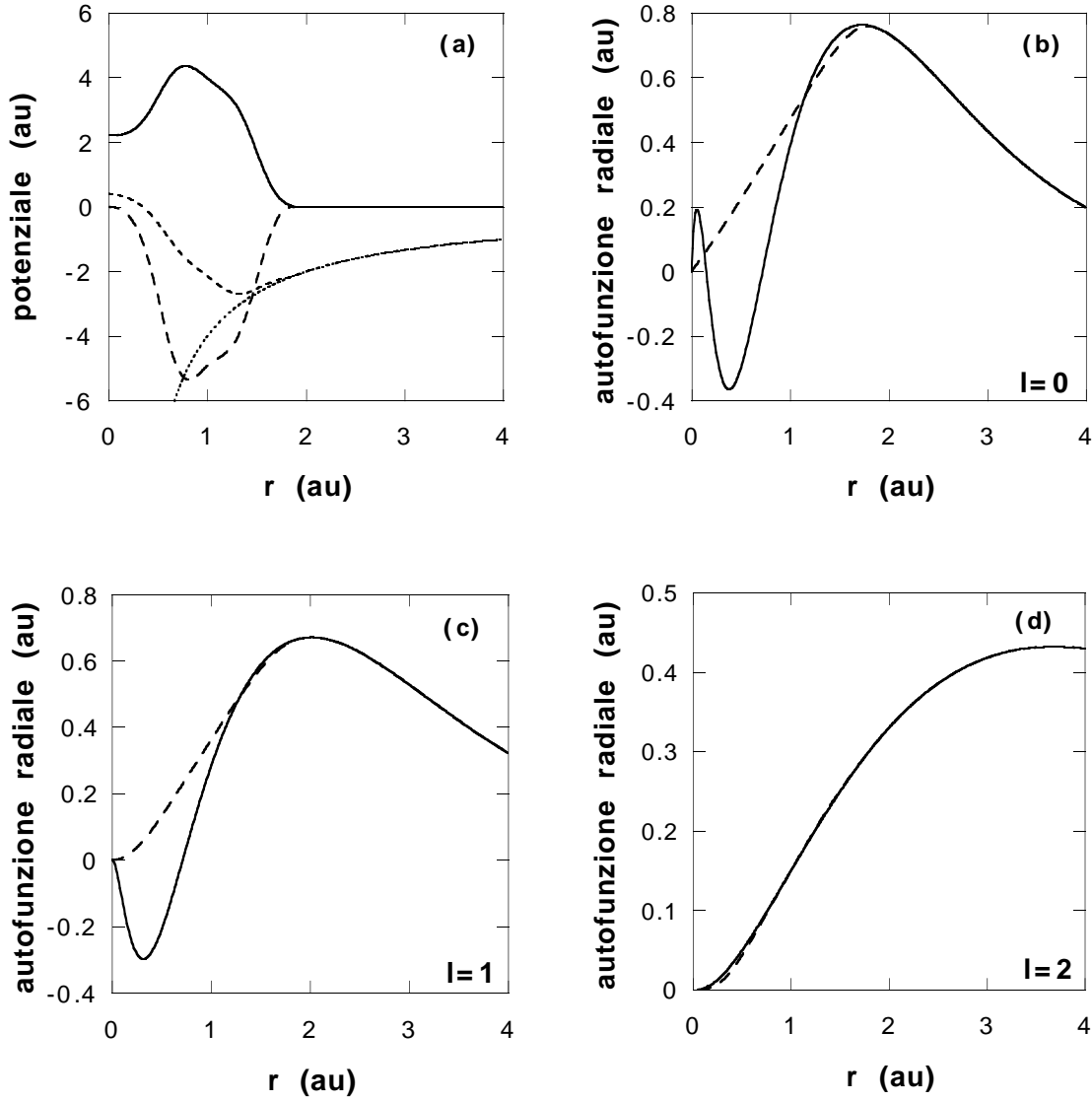


Figura 1.1: Pseudohamiltoniano per Si: la configurazione di riferimento è $s^{0.8}p^{2.4}d^{0.2}$ e il raggio di core $r_c = 1.91$ au. (a) Funzioni $A(r)$ (linea continua), $B(r)$ (tratteggio lungo), $v_{ion}(r)$ (tratteggio corto) e potenziale coulombiano $-Z_v/r$ (linea punteggiata). (b) funzione d'onda s (c) funzione d'onda p e (d) funzione d'onda d per l'atomo full-core (linea continua) e lo pseudohamiltoniano (linea tratteggiata) nella configurazione di riferimento (vedere testo).

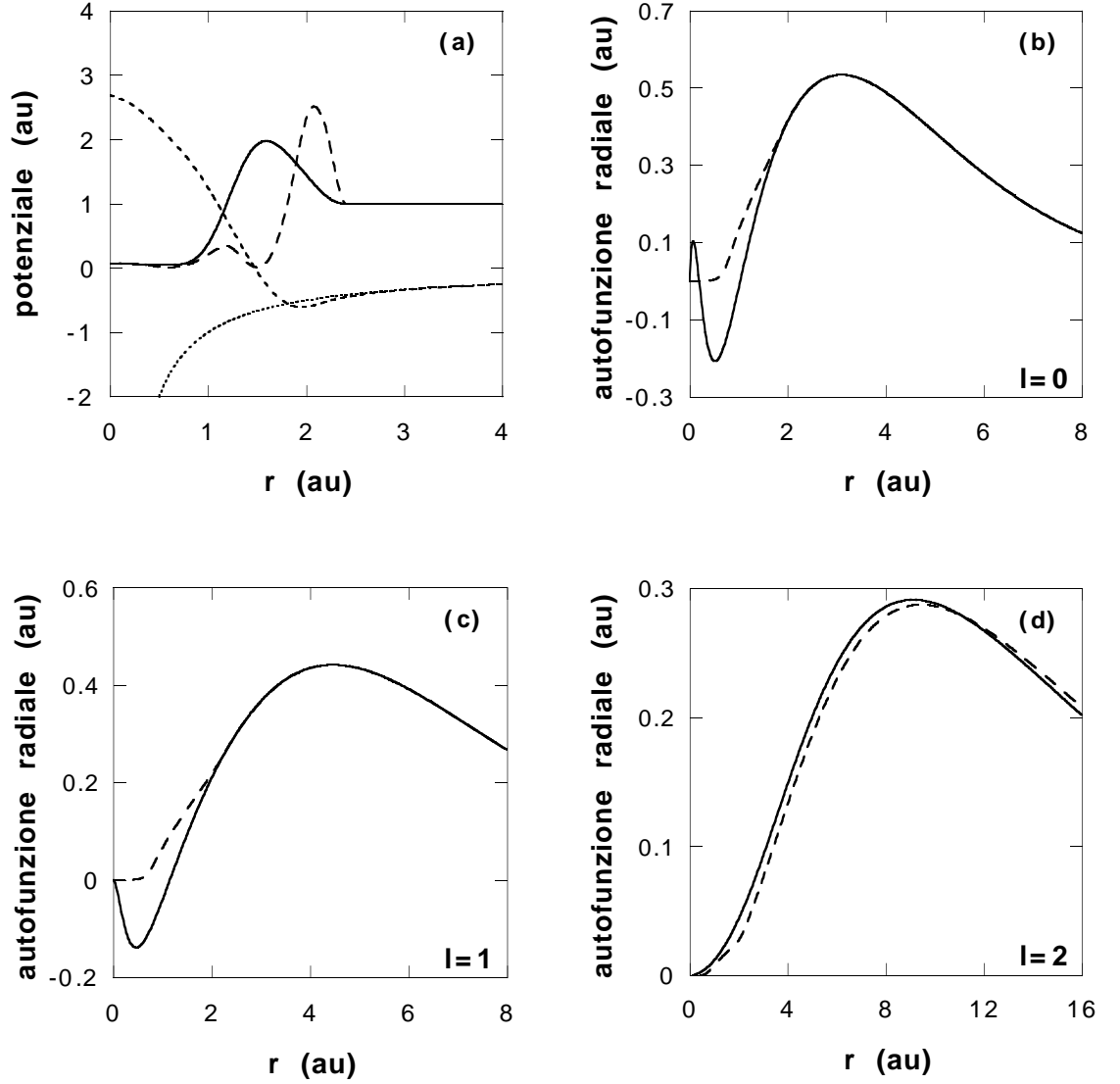


Figura 1.2: Pseudohamiltoniano per Na: la configurazione di riferimento è $s^{0.6}p^{0.1}$ e il raggio di core $r_c = 2.40$ au. (a) Funzioni $A(r)$ (linea continua), $B(r)$ (tratteggio lungo), $v_{ion}(r)$ (tratteggio corto) e potenziale coulombiano $-Z_v/r$ (linea punteggiata). (b) funzione d'onda s (c) funzione d'onda p e (d) funzione d'onda d per l'atomo full-core (linea continua) e lo pseudohamiltoniano (linea tratteggiata) nella configurazione $s^{0.6}p^{0.1}d^0$ (vedere testo).

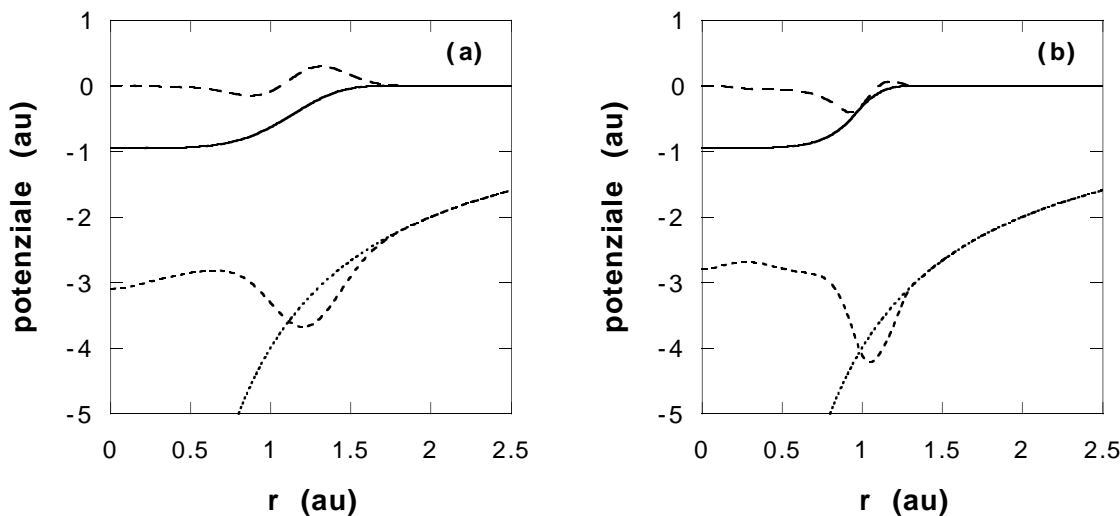


Figura 1.3: Pseudohamiltoniani per C: la configurazione di riferimento è s^2p^2 . $A(r)$, $B(r)$, $v_{ion}(r)$ e il potenziale coulombiano $-Z_v/r$ sono come in Fig. 1.2. (a) pseudohamiltoniano generato come descritto nel Rif. [6] (vedere nota 8); (b) pseudohamiltoniano con derivate logaritmiche migliorate (vedere anche testo).

per due diversi pseudohamiltoniani per il carbonio (mostrati in Fig. 1.3). In ogni riquadro sono mostrate tre derivate logaritmiche: quella per l'atomo full-core (linea continua) e per gli pseudopotenziali non locali (tratteggio corto), che sono così vicine l'una all'altra da apparire indistinguibili, e per lo pseudohamiltoniano (tratteggio lungo), che invece, prima o poi, si allontana dalle curve relative al full-core e agli pseudopotenziali non locali. I riquadri superiori si riferiscono a uno pseudohamiltoniano per il carbonio [Fig. 1.3(a)] che corrisponde esattamente a uno pseudopotenziale non locale⁸ (usato come confronto in tutta la figura); per essere precisi, nello stato di riferimento, lo pseudohamiltoniano del riquadro superiore ha le stesse funzioni d'onda s e p dello pseudopotenziale di confronto. Come ci si aspetta dalle Eq. (1.47,1.49), all'energia di riferimento (indicata un pallino), le derivate logaritmiche e la loro derivata prima in energia sono le stesse per tutte le curve. È anche evidente che le derivate in energia di ordine superiore devono essere diverse per lo pseudohamiltoniano e gli pseudopotenziali non locali, perché le due curve, a un certo punto, si allontanano in maniera

⁸Questo pseudohamiltoniano, costruito come descritto nel Rif. [6], è generato a partire da uno pseudopotenziale che differisce da quello di Bachelet, Hamann e Schlüter per il carbonio[40] per il solo raggio di core, più grande del 30%. I parametri usati per costruire lo pseudohamiltoniano sono $r_c = 1.2$, $k = 5$ e $a_0 = -0.95$.

evidente l'una dall'altra. Questo si può capire confrontando le Eq. (1.50,1.51) alla fine della Sez. 1.2.3, che riguardano la conservazione della norma “estesa” per gli pseudopotenziali[37] e per gli pseudohamiltoniani[38]. I riquadri superiori nella Fig. 1.4 mostrano che, per il carbonio, questo effetto è grande; che l'effetto sia fisicamente rilevante può essere visto da calcoli atomici indipendenti (non presentati qui), che mostrano come lo pseudohamiltoniano del riquadro superiore, le cui funzioni d'onda di stato fondamentale sono identiche a quelle dello pseudopotenziale, sia di fatto molto meno trasferibile (vedere nota 2). Per ottenere uno pseudohamiltoniano più trasferibile [Fig. 1.3(b)], è possibile, per esempio, per ogni valore di ℓ , imporre l'accordo fra derivate logaritmiche pseudo e full-core a due diverse energie di riferimento invece che a una sola. Tale vincolo addizionale, menzionato come una possibilità in Sez. 1.3.1, è stato realizzato nel caso del carbonio. I risultati sono mostrati nei pannelli inferiori della Fig. 1.4. Osserviamo che, dopo aver imposto questo vincolo aggiuntivo, le derivate logaritmiche dello pseudohamiltoniano seguono quelle full-core lungo un intervallo di energie più ampio; calcoli auto-consistenti atomici indipendenti suggeriscono che la trasferibilità è aumentata considerevolmente. Ciò nonostante, la trasferibilità dello pseudohamiltoniano per il carbonio (come per gli altri elementi della prima riga) risulta meno soddisfacente di quella degli pseudopotenziali non locali. Perché gli elementi della prima riga sono più difficili da trattare utilizzando gli pseudohamiltoniani? La ragione è probabilmente la seguente: per avere buone derivate superiori in energia delle derivate logaritmiche è necessario, nel caso dello pseudohamiltoniano, l'uguaglianza di integrali che contengono sia la funzione d'onda radiale di valenza che la funzione $a(r)$ [Eq. (1.51)]. Ma per gli atomi della prima riga c'è poca libertà per la funzione d'onda radiale p che non può essere molto diversa dalla funzione d'onda originale full-core: oltre ad avere lo stesso andamento nell'origine, la stessa norma all'interno del core, lo stesso valore e la stessa derivata nel raggio di core, *entrambe sono senza nodi*. Di conseguenza, abbiamo due diverse situazioni per gli pseudopotenziali non locali e gli pseudohamiltoniani della prima riga: per i primi la derivata logaritmica dell'onda p è quasi automaticamente molto buona [tutte le derivate in energia di ordine superiore danno integrali quasi identici nel caso full-core e degli pseu-

dopotenziali non locali nell'Eq. (1.50)], mentre per gli ultimi la stessa fisica (poca libertà nel core per le funzioni d'onda p della prima riga), assieme ad una funzione $a(r)$ diversa da 1 in tutti gli integrali rilevanti dell'Eq. (1.51), conduce - anche quando ci si sforza di migliorare la dipendenza dall'energia, come nei riquadri inferiori della Fig. 1.4 - a derivate logaritmiche peggiori. Derivate logaritmiche simili sono state ottenute per Na e Si. Questi elementi appartengono alla seconda riga, e il problema appena discusso per il carbonio (e in generale per gli elementi della prima riga), che naturalmente è legato alla sua marcata non località, è molto meno pronunciato; come mostrato nelle Fig. 1.5,1.6, si possono ottenere pseudohamiltoniani molto più trasferibili. In queste due figure le derivate logaritmiche s , p , e d sono mostrate in funzione dell'energia. Le curve con linea continua, tratteggio corto, e tratteggio lungo corrispondono rispettivamente all'atomo full-core, agli pseudopotenziali non locali e allo pseudohamiltoniano; qui stiamo usando gli pseudopotenziali non locali standard[40]. Per la costruzione dello pseudohamiltoniano per Na (Fig. 1.5), non è stata fatto alcun tentativo di ottimizzare l'onda d -wave [le quantità con $\ell = 2$ non sono state incluse nella funzione di costo, Eq. (1.74)]; la derivata logaritmica corrispondente è evidentemente meno buona, ma ancora accettabile, anche perché gli stati d sono sufficientemente alti in energia e non ci si aspetta che giochino un ruolo importante nella chimica e nella fisica dello ione. La derivata logaritmica s è invece, eccellente e quella p è addirittura meglio di quella degli pseudopotenziali non locali. Il caso del Si, infine, è una buona introduzione per il prossimo gruppo di calcoli atomici, non basati sulle derivate logaritmiche. Dai grafici delle derivate logaritmiche (Fig. 1.6) gli pseudopotenziali non locali sembrerebbero un po' più trasferibili dello pseudohamiltoniano (le curve relative agli pseudopotenziali seguono più da vicino quelle full-core), ma calcoli per il cristallo, e altri calcoli auto-consistenti, suggeriscono invece che la trasferibilità dello pseudohamiltoniano ottenuto per Si sia invece un po' meglio che non quella degli pseudopotenziali non locali. Queste differenze nella trasferibilità ionica sono a un livello di accuratezza molto più sottile di quanto appena visto per il carbonio, ma sono un buon modo per enfatizzare un aspetto della teoria degli pseudopotenziali ben nota agli specialisti, ma solo recentemente chiarita e razionalizzata da Teter[25], e

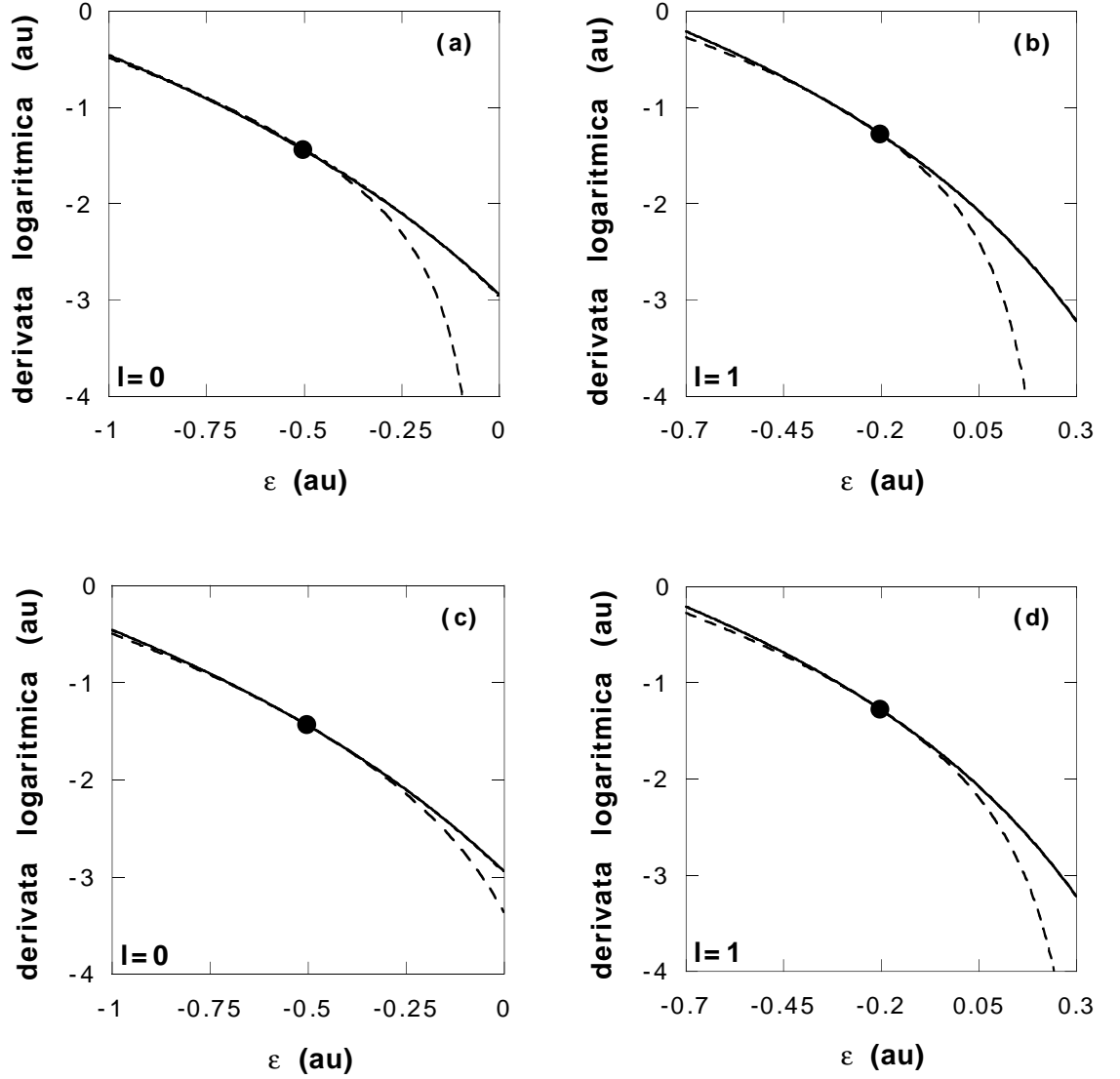


Figura 1.4: Derivate logaritmiche per C: derivate logaritmiche per (a) onda s e (b) onda p in funzione dell'energia per l'atomo full-core (linea continua), lo pseudohamiltoniano descritto nel testo (tratteggio) e gli pseudopotenziali non locali corrispondenti (tratteggio corto) nello stato fondamentale s^2p^2 (vedere testo). Derivate logaritmiche per (c) onda s e (d) onda p in funzione dell'energia per l'atomo (full-core), un diverso pseudohamiltoniano costruito come descritto nel testo (tratteggio) e gli stessi pseudopotenziali non locali di cui sopra (tratteggio corto) nello stato fondamentale s^2p^2 . Le curve relative agli pseudopotenziali e all'atomo full-core si sovrappongono quasi completamente.

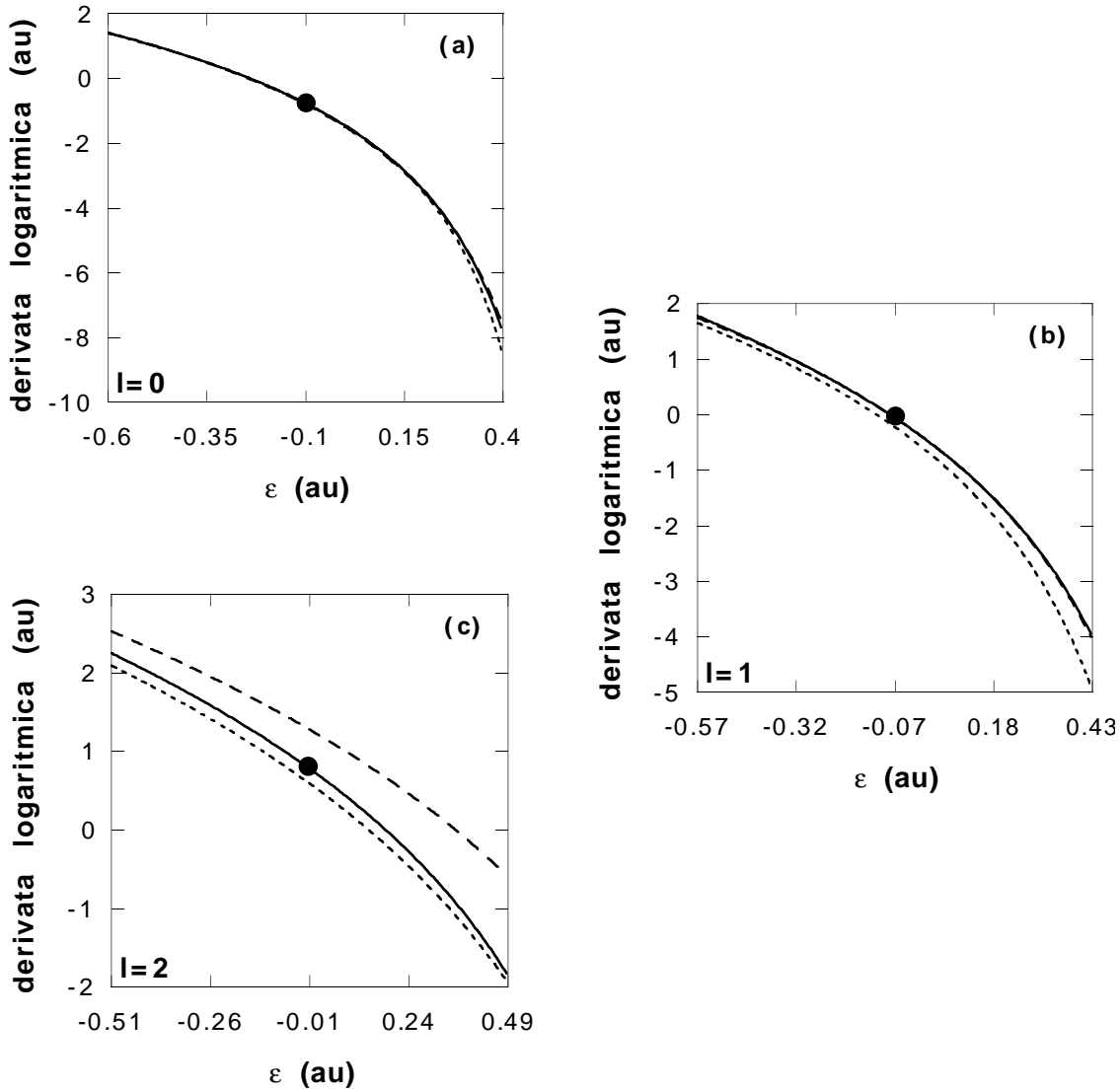


Figura 1.5: Derivate logaritmiche per Na: derivate logaritmiche per (a) onda s e (b) onda p in funzione dell'energia nel caso dell'atomo full-core (linea continua), lo pseudohamiltoniano (tratteggio) e gli pseudopotenziali non locali del Rif. [40] (tratteggio corto) nello stato fondamentale s^1 , (c) derivata logaritmica per l'onda d come sopra ma per la configurazione $s^{0.6}p^{0.1}d^0$. Le derivate logaritmiche s e p per l'atomo full-core e lo pseudohamiltoniano sono quasi indistinguibili. Vedere anche il testo.

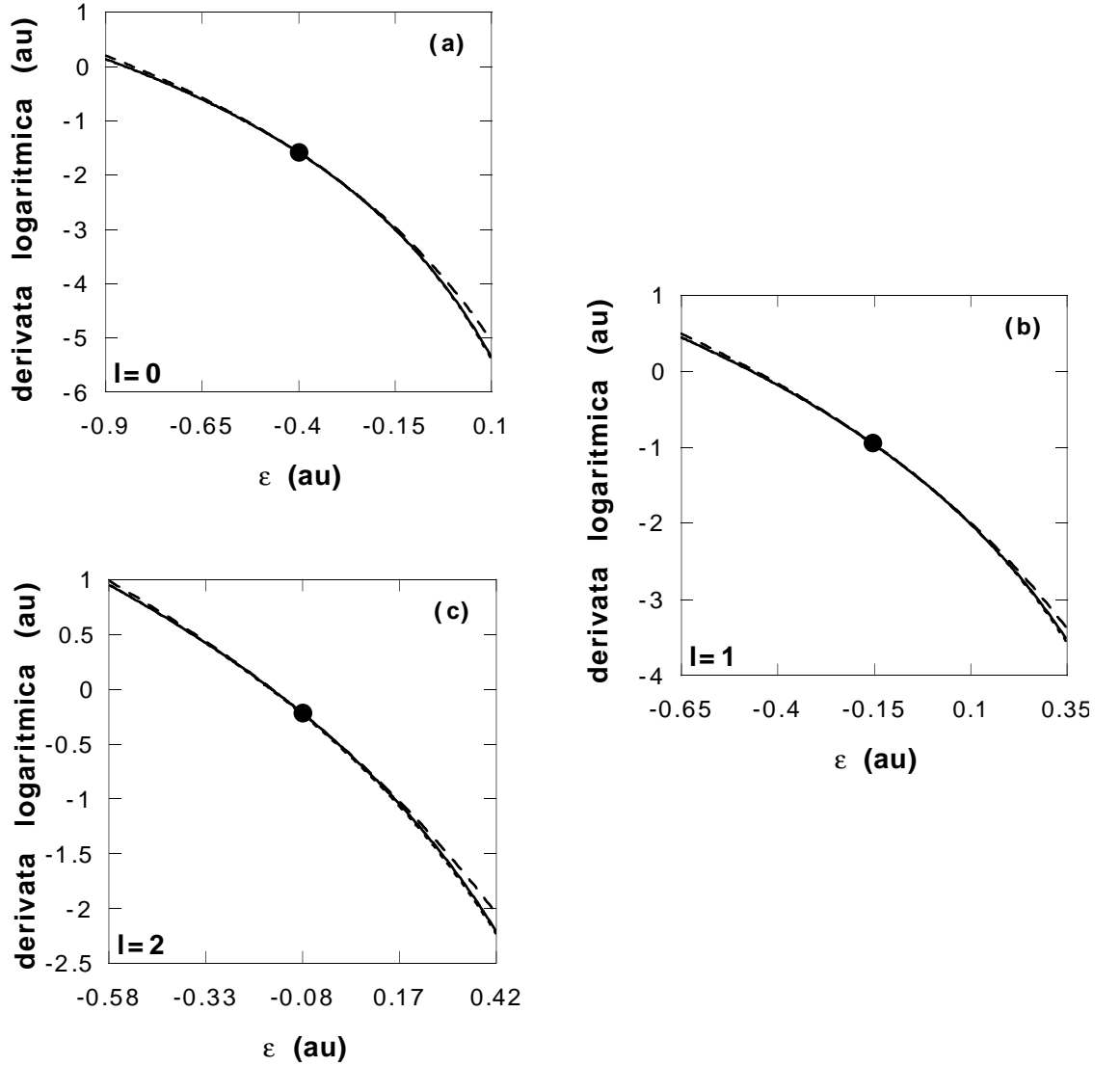


Figura 1.6: Derivate logaritmiche per Si: derivate logaritmiche per (a) onda s e (b) onda p in funzione dell'energia per l'atomo full-core (linea continua), lo pseudohamiltoniano (tratteggio) e gli pseudopotenziali non locali del Rif. [40] (tratteggio corto) nello stato fondamentale s^2p^2 , (c) derivata logaritmica per l'onda d come sopra ma per la configurazione di riferimento. Le curve per l'atomo full-core e gli pseudopotenziali si sovrappongono esattamente. Vedere anche il testo.

cioè che l'accordo fra le derivate logaritmiche full-core e pseudo da solo informazioni parziali sulla trasferibilità ionica. Spieghiamo brevemente perché. Per calcolare le derivate logaritmiche, la densità di carica elettronica viene congelata in qualche configurazione di valenza fissata (così anche il potenziale di schermo v_{HXC} è fissato), e l'equazione di Schrödinger radiale [Eq. (1.33,1.41,1.52)] viene risolta per vari valori dell'energia. L'energia viene variata ma la densità di carica rimane fissata. Tale variazione dell'energia, così, simula solo parzialmente il cambiamento nelle condizioni al contorno a cui sono soggette le funzioni d'onda quando l'atomo è posto in un ambiente molecolare o di stato solido: la corretta dipendenza dall'energia implica una buona trasferibilità ionica solo quando la quantità di carica di valenza contenuta nel core è molto piccola[24, 40], e core e valenza non hanno una sovrapposizione rilevante[52]. più accuratamente sono soddisfatte queste condizioni, meno importante è il riarrangiamento auto-consistente della densità di carica dentro al core, e più affidabili sono le derivate logaritmiche come indicatori di trasferibilità. Se, invece, queste due condizioni non sono soddisfatte, allora altri controlli che coinvolgono l'auto-consistenza sono più indicativi: ne presentiamo ora qualcuno. In confronto alla “chemical hardness” introdotta da Teter[25], che ha il vantaggio di trattare perturbazioni della carica non sferiche, i nostri controlli sono, a differenza di quest'ultima, estesi a distorsioni finite della carica. Abbiamo eseguito tre diverse prove che coinvolgono la risposta auto-consistente del sistema nelle seguenti condizioni: cambiamento delle condizioni al contorno atomiche, cambiamento nell'occupazione degli orbitali (presentate in questa sezione) e atomo nel cristallo (presentate nella prossima sezione). Abbiamo confrontato gli autovalori di singola particella ϵ_ℓ^{PH} , ϵ_ℓ^{FC} e le energie di eccitazione $E^{PH} - E_{gs}^{PH}$, $E^{FC} - E_{gs}^{FC}$ (e cioè le differenze di energie totali fra stati eccitati e lo stato fondamentale dell'atomo isolato) dello pseudohamiltoniano e dell'atomo full-core. Per completezza, abbiamo incluso nel nostro confronto anche gli pseudopotenziali non locali standard[40] (indice PP nelle quantità calcolate), in quanto sono stati ampiamente utilizzati con successo e perciò rappresentano un ovvio riferimento qualitativo. Prima di discutere i calcoli atomici, notiamo che il confronto appropriato è fra atomo full-core e sistemi di sola valenza all'interno della stessa (approssimata) DFT-

LDA. Ancora più importante è notare che, siccome la teoria degli pseudopotenziali si basa sull'approssimazione di core congelato[53], i calcoli full-core di confronto dovrebbero essere effettuati tenendo il core congelato nella configurazione di riferimento specificata. Mentre i risultati presentati in questa sezione non cambiano sostanzialmente se sostituiamo i calcoli a core congelato con calcoli a core rilassato, vedremo nella prossima sezione che questa differenza può essere rilevante in alcune situazioni. In relazione a quanto sopra detto, i risultati sperimentali sono riportati solo per completezza. Veniamo ora alla discussione delle prove di trasferibilità.

Nel primo gruppo di calcoli abbiamo valutato la variazione degli autovalori e delle energie di eccitazione pseudo rispetto alle corrispondenti quantità full-core, al variare delle condizioni al contorno per le funzioni d'onda atomiche. Gli orbitali atomici sono confinati in una sfera di raggio R , o, in altre parole, la posizione R della superficie nodale radiale, definita dall'equazione

$$\chi_\ell(R) = 0 \quad \forall \ell, \quad (1.101)$$

è stata mossa da $R = +\infty$ (atomo isolato) verso raggi sempre più piccoli, in modo tale da coprire un ampio intervallo di condizioni al contorno sferiche, che in qualche senso potrebbero simulare la transizione dall'atomo isolato verso situazioni di maggior impaccamento[38]. In questo gruppo di calcoli l'occupazione degli orbitali è sempre quella dello stato fondamentale, cosicché, per gli atomi considerati, solo i momenti angolari $\ell = 0, 1$ sono coinvolti; la Fig. 1.7 mostra i risultati per Si [riquadri superiori (a) e (b)] e Na [riquadri inferiori (c) e (d)]. I riquadri a sinistra nella Fig. 1.7 [(a) e (c)] riportano gli autovalori di singola particella. Essi mostrano $\Delta\epsilon_\ell^{PH} = \epsilon_\ell^{PH} - \epsilon_\ell^{FC}$ ($\Delta\epsilon_s^{PH}$ linea continua, $\Delta\epsilon_p^{PH}$ tratteggio lungo) e $\Delta\epsilon_\ell^{PP} = \epsilon_\ell^{PP} - \epsilon_\ell^{FC}$ ($\Delta\epsilon_s^{PP}$ tratteggio corto, $\Delta\epsilon_p^{PP}$ punteggiato), in funzione di R^{-1} . In Fig. 1.7(a) le curve s e p per lo pseudohamiltoniano per Si sono lievemente più piatte delle corrispondenti curve per gli pseudopotenziali non locali. Questo significa che, mentre la superficie nodale si muove verso l'interno, gli autovalori dello pseudohamiltoniano seguono quelli full-core un po' più da vicino che non quelli degli pseudopoten-

ziali, e cioè il comportamento opposto di quello che l'analisi delle derivate logaritmiche avrebbe suggerito⁹. In Fig. 1.7(c) la curva p per lo pseudohamiltoniano per Na è più vicina a zero di quella per gli pseudopotenziali non locali, questa volta in accordo con la previsione fatta sulla base delle derivate logaritmiche in Fig. 1.5(b). Il comportamento è l'opposto per le curve s (ma a un livello più fine), ancora una volta in contrasto con l'andamento delle derivate logaritmiche di Fig. 1.5(a), dove la curva per lo pseudohamiltoniano è più vicina a quella full-core. I riquadri di destra nella Fig. 1.7 [(b) e (d)] mostrano le differenze di energie totali: $\Delta E^{PH} = (E^{PH} - E_{gs}^{PH}) - (E^{FC} - E_{gs}^{FC})$ (linea continua) e $\Delta E^{PP} = (E^{PP} - E_{gs}^{PP}) - (E^{FC} - E_{gs}^{FC})$ (tratteggio), in funzione di R^{-1} , sia per Si che per Na. Qui la differenza fra pseudohamiltoniano e pseudopotenziali non locali è meno pronunciata. Per entrambi gli pseudoioni le curve sono molto simili, e questo può essere facilmente capito in termini di differenze fra atomo full-core e atomo di sola valenza; queste differenze sono, infatti, quasi indipendenti dallo schema con cui si costruisce lo pseudoatomo. Quando la densità di carica atomica è confinata all'interno di una sfera di raggio R sempre più piccolo, core e valenza incominciano a sovrapporsi in maniera significativa, e il funzionale di scambio e correlazione della DFT-LDA risente della mancanza della carica di core nel caso pseudo. Questo effetto sarebbe grandemente ridotto facendo uso della correzione di core non lineare ideata da Louie, Froyen, e Cohen[52], ma, d'altra parte, l'introduzione di una correzione dipendente dalla densità renderebbe il nostro pseudohamiltoniano inutilizzabile in connessione con le tecniche Monte Carlo Quantistico[54]. La Fig. 1.8 mostra le stesse quantità della Fig. 1.7, ma ora R è il raggio di una specie di sfera di Wigner-Seitz che contiene tutta la carica atomica e dove

$$\chi'_\ell(R) = 0 \quad \forall \ell. \quad (1.102)$$

In altre parole la differenza fra i risultati mostrati in Fig. 1.7 e quelli in Fig. 1.8 è che nel primo gruppo di “atomi confinati” la funzione d'onda si annulla sul bordo della sfera

⁹La differenza per $R^{-1} \rightarrow 0$ è dovuta alla scelta dello stato di riferimento: lo stato fondamentale è il riferimento per gli pseudopotenziali non locali ma non per lo pseudohamiltoniano (Sez. 1.4.2). Quindi, mentre gli pseudopotenziali non locali e l'atomo full-core hanno gli stessi autovalori per $R^{-1} \rightarrow 0$ (stato fondamentale), gli autovalori dello pseudohamiltoniano risultano lievemente diversi anche per $R^{-1} \rightarrow 0$.

confinante, mentre nel secondo gruppo si annulla la sua derivata. Ne il calcolo mostrato in Fig. 1.7, ne quello appena descritto, mostrato in Fig. 1.8, hanno un'esatta relazione con la formazione di legami nelle molecole o di bande nei solidi, ma possono fornire utili informazioni complementari su questo tipo di situazioni chimiche e quindi sulla trasferibilità. La situazione per Na in Fig. 1.8(c) è la stessa della Fig. 1.7(c), mentre il comportamento di Si in Fig. 1.8(a) è l'opposto rispetto alla Fig. 1.7(a). Questo dimostra come la trasferibilità a livelli molto accurati sia un gioco sottile: per Na, dove la trasferibilità complessiva è meno buona, le derivate logaritmiche e l'andamento degli autovalori danno un quadro coerente, mentre, per Si, dove la trasferibilità complessiva è ottima, i diversi controlli danno risultati contraddittori e le lievi differenze non possono essere dedotte in modo definitivo dalle derivate logaritmiche. Per i riquadri di destra nelle Fig. 1.8 [(b) e (d)] (differenze di energie totali), valgono le stesse considerazioni fatte per i corrispondenti riquadri in Fig. 1.7 con una quasi perfetta sovrapposizione fra le curve relative allo pseudohamiltoniano e agli pseudopotenziali.

Nel secondo gruppo di calcoli abbiamo studiato la risposta dello pseudohamiltoniano e degli pseudopotenziali non locali al cambiamento dell'occupazione degli orbitali (vedere Appendice A), confrontandola con la corrispondente risposta dell'atomo full-core[8]. Le energie di singola particella e le energie totali sono definite come sopra, e l'occupazione degli orbitali è stata variata in due modi diversi: in un caso la carica totale elettronica è stata tenuta costante ed uguale al suo valore nello stato fondamentale, e cioè la carica di valenza è stata spostata da un orbitale a un altro (Fig. 1.9), nell'altro caso la carica totale è stata variata, e diminuita a partire dallo stato fondamentale fino all'atomo doppiamente ionizzato nel caso dell'atomo di Si, o ionizzato una volta nel caso dell'atomo di Na (Fig. 1.10). più precisamente, nelle Fig. 1.9(a)-(b), l'occupazione degli orbitali s e p è stata variata da s^2p^2 (stato fondamentale) a s^0p^4 : le curve che corrispondono allo pseudohamiltoniano sono più piatte di quelle per gli pseudopotenziali non locali. Questo significa, ancora una volta, che lo pseudohamiltoniano segue più da vicino la risposta full-core che non gli pseudopotenziali. I riquadri inferiori nella Fig. 1.9 [(c) e (d)] mostrano, per Na, lo stesso andamento appena descritto

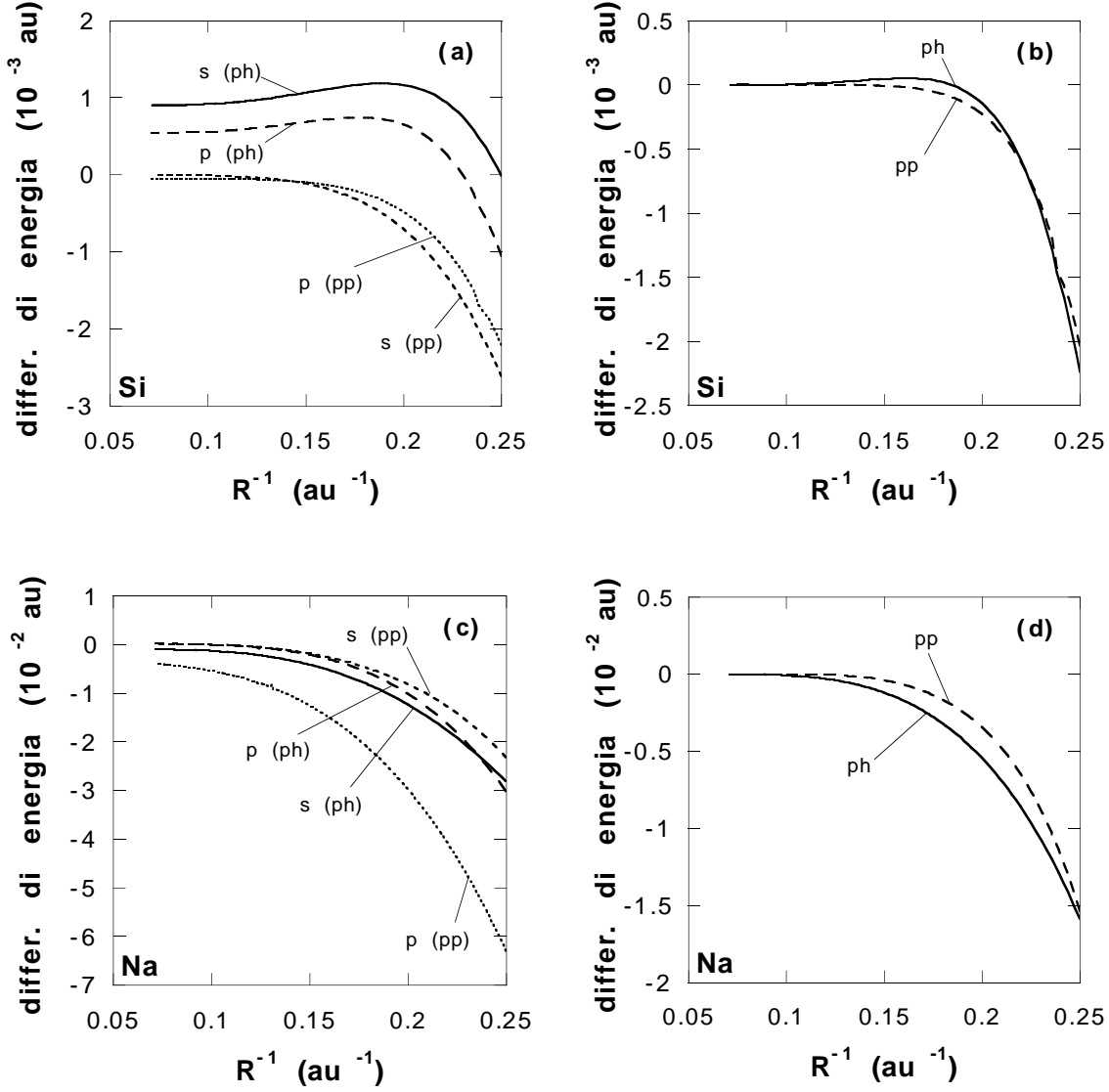


Figura 1.7: In funzione dell'inverso della distanza dal nucleo della superficie nodale radiale (R^{-1}) riportiamo: (a) differenze fra gli autovalori s e p dello pseudohamiltoniano e degli pseudopotenziali non locali [40] e quelli full-core per Si: $\Delta\epsilon^{PH} = \epsilon^{PH} - \epsilon^{FC}$ ($\Delta\epsilon_s^{PH}$: linea continua, $\Delta\epsilon_p^{PH}$: tratteggio lungo) e $\Delta\epsilon^{PP} = \epsilon^{PP} - \epsilon^{FC}$ ($\Delta\epsilon_s^{PP}$: tratteggio corto, $\Delta\epsilon_p^{PP}$: punteggiato), (b) differenze fra le energie di eccitazione dello pseudohamiltoniano e degli pseudopotenziali non locali [40] e quelle full-core per Si: $\Delta E^{PH} = (E^{PH} - E_{gs}^{PH}) - (E^{FC} - E_{gs}^{FC})$ (linea continua) e $\Delta E^{PP} = (E^{PP} - E_{gs}^{PP}) - (E^{FC} - E_{gs}^{FC})$ (tratteggio lungo), (c)-(d) le stesse quantità di (a)-(b) ma per Na. Le etichette s e p si riferiscono rispettivamente a $\ell = 0$ e $\ell = 1$, ph e pp allo pseudohamiltoniano e agli pseudopotenziali. Vedere anche il testo.

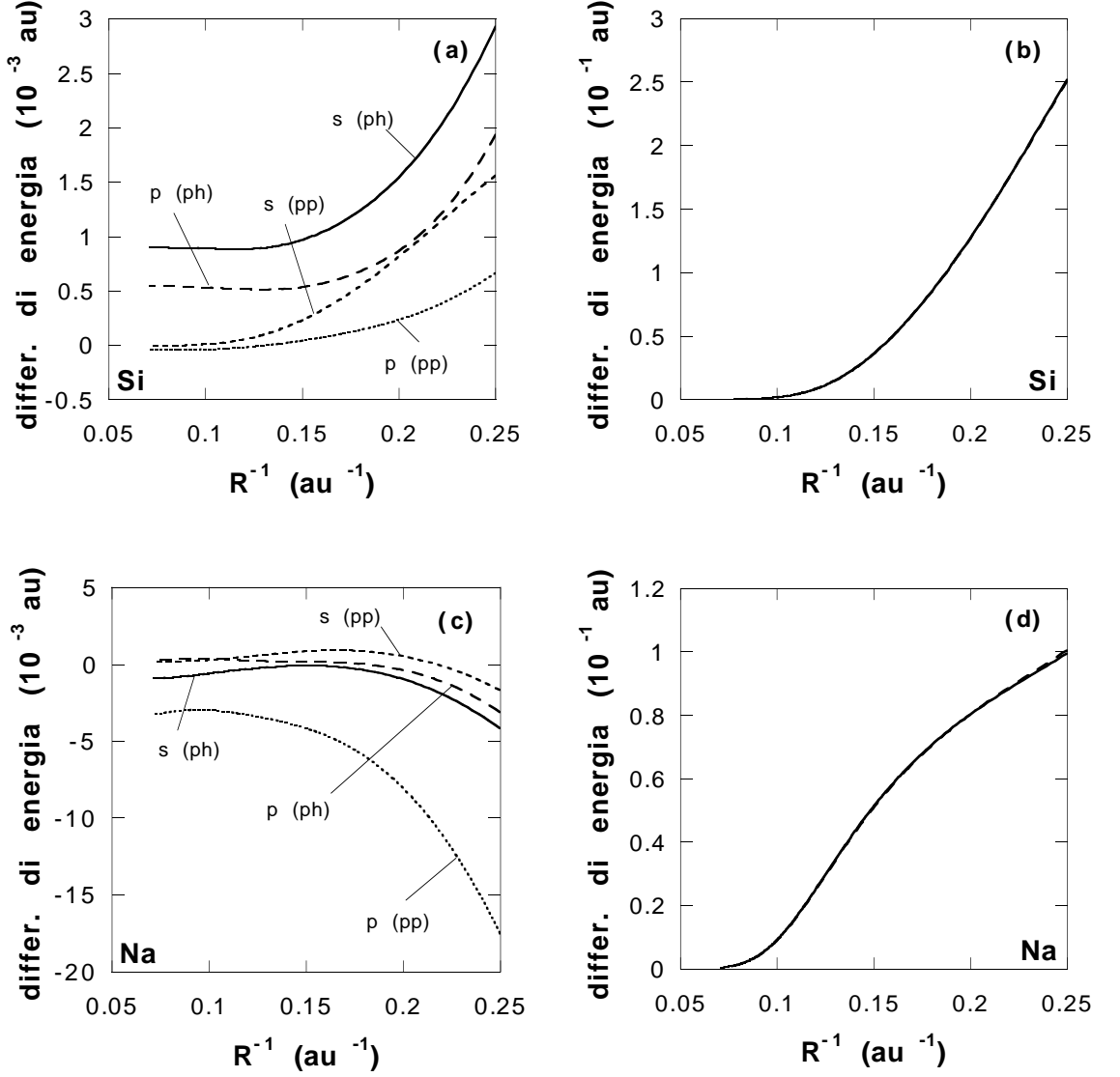


Figura 1.8: In funzione dell'inverso della distanza dal nucleo della superficie nodale della derivata radiale (R^{-1}) riportiamo: (a) differenze fra gli autovalori s e p dello pseudohamiltoniano e degli pseudopotenziali non locali [40] e quelli full-core per Si: $\Delta\epsilon^{PH} = \epsilon^{PH} - \epsilon^{FC}$ ($\Delta\epsilon_s^{PH}$: linea continua, $\Delta\epsilon_p^{PH}$: tratteggio lungo) e $\Delta\epsilon^{PP} = \epsilon^{PP} - \epsilon^{FC}$ ($\Delta\epsilon_s^{PP}$: tratteggio corto, $\Delta\epsilon_p^{PP}$: punteggiato), (b) differenze fra le energie di eccitazione dello pseudohamiltoniano e degli pseudopotenziali non locali [40] e quelle full-core per Si: $\Delta E^{PH} = (E^{PH} - E_{gs}^{PH}) - (E^{FC} - E_{gs}^{FC})$ (linea continua) e $\Delta E^{PP} = (E^{PP} - E_{gs}^{PP}) - (E^{FC} - E_{gs}^{FC})$ (tratteggio lungo), (c)-(d) le stesse quantità di (a)-(b) ma per Na. Le curve delle energie di eccitazione sono quasi perfettamente sovrapposte. Le etichette sono le stesse della Fig. 1.7. Vedere anche il testo.

per Si (ma ora l'occupazione va da s^1p^0 a s^0p^1), e la discrepanza fra pseudohamiltoniano e pseudopotenziali non locali è ancora più marcata. Nella “prova di ionizzazione” (Fig. 1.10), l'occupazione per l'orbitale s di Si è tenuta fissata e uguale a s^2 , mentre l'occupazione p varia da p^0 (ione Si^{2-}) a p^2 [riquadri superiori (a) e (b)]; per Na, viene variata solo l'occupazione dell'orbitale s da s^0 a s^1 , con l'orbitale p non occupato [Fig. 1.10(c)-(d)]. Sia per Si che per Na, la risposta degli autovalori dello pseudohamiltoniano è più simile a quella full-core degli pseudopotenziali non locali [Fig. 1.10, riquadri (a) e (c)]; nel caso delle energie totali la situazione è la stessa per Na [Fig. 1.10(d)] ma è l'opposto per Si [Fig. 1.10(b)], con la curva relativa agli pseudopotenziali non locali che segue meglio quella dell'atomo full-core. Osserviamo, comunque, che, sia in questa prova che in quelle precedenti, la scala di energia alla quale il comportamento dei due diversi pseudoatomi di silicio (pseudopotenziali non locali contro pseudohamiltoniano) può differire è così piccola che risulta legittimo chiedersi quale dei vari criteri di trasferibilità vada considerato come il più significativo per la misura di questa proprietà. A questo riguardo, non ci aspettiamo che calcoli su cristalli o molecole siano in grado fornire ulteriori dettagli al quadro già costruito mediante i controlli atomici (per quanto riguarda la portabilità) nel caso del sodio, mentre nel caso del silicio questi potrebbero rappresentare l'unico modo per risolvere il quesito. Questo quesito sulla trasferibilità non è in alcun modo un problema fazioso, dato che è precisamente su questa scala di energia che recenti calcoli Monte Carlo Variazionale e di Diffusione, basati rispettivamente su pseudopotenziali non locali e pseudohamiltoniani, sono risultati in disaccordo[5, 22].

1.4.4 Proprietà dei cristalli

Per quantificare le affermazioni precedenti, prima di concludere il nostro lavoro, presentiamo i risultati dei calcoli delle proprietà dei cristalli di silicio e sodio. Le condizioni al contorno sperimentate da un atomo isolato e all'interno di un cristallo sono diverse, e tale differenza è proprio il tipo di trasferibilità a cui siamo interessati; questi calcoli permettono un controllo indipendente e diretto della qualità dello pseudohamiltoniano in esame, e completano gli altri controlli discusso precedentemente (specialmente per quanto riguarda i termini non

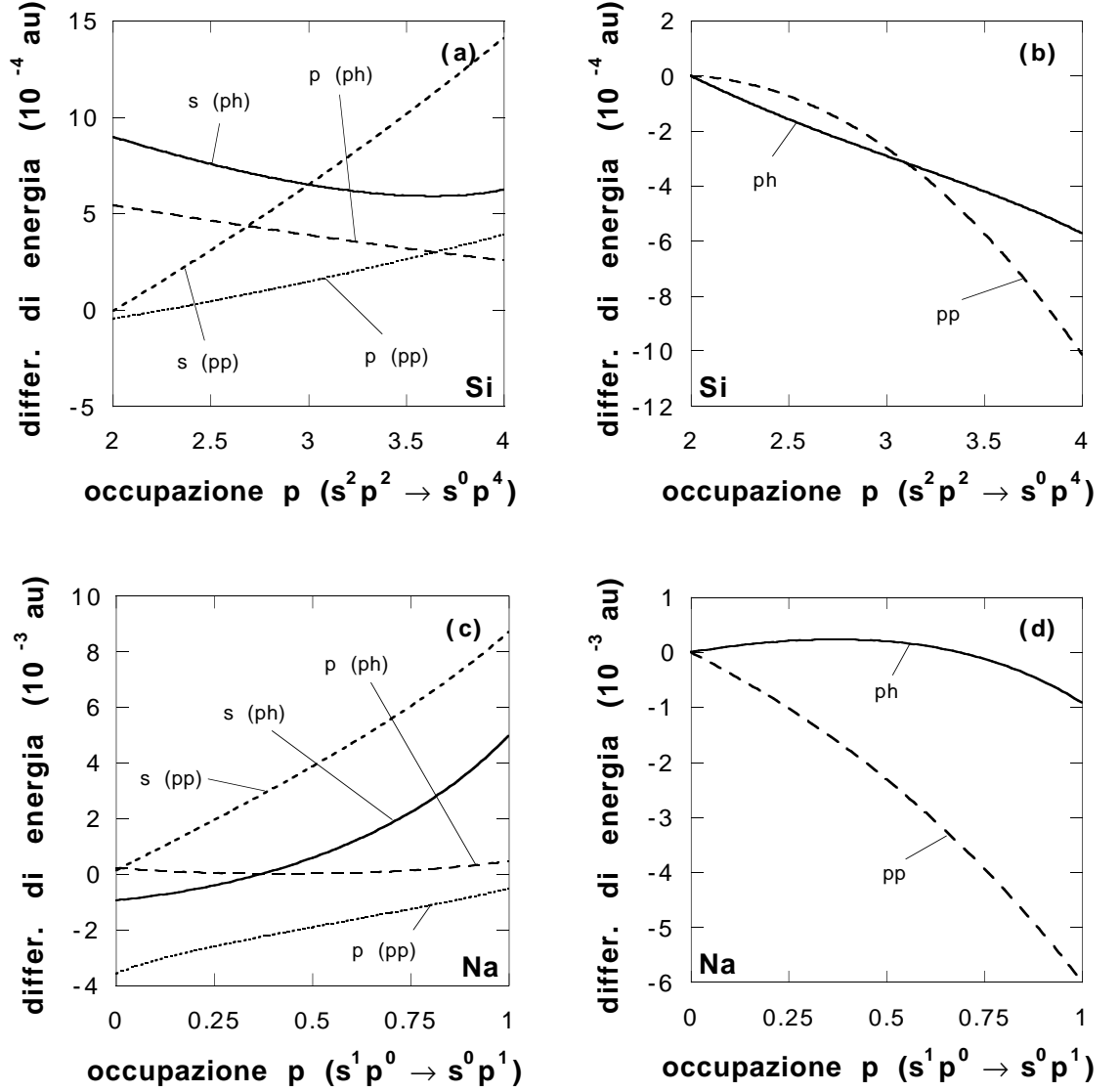


Figura 1.9: In funzione dei numeri di occupazione degli orbitali p , sono riportati: (a) $\Delta\epsilon^{PH}$ ($\Delta\epsilon_s^{PH}$: linea continua, $\Delta\epsilon_p^{PH}$: tratteggio lungo) e $\Delta\epsilon^{PP}$ ($\Delta\epsilon_s^{PP}$: tratteggio corto, $\Delta\epsilon_p^{PP}$: punteggiato), (b) ΔE^{PH} (linea continua) e ΔE^{PP} (tratteggio lungo) per Si variando la configurazione da $s^2 p^2$ a $s^0 p^4$, (c)-(d) le stesse quantità per Na variando l'occupazione da $s^1 p^0$ a $s^0 p^1$. Le etichette sono le stesse della Fig. 1.7. Vedere anche il testo.

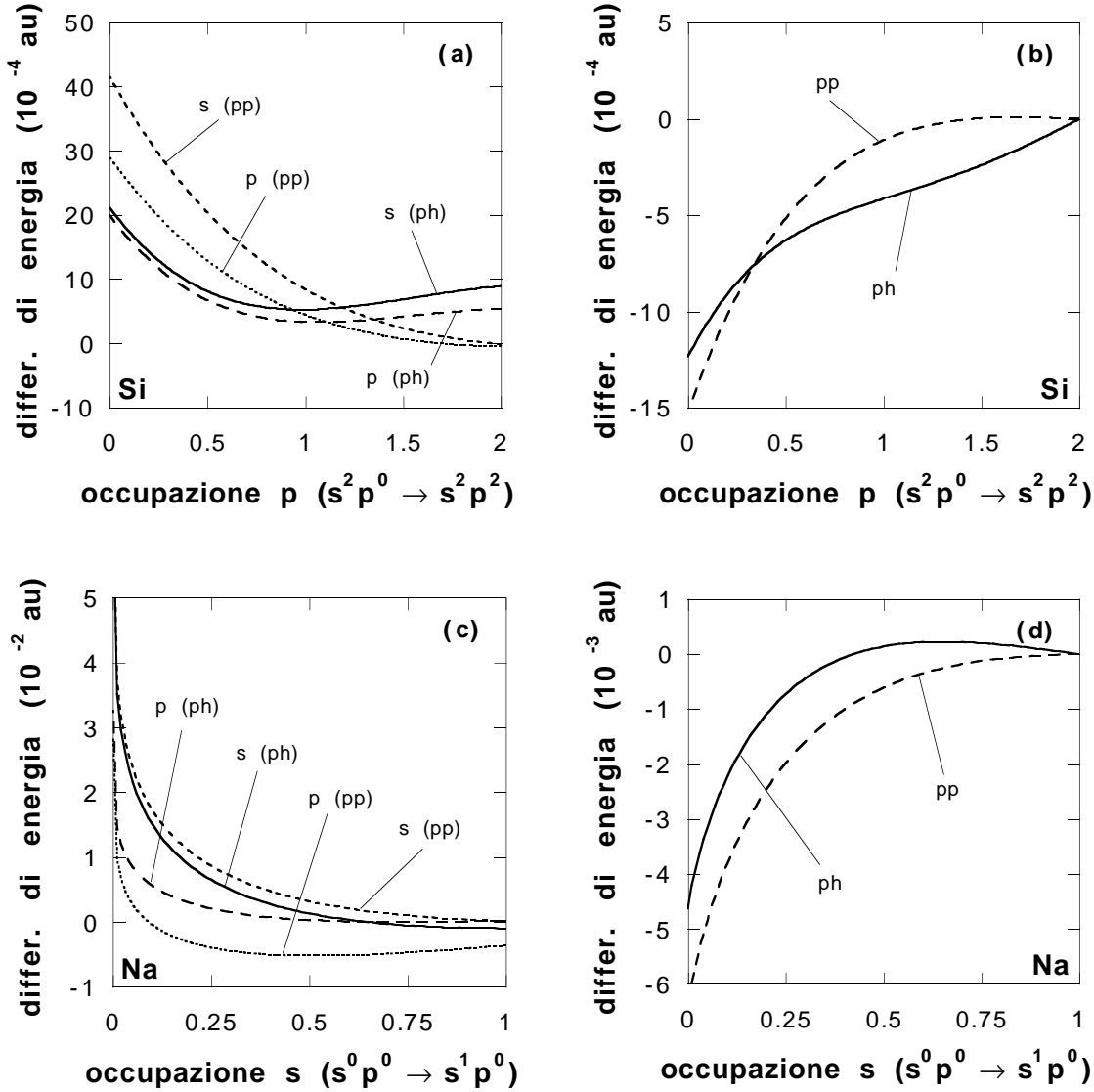


Figura 1.10: (a) $\Delta\epsilon^{PH}$ ($\Delta\epsilon_s^{PH}$: linea continua, $\Delta\epsilon_p^{PH}$: tratteggio lungo) and $\Delta\epsilon^{PP}$ ($\Delta\epsilon_s^{PP}$: tratteggio corto, $\Delta\epsilon_p^{PP}$: punteggiato) e (b) ΔE^{PH} (linea continua) e ΔE^{PP} (tratteggio lungo) sono riportati in funzione dell'occupazione dell'orbitale p per Si, variando la configurazione da s^2p^0 a s^2p^2 . (c)-(d) le stesse quantità per Na sono mostrate in funzione dell'occupazione dell'orbitale s , variando l'occupazione da s^0p^0 a s^1p^0 . Le etichette sono le stesse della Fig. 1.7. Vedere anche il testo.

sferici). Una buona trasferibilità dello pseudohamiltoniano in DFT-LDA appare inoltre un requisito importante per un utilizzo affidabile nelle simulazioni QMC.

Abbiamo studiato lo stato di Si isolante nella struttura del diamante e quella di Na metallico nella struttura bcc, a temperatura zero, mediante calcoli di energia totale in DFT-LDA. Un programma per il calcolo delle energie totali mediante il metodo delle onde piane, opportunamente modificato, è stato usato per eseguire i calcoli sul cristallo di sola valenza mediante pseudohamiltoniani e pseudopotenziali non locali[18], mentre il cristallo full-core è stato studiato con il metodo *full-potential LMTO*[55]. I calcoli sul cristallo full-core sono stati effettuati sia rilassando in modo auto-consistente tutti gli stati elettronici, sia congelando gli stati di core nella configurazione dello stato fondamentale atomico. In quest'ultimo caso, la densità di carica di core da utilizzare nel calcolo dei termini elettrostatico e di scambio e correlazione è stata ottenuta sovrapponendo i core dell'atomo libero a ogni iterazione, mentre gli autovalori di singola particella sono stati sommati all'energia totale (per i dettagli vedere il Rif.[56]). Val la pena di menzionare alcuni punti tecnici. Come si può vedere dalle Fig. 1.1 e 1.2, lo pseudohamiltoniano può richiedere un taglio in energia abbastanza elevato in un calcolo mediante onde piane. La convergenza delle quantità di interesse è stata raggiunta con un taglio in energia di 50 Ry per Si, e 25 Ry per Na. La costruzione delle matrici hamiltoniane è da 3 a 5 volte più veloce che con gli pseudopotenziali non locali standard, per le matrici con dimensione fino a 1500 utilizzate nel nostro caso. D'altra parte, l'operatore cinetico dello pseudohamiltoniano ha elementi di matrice non diagonali nello spazio di Fourier, cosicché il taglio in energia va incrementato gradualmente durante il ciclo auto-consistente; la convergenza di quest'ultimo risulta in qualche modo rallentata. Nei calcoli full-core, il cristallo viene rappresentato da sfere impacchettate strettamente, non sovrapposte, centrate sugli atomi, con il raggio *muffin-tin* pari al 98% di quello delle sfere. Come viene fatto di solito nei metodi LMTO[57], abbiamo inserito delle sfere vuote, dello stesso raggio di quelle atomiche, nelle regioni interstiziali della struttura del diamante. La base usata consiste di tre funzioni di Hankel con energie di decadimento -0.7 , -1.0 , -2.3 Ry, aumentate per $\ell \leq 2$, e ciò corrisponde a 27 LMTO per atomo. Sia per Si che per Na

abbiamo incluso in valenza gli stati $3s$, $3p$ e $3d$; per Na gli stati $2p$ sono stati trattati come stati di semi-core delocalizzati.

La somma nella zona Brillouin per Si è stata effettuata usando la griglia standard di 10 punti[58]. Il suo trattamento nel caso di Na merita alcuni commenti. Nei metalli, per accelerare la convergenza della somma in presenza di una superficie di Fermi, viene introdotto un allargamento gaussiano della distribuzione in energia degli stati elettronici. Questo è equivalente ad assegnare una temperatura fittizia, diversa da zero, al gas di elettroni. L'arrotondamento della distribuzione a gradino può essere ottenuto, per esempio, utilizzando una distribuzione di Fermi-Dirac con $T \neq 0$, o approssimando la funzione δ con una gaussiana di larghezza $\Delta = T$ (le due procedure non sono equivalenti). Il problema è che il risultato esatto si ha solo nel limite $\Delta \rightarrow 0$, e anche raggiungere questo limite richiede un numero molto elevato (in linea di principio infinito) di punti k . Una soluzione operativa è data dalla procedura di Gillan[59], che ha elaborato una ricetta per una “energia libera generalizzata” la quale devia dal valore $\Delta = 0$ per l'energia totale per termini di ordine $O(\Delta^3)$. De Vita[60] ha recentemente generalizzato questo metodo a schemi di allargamento arbitrari, indicando che la deviazione potrebbe essere addirittura di ordine $O(\Delta^4)$: il suo trattamento è equivalente in linea di principio alla formula di Methfessel-Paxton al primo ordine[61], come confermano i nostri risultati per il sistema studiato. Abbiamo lavorato con un allargamento gaussiano di 0.075 Ry, e abbiamo trovato che la griglia di 44 punti k [58] è sufficiente per dare risultati a convergenza. Osserviamo che il calcolo dell'energia totale per Δ finito con le tecniche suggerite da Needs, Martin, e Nielsen[62] e Fu e Ho[63], può dare dei problemi, per quanto riguarda l'energia coesiva e la costante reticolare di equilibrio. Questo fatto è chiaramente mostrato nella Tab. 1.6, dove confrontiamo schemi di allargamento differenti nel caso del cristallo di sodio; qui l'interazione elettrone ione è sempre descritta dallo pseudohamiltoniano. Sulla base della Tab. 1.6 abbiamo scelto lo schema di Gillan-De Vita in tutti i calcoli successivi per il cristallo di sodio. I confronti fra pseudohamiltoniano, pseudopotenziali e atomo full-core sono mostrati nelle Tab. 1.7 and 1.8. Le prestazioni dello pseudohamiltoniano per Si confrontate con i risultati full-core sono particolarmente buone;

metodo	a (Å)	E_c (eV/atomo)
Needs-Martin-Nielsen	4.07	0.61
Fu-Ho	4.36	1.10
Gillan-De Vita	4.19	0.86

Tavola 1.6: Confronto fra costanti reticolari a ed energie coesive E_c per Na, calcolate utilizzando lo pseudohamiltoniano con i metodi di Needs, Martin, e Nielsen[62], Fu e Ho[63], e Gillan e De Vita[59, 60].

metodo	a (Å)	E_c (eV/atomo)	B_0 (Kbar)
PH	5.44	5.04	912
PH (Rif. [6])	5.46	5.08	916
PP (Rif. [40])	5.37	5.31	968
Full-core (core congelato)	5.42	5.04	960
Full-core (core rilassato)	5.41	5.25	958
Full-core (core rilassato, FLAPW[64])	5.41	5.28	960
Full-core (core rilassato, FLAPW[65])	5.43	5.24	998
Esperimento (Rif. [66])	5.43	4.63	992

Tavola 1.7: Proprietà del cristallo di Si: a è la costante reticolare, E_c l'energia coesiva, B_0 la compressibilità; PH si riferisce allo pseudohamiltoniano, PP agli pseudopotenziali non locali (vedere testo).

in particolare, osserviamo che l'energia coesiva full-core è riprodotta molto bene. Per Na, le proprietà strutturali del cristallo descritto mediante lo pseudohamiltoniano sono eccellenti, ma l'energia coesiva è sottostimata rispetto al risultato full-core. I soddisfacenti risultati ottenuti sono particolarmente rilevanti in considerazione delle difficoltà normalmente incontrate nella descrizione dei metalli alcalini con il metodo degli pseudopotenziali. Sia Si che Na hanno una costante reticolare e una compressibilità leggermente più grandi quando il core viene congelato nella configurazione di atomo libero, un effetto discusso in maggior dettaglio nel Rif. [56].

1.5 Conclusioni

La costruzione di pseudohamiltoniani accurati, che sono strumenti utili e concettualmente semplici per le simulazioni Monte Carlo Quantistico dei sistemi di sola valenza, risulta molto

metodo	a (Å)	E_c (eV/atomo)	B_0 (Kbar)
PH	4.19	0.86	75
PP (Rif. [40])	3.87	1.15	110
Full-core	4.08	1.21	94
Esperimento (Rif. [66])	4.20	1.13	68

Tavola 1.8: Proprietà del cristallo di Na: a è la costante reticolare, E_c l'energia coesiva, B_0 la compressibilità; PH si riferisce allo pseudohamiltoniano, PP agli pseudopotenziali non locali (vedere testo).

più difficile di quella degli pseudopotenziali non locali; più precisamente è il loro carattere locale che, a causa dei vincoli da questo introdotti, rende inapplicabile ogni costruzione simile a quella di Hamann, Schlüter, e Chiang, basata sull'inversione numerica diretta. Inoltre, a differenza degli pseudopotenziali non locali, gli pseudohamiltoniani locali non possono, neanche in linea di principio, trattare gli ioni dei metalli di transizione, e incontrano difficoltà pratiche considerevoli anche con gli elementi (fortemente non locali) della prima riga. In questo lavoro, oltre a descrivere dettagliatamente il metodo dello pseudohamiltoniano, proponiamo una nuova tecnica numerica, basata sull'annealing simulato, e mostriamo come sia in grado di generare pseudohamiltoniani accurati per quella parte della Tavola Periodica per la quale cui essi funzionano bene (gli atomi con legami s - p). Mentre per gli elementi della prima riga e per quelli di transizione sono necessarie metodologie concettualmente meno semplici, per quegli elementi chimici per cui il metodo funziona, lo pseudohamiltoniano rappresenta uno strumento semplice e accurato.

Per quanto riguarda la trasferibilità, i nostri calcoli su atomi e cristalli suggeriscono che la qualità dei nostri nuovi pseudohamiltoniani è comparabile a quella degli pseudopotenziali non locali standard. Un risultato sorprendente è che l'energia coesiva di un cristallo di silicio calcolata a core rilassato è più grande (nella DFT-LDA) di circa 0.2 eV/atomo di quella a core congelato. Questa semplice osservazione, mai presentata in letteratura, almeno per quanto ne sappiamo, suggerisce qualche cautela nel confrontare, su questa scala di energia, le energie coesive calcolate mediante i metodi Monte Carlo Quantistico di sola valenza con

l'esperimento[5, 22]. Inoltre, per quanto riguarda le energie coesive, lo pseudohamiltoniano risulta leggermente più trasferibile, sulla scala degli 0.2 eV/atomo, degli pseudopotenziali non locali standard. Una spiegazione di quanto osservato probabilmente si troverebbe includendo consistentemente la correzione di core non lineare di Louie, Froyen, e Cohen[52] in tutti i nostri calcoli. Questo non è stato fatto perché, come già sottolineato, questa correzione dipendente dalla densità non si presta a un'estensione delle simulazioni Monte Carlo Quantistico di sola valenza[54], che sono la motivazione iniziale per l'introduzione degli pseudohamiltoniani locali.

La nostra idea è che, a livello delle piccole differenze menzionate prima nel caso del silicio, ma anche per spiegare alcuni dei risultati ottenuti per il sodio (Tab. 1.7 e 1.8), probabilmente lo pseudohamiltoniano tende a compensare gli errori dovuti al termine di Hartree e di scambio e correlazione, nel senso discusso da Teter[25], in un modo un po' più efficiente che non gli pseudopotenziali non locali. Controlli sistematici della "chemical hardness" così come una tabulazione completa degli pseudohamiltoniani per gli elementi *s-p*, rappresentano la naturale estensione di questo lavoro, e sono attualmente in corso[39].

1.6 Ringraziamenti

Ringraziamo D. M. Ceperley, M. Foulkes e G. Senatore per le utili discussioni intercorse. Un ringraziamento particolare va a L. Mitáš che per primo ha avuto l'idea di utilizzare i metodi stocastici per costruire gli pseudohamiltoniani e ci ha insegnato cos'è uno RST. Ringraziamo anche A. Continenza, C. Filippi e C. J. Umrigar per averci fornito i risultati di due calcoli FLAPW indipendenti per il cristallo di silicio. Stefano Baroni e Michael Methfessel ci hanno messo a disposizione i loro programmi PPPW e FPLMTO per i calcoli sul cristallo. Questo lavoro è stato sostenuto dal Consiglio Nazionale delle Ricerche, nell'ambito del "Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo" mediante i finanziamenti no. 89.0006.69 e 89.00051.69.

2 Monte Carlo Quantistico

2.1 Introduzione

In questo capitolo ci occuperemo di una particolare classe di tecniche numeriche basate su cammini aleatori in uno spazio multidimensionale, che rientrano nella categoria dei metodi Monte Carlo ideati per risolvere l'equazione di Schrödinger per un sistema di molte particelle interagenti (e per questo chiamate anche con il termine simulazioni quantistiche). Esse sono indicate generalmente con il nome di metodi Monte Carlo Quantistico (QMC). Ancora più in dettaglio, tratteremo il metodo Monte Carlo Variazionale (VMC)[16] e il metodo Monte Carlo di Diffusione (DMC)[16]. Entrambi i metodi ci consentono di calcolare il valor medio di un operatore hermitiano su una funzione d'onda a molti corpi (a spin totale fissato) senza dover ricorrere all'approssimazione di particella singola, e quindi a una descrizione approssimata dell'interazione elettrone-elettrone. Nel VMC la funzione d'onda a molti corpi è nota (nel nostro caso consideriamo un'approssimazione per lo stato fondamentale del sistema in esame), e il calcolo del valor medio si riduce al calcolo di un integrale multidimensionale. Nel DMC, invece, la funzione d'onda a molti corpi è lo stato fondamentale sconosciuto del sistema e, prima di passare al calcolo del valor medio, il metodo ci permette di estrarre le informazioni necessarie sullo stato fondamentale, mediante un'opportuna simulazione dell'equazione di Schrödinger. L'idea base dei metodi QMC è quella di calcolare i valori medi di operatori hermitiani (noi considereremo sempre l'operatore hamiltoniano \hat{H}) mediante una somma su un numero finito di punti significativi, ottenuti attraverso un campionamento stocastico dello spazio d'interesse (campionamento d'importanza), evitando così di valutare direttamente un complicato integrale multidimensionale. Per entrambi i metodi i risultati numerici non sono deterministici ma sono sempre accompagnati da un'incertezza statistica dovuta al campionamento Monte Carlo finito, che però è sempre possibile ridurre sotto una soglia stabilita pur di lasciare procedere la simulazione per un tempo sufficientemente lungo.

L'accuratezza dei metodi Monte Carlo Quantistico dipende da vari parametri; tra questi è importantissimo il numero N di punti che si utilizzano per il calcolo dei valori medi: come in tutti i metodi statistici il risultato del calcolo è affetto da un errore inversamente

proporzionale alla radice quadrata di N . La necessità di usare N molto grandi, assieme alle caratteristiche della particolare tecnica utilizzata per raggiungere il campionamento di importanza desiderato, rende i metodi QMC molto più dispendiosi, dal punto di vista computazionale, di altri metodi di risoluzione dell'equazione di Schrödinger che però sono approssimati. È stato stimato che negli atomi il tempo di calcolo cresce con il numero atomico Z in modo proporzionale a $Z^{5.5}$ - $Z^{6.5}$ [2, 3]. La presenza degli elettroni di core rallenta drasticamente la convergenza della simulazione in modo tale che calcoli su atomi con numero atomico maggiore di 10, cioè più pesanti del Neon, sono stati fino a questo momento praticamente improponibili. La ragione di tale rallentamento sta nel fatto che il corretto campionamento della regione di core, in cui la densità di carica varia rapidamente su brevi distanze è enormemente più dispendioso di quello necessario nella regione di valenza in cui le variazioni della densità di carica sono molto più dolci e distribuite su distanze più grandi. Di conseguenza, i metodi QMC sono stati finora usati con successo per il calcolo delle proprietà di atomi leggeri e di piccole molecole[4]. Solo molto recentemente, grazie ai progressi nella tecnica delle simulazioni QMC[6, 7] e alla disponibilità di calcolatori sempre più potenti, l'interesse si sta spostando anche verso sistemi più complessi quali i solidi[5, 22, 7]. Se potessimo trattare solo gli elettroni di valenza, che sono gli unici che partecipano direttamente alla determinazione di molte proprietà d'interesse, la situazione sarebbe molto semplificata. Indicando con Z_v il numero degli elettroni di valenza, nel caso di un atomo di sola valenza è stata ipotizzata una dipendenza del tempo di calcolo da Z_v che non dovrebbe andare oltre $Z_v^{3.4}$ [3]. La differenza con il caso in cui si considerano tutti gli elettroni è sostanziale: prima di tutto l'esponente è più piccolo, inoltre, mentre il numero atomico Z aumenta lungo tutta la Tavola Periodica, il numero Z_v rimane piccolo per tutti gli atomi.

Dalle considerazioni appena fatte e da quanto detto nella Sez. 1.1 appare chiaro come l'utilizzo di operatori di sola valenza costituisca un'approssimazione necessaria (almeno allo stato attuale) per affrontare lo studio di sistemi "complessi" mediante i metodi QMC. Poiché, però, non sono ancora disponibili operatori di sola valenza generati consistentemente in

ambito QMC[10], riteniamo che l'utilizzo di hamiltoniani generati all'interno di teorie che trattano in maniera approssimata l'interazione elettrone-elettrone (DFT-LDA) debba essere soggetto a verifica al fine di quantificare l'incidenza dell'ulteriore approssimazione, e cioè di quantificare quella che chiamiamo portabilità dell'operatore fra metodi che approssimano in modo diverso l'interazione elettrone-elettrone. I risultati presentati nella Sez. 2.6 hanno come fine proprio quello di misurare la portabilità di una particolare classe di operatori di sola valenza e cioè degli pseudopotenziali standard a norma conservata[40].

Il nostro studio inizia con il VMC e l'estensione proposta da Umrigar *et al.*[69] per poi continuare con il DMC sia nella versione di Reynolds *et al.*[4] che in quella più recente di Umrigar *et al.*[70]

2.2 Monte Carlo Variazionale

Nella sua versione più semplice il metodo permette di ottenere, data una funzione d'onda di prova Ψ_T (a norma finita) per il problema di Schrödinger a molti corpi¹, il valor medio di qualunque operatore hermitiano \hat{A} e cioè

$$\langle \hat{A} \rangle = \frac{\langle \Psi_T | \hat{A} | \Psi_T \rangle}{\langle \Psi_T | \Psi_T \rangle}, \quad (2.1)$$

anche se nel seguito, per fissare le idee, porremo sempre $\hat{A} = \hat{H}$ operatore hamiltoniano: $E = \langle \hat{H} \rangle$ è l'energia variazionale. Introdotto il vettore \mathbf{R} nello spazio delle configurazioni si ha

$$E = \frac{\int \Psi_T^* \hat{H} \Psi_T d\mathbf{R}}{\int \Psi_T^* \Psi_T d\mathbf{R}} = \frac{\int |\Psi_T|^2 \frac{\hat{H} \Psi_T}{\Psi_T} d\mathbf{R}}{\int |\Psi_T|^2 d\mathbf{R}}. \quad (2.2)$$

Se ora introduciamo un insieme di N punti (o configurazioni o *walker*) nello spazio delle configurazioni, $\{\mathbf{R}_i\}$, distribuiti secondo $|\Psi_T|^2$, abbiamo che

$$E \simeq \frac{1}{N} \sum_{i=1}^N \frac{\hat{H} \Psi_T(\mathbf{R}_i)}{\Psi_T(\mathbf{R}_i)}, \quad (2.3)$$

¹Per i sistemi da noi considerati, osserviamo che l'invarianza per inversione temporale dell'equazione di Schrödinger ci permette di scegliere funzioni d'onda ovunque reali[28].

entro l'incertezza statistica dovuta al numero finito di punti (vedremo come questa si può stimare nella Sez. 2.5).

Il tipo di campionamento in cui i punti vengono distribuiti secondo $|\Psi_T|^2$ prende il nome di campionamento di importanza e ha un enorme vantaggio rispetto a un campionamento uniforme mediante il quale la stima dell'energia diventa la media pesata

$$E \simeq \frac{\sum_{i=1}^N |\Psi_T(\mathbf{R}_i)|^2 \frac{\hat{H}\Psi_T(\mathbf{R}_i)}{\Psi_T(\mathbf{R}_i)}}{\sum_{i=1}^N |\Psi_T(\mathbf{R}_i)|^2}. \quad (2.4)$$

Infatti, poiché la regione dello spazio delle configurazioni che contribuisce al valore di E , Eq. (2.2), è quella in cui Ψ_T differisce da zero, il campionamento di importanza garantisce la massima efficienza concentrando i suoi punti solo nella regione rilevante. Al contrario, nel campionamento uniforme, molti dei punti si troveranno in regioni in cui Ψ_T è molto prossima a zero e perciò porteranno un contributo irrilevante nell'Eq. (2.4). Questo implica che solo una piccola parte dei punti contribuisce significativamente al valore di E e, a parità nel numero di punti con il campionamento di importanza, l'incertezza statistica risulta molto maggiore.

La distribuzione secondo $|\Psi_T|^2$ viene fatta utilizzando l'algoritmo di Metropolis *et al.*[44], introducendo una dinamica stocastica per le configurazioni, governata dalla legge:

$$\mathbf{R} \rightarrow \mathbf{R}' = \mathbf{R} + \lambda \boldsymbol{\eta} \quad (2.5)$$

con $\lambda > 0$ arbitrario ma dell'ordine di grandezza della scala di lunghezza tipica del problema e $\boldsymbol{\eta}$ vettore stocastico le cui componenti hanno distribuzione uniforme[43] nell'intervallo $(-1, 1)$. La dinamica è subordinata alla probabilità di transizione²:

$$P(\mathbf{R} \rightarrow \mathbf{R}') = \min \left\{ \left| \frac{\Psi_T(\mathbf{R}')}{\Psi_T(\mathbf{R})} \right|^2, 1 \right\}. \quad (2.6)$$

La scelta di λ deve essere tale da garantire la massima efficienza di campionamento e cioè un numero medio di transizioni intorno al 50% (operativamente fra il 30% e il 70%).

²In pratica, la realizzazione di un evento che ha probabilità P di verificarsi, può essere decisa usando i numeri (pseudo-) casuali. Detto ξ un tale numero generato dalla distribuzione uniforme sull'intervallo $[0, 1)$, sia \bar{P} la parte intera di $P + \xi$: se $\bar{P} = 1$ l'evento viene realizzato, altrimenti, se $\bar{P} = 0$, no.

Detto K il nucleo stocastico associato alla dinamica introdotta e N_p il numero di particelle nel sistema, risulta:

$$K(\mathbf{R} \rightarrow \mathbf{R}') = \begin{cases} \frac{1}{(2\lambda)^{3N_p}} & |\mathbf{R} - \mathbf{R}'| < \lambda \\ 0 & |\mathbf{R} - \mathbf{R}'| \geq \lambda \end{cases} \quad (2.7)$$

ed è facile verificare che è soddisfatta l'equazione del bilancio dettagliato

$$\frac{P(\mathbf{R} \rightarrow \mathbf{R}')}{P(\mathbf{R}' \rightarrow \mathbf{R})} = \frac{|\Psi_T(\mathbf{R}')|^2}{|\Psi_T(\mathbf{R})|^2} \frac{K(\mathbf{R}' \rightarrow \mathbf{R})}{K(\mathbf{R} \rightarrow \mathbf{R}')}. \quad (2.8)$$

Questa implica la condizione di equilibrio

$$\frac{K(\mathbf{R} \rightarrow \mathbf{R}')P(\mathbf{R} \rightarrow \mathbf{R}')}{K(\mathbf{R}' \rightarrow \mathbf{R})P(\mathbf{R}' \rightarrow \mathbf{R})} = \frac{|\Psi_T(\mathbf{R}')|^2}{|\Psi_T(\mathbf{R})|^2}, \quad (2.9)$$

e perciò la distribuzione di equilibrio è $|\Psi_T|^2$. Infatti, posto per semplicità

$$f(\mathbf{R}) = |\Psi_T(\mathbf{R})|^2, \quad (2.10)$$

sia $\rho(\mathbf{R})$ la densità con cui sono distribuiti gli N punti nello spazio delle configurazioni. Calcoliamo il numero dei punti per unità di volume che vanno da \mathbf{R} a \mathbf{R}' , $N(\mathbf{R} \rightarrow \mathbf{R}')$, e da \mathbf{R}' a \mathbf{R} , $N(\mathbf{R}' \rightarrow \mathbf{R})$. Si ha:

$$N(\mathbf{R} \rightarrow \mathbf{R}') = \rho(\mathbf{R})K(\mathbf{R} \rightarrow \mathbf{R}')P(\mathbf{R} \rightarrow \mathbf{R}') \quad (2.11)$$

$$N(\mathbf{R}' \rightarrow \mathbf{R}) = \rho(\mathbf{R}')K(\mathbf{R}' \rightarrow \mathbf{R})P(\mathbf{R}' \rightarrow \mathbf{R}) \quad (2.12)$$

Il movimento netto da \mathbf{R} a \mathbf{R}' è perciò

$$\begin{aligned} N(\mathbf{R} \rightarrow \mathbf{R}') - N(\mathbf{R}' \rightarrow \mathbf{R}) &= \\ &= \rho(\mathbf{R})K(\mathbf{R} \rightarrow \mathbf{R}')P(\mathbf{R} \rightarrow \mathbf{R}') - \rho(\mathbf{R}')K(\mathbf{R}' \rightarrow \mathbf{R})P(\mathbf{R}' \rightarrow \mathbf{R}). \end{aligned} \quad (2.13)$$

All'equilibrio questo deve essere nullo e quindi

$$\rho(\mathbf{R})K(\mathbf{R} \rightarrow \mathbf{R}')P(\mathbf{R} \rightarrow \mathbf{R}') = \rho(\mathbf{R}')K(\mathbf{R}' \rightarrow \mathbf{R})P(\mathbf{R}' \rightarrow \mathbf{R}), \quad (2.14)$$

da cui si ottiene, utilizzando l'Eq. (2.9)

$$\frac{\rho(\mathbf{R})}{\rho(\mathbf{R}')} = \frac{K(\mathbf{R}' \rightarrow \mathbf{R})P(\mathbf{R}' \rightarrow \mathbf{R})}{K(\mathbf{R} \rightarrow \mathbf{R}')P(\mathbf{R} \rightarrow \mathbf{R}')} = \frac{f(\mathbf{R})}{f(\mathbf{R}')}. \quad (2.15)$$

Se $\rho(\mathbf{R})$ è normalizzato a N e $f(\mathbf{R})$ è normalizzato a 1 si ha:

$$\frac{\rho(\mathbf{R})}{\rho(\mathbf{R}')} = \frac{f(\mathbf{R})}{f(\mathbf{R}')} \quad (2.16)$$

$$\Rightarrow \frac{\int \rho(\mathbf{R}) d\mathbf{R}}{\rho(\mathbf{R}')} = \frac{\int f(\mathbf{R}) d\mathbf{R}}{f(\mathbf{R}')} \quad (2.17)$$

da cui segue

$$\rho(\mathbf{R}') = N f(\mathbf{R}') \quad (2.18)$$

e quindi all'equilibrio gli N punti campionano f . Una volta raggiunto l'equilibrio siamo in grado di calcolare l'energia mediante l'Eq. (2.2). Operativamente, partendo da un qualunque insieme di configurazioni, applicando in successione l'Eq. (2.5) che genera un nuovo insieme (virtuale) e l'Eq. (2.6) che seleziona quali delle configurazioni virtuali prendono il posto di quelle di partenza, e iterando per un numero sufficiente di volte questa procedura, si arriva alla condizione di equilibrio, Eq. (2.9), e cioè a un insieme di punti distribuiti secondo $|\Psi_T|^2$.

Se la funzione Ψ_T contiene dei parametri liberi, è possibile utilizzare il metodo appena esposto per determinare il valore dei parametri per cui $\langle \hat{H} \rangle$ risulta minimo (questo è utile per il DMC). La via più diretta è quella di affiancare al VMC un algoritmo di minimizzazione (quale il metodo dei semplici[43]). In questo modo, se $\{\alpha_i\}$ è l'insieme dei parametri da cui dipende $\Psi_T(\mathbf{R}, \{\alpha_i\})$ e $E(\{\alpha_i\})$ il corrispondente valore di $\langle \hat{H} \rangle$, è possibile determinare l'insieme $\{\bar{\alpha}_i\}$ tale che

$$E(\{\bar{\alpha}_i\}) = \min_{\{\alpha_i\}} E(\{\alpha_i\}). \quad (2.19)$$

Naturalmente è necessario eseguire il calcolo di $E(\{\alpha_i\})$ mediante VMC per ogni passo della procedura di minimizzazione e ciò può costituire un onere computazionale rilevante, soprattutto se il numero dei parametri è grande. Un algoritmo alternativo, molto più conveniente, è stato ideato da Umrigar *et al.*[69].

2.2.1 Minimizzazione della varianza

Invece di lavorare direttamente con $\langle \hat{H} \rangle$, nell'algoritmo di Umrigar si minimizza la varianza

$$\sigma^2(\hat{H}) = \langle \hat{H}^2 \rangle - \langle \hat{H} \rangle^2 \geq 0, \quad (2.20)$$

Se Ψ_T è autofunzione dello stato fondamentale (o di uno stato eccitato) di \hat{H} si ha

$$\hat{H}\Psi_T = E\Psi_T, \quad (2.21)$$

da cui segue

$$\langle \hat{H} \rangle = E, \quad (2.22)$$

$$\langle \hat{H}^2 \rangle = E^2 \quad (2.23)$$

e dunque, solo in questo caso, risulta

$$\sigma^2(\hat{H}) = E^2 - E^2 = 0. \quad (2.24)$$

Da quanto detto risulta sempre

$$\sigma^2(\hat{H}) \geq 0 \quad (2.25)$$

in cui vale l'uguaglianza solo se Ψ_T è autofunzione di \hat{H} . Per questo, trovare i parametri che rendono minima la varianza $\sigma^2(\hat{H})$ è equivalente (a patto di partire da una Ψ_T sufficientemente vicina allo stato fondamentale) a trovare i parametri $\{\alpha_j\}$ che minimizzano l'energia.

Cerchiamo, ora, di ottenere un'equazione equivalente alla (2.20), che allo stesso tempo si possa minimizzare in un modo abbastanza semplice senza dover ricorrere a un calcolo VMC completo ad ogni passo della procedura di minimizzazione. Per la (2.2) risulta che

$$E(\{\alpha_j\}) = \langle \hat{H} \rangle = \left\langle \frac{\hat{H}\Psi_T}{\Psi_T} \right\rangle, \quad (2.26)$$

e scrivendo $\langle \hat{H}^2 \rangle$ come

$$\langle \hat{H}^2 \rangle = \frac{\int \Psi_T^* \hat{H}^2 \Psi_T d\mathbf{R}}{\int |\Psi_T|^2 d\mathbf{R}} = \frac{\int |\hat{H}\Psi_T|^2 d\mathbf{R}}{\int |\Psi_T|^2 d\mathbf{R}}, \quad (2.27)$$

l'Eq. (2.20) diventa

$$\sigma^2(\hat{H}) = \frac{\int |\hat{H}\Psi_T|^2 d\mathbf{R}}{\int |\Psi_T|^2 d\mathbf{R}} - E(\{\alpha_j\})^2 = \frac{\int \left[\left| \frac{\hat{H}\Psi_T}{\Psi_T} \right|^2 - E(\{\alpha_j\})^2 \right] |\Psi_T|^2 d\mathbf{R}}{\int |\Psi_T|^2 d\mathbf{R}} \quad (2.28)$$

e, per la (2.26), si ha

$$\sigma^2(\hat{H}) = \frac{\int \left[\frac{\hat{H}\Psi_T}{\Psi_T} - E(\{\alpha_j\}) \right]^2 |\Psi_T|^2 d\mathbf{R}}{\int |\Psi_T|^2 d\mathbf{R}}. \quad (2.29)$$

Scelto un insieme iniziale di parametri $\{\alpha_j^{(0)}\}$, sia $\Psi_T^{(0)} = \Psi_T(\mathbf{R}, \{\alpha_j^{(0)}\})$ e, moltiplicando gli integrandi al numeratore e al denominatore per $|\Psi_T^{(0)}|^2/|\Psi_T^{(0)}|^2$, otteniamo

$$\sigma^2(\hat{H}) = \frac{\int \left[\frac{\hat{H}\Psi_T}{\Psi_T} - E(\{\alpha_j\}) \right]^2 \left| \frac{\Psi_T}{\Psi_T^{(0)}} \right|^2 |\Psi_T^{(0)}|^2 d\mathbf{R}}{\int \left| \frac{\Psi_T}{\Psi_T^{(0)}} \right|^2 |\Psi_T^{(0)}|^2 d\mathbf{R}}. \quad (2.30)$$

Mediante un campionamento di importanza secondo $|\Psi_T^{(0)}|^2$ possiamo approssimare l'integrale (2.30) con una somma su N punti³

$$\sigma^2(\hat{H}) \simeq \frac{\sum_{i=1}^N \left[\frac{\hat{H}\Psi_T(\mathbf{R}_i, \{\alpha_j\})}{\Psi_T(\mathbf{R}_i, \{\alpha_j\})} - E(\{\alpha_j\}) \right]^2 \left| \frac{\Psi_T(\mathbf{R}_i, \{\alpha_j\})}{\Psi_T^{(0)}} \right|^2}{\sum_{i=1}^N \left| \frac{\Psi_T(\mathbf{R}_i, \{\alpha_j\})}{\Psi_T^{(0)}} \right|^2} \quad (2.31)$$

che, ponendo

$$w_i(\{\alpha_j\}) = \left| \frac{\Psi_T(\mathbf{R}_i, \{\alpha_j\})}{\Psi_T^{(0)}(\mathbf{R}_i)} \right|^2, \quad (2.32)$$

diventa

$$\sigma^2(\hat{H}) \simeq \frac{\sum_{i=1}^N \left[\frac{\hat{H}\Psi_T(\mathbf{R}_i, \{\alpha_j\})}{\Psi_T(\mathbf{R}_i, \{\alpha_j\})} - E(\{\alpha_j\}) \right]^2 w_i(\{\alpha_j\})}{\sum_{i=1}^N w_i(\{\alpha_j\})}. \quad (2.33)$$

Questa espressione è l'equivalente della (2.20) campionata con un numero finito di punti. Una sua minimizzazione diretta, però, implica ancora il calcolo di $E(\{\alpha_j\})$ mediante VMC ad ogni passo e dunque bisogna procedere in un modo alternativo. Vediamo come, illustrando passo per passo la procedura di ottimizzazione[69].

³Il numero ottimale N di configurazioni dipende dal numero dei parametri $\{\alpha_i\}$ e, come ordine di grandezza, è intorno a 1000.

1. Si parte dall'insieme iniziale di parametri $\{\alpha_j^{(0)}\}$, si esegue un VMC per determinare $E^{(0)} = E(\{\alpha_j^{(0)}\})$ e distribuire N configurazioni $\{\mathbf{R}_i\}$ (che rimangono fissate nei passi 2-4) secondo

$$|\Psi_T^{(0)}(\mathbf{R})|^2 = |\Psi_T(\mathbf{R}, \{\alpha_j^{(0)}\})|^2. \quad (2.34)$$

2. Si minimizza rispetto ai parametri $\{\alpha_j\}$ l'espressione

$$\sigma^2(\{\alpha_j\}) = \frac{\sum_{i=1}^N \left[\frac{\hat{H}\Psi_T(\mathbf{R}_i, \{\alpha_j\})}{\Psi_T(\mathbf{R}_i, \{\alpha_j\})} - E^{(0)} \right]^2 w_i(\{\alpha_j\})}{\sum_{i=1}^N w_i(\{\alpha_j\})} \quad (2.35)$$

che differisce dalla (2.33) per la presenza di $E^{(0)}$ al posto di $E(\{\alpha_j\})$, il cui calcolo, come detto, richiederebbe un VMC ad ogni passo della minimizzazione. In questo modo si ottiene un insieme di parametri $\{\alpha_j^{(1)}\}$ che rendono minima $\sigma^2(\{\alpha_j\})$.

3. L'insieme dei parametri ottenuti nel passo precedente viene utilizzato per calcolare l'energia $E^{(1)} = E(\{\alpha_j^{(1)}\})$ mediante un VMC. Si sostituisce $E^{(1)}$ a $E^{(0)}$ nell'Eq. (2.35) e, lasciando le N configurazioni sempre invariate (distribuite secondo $|\Psi_T^{(0)}|^2$), si procede a una nuova minimizzazione dell'Eq. (2.35) come nel passo 2.
4. La procedura appena delineata nel passo 3 si ripete fino al raggiungimento dell'autoconsistenza sull'energia: due o tre iterazioni di solito sono sufficienti.

Alla fine della procedura delineata, l'energia che compare nell'Eq. (2.35) è consistente con i parametri di minimo e l'Eq. (2.35) coincide con l'Eq. (2.33). Se i parametri ottenuti una volta raggiunta l'autoconsistenza per l'energia differiscono molto da quelli iniziali, si può ripetere tutta la sequenza iterativa 1-4 usando i parametri in uscita come nuova approssimazione iniziale e corrispondentemente eseguire un nuovo campionamento d'importanza. In questa maniera, è possibile raggiungere l'autoconsistenza anche sui parametri della funzione d'onda (entro la statistica, due o tre ripetizioni sono sufficienti) e, sottraendo i parametri in ingresso da quelli in uscita, stimare l'errore statistico su di essi causato dal numero finito N di configurazioni. Il metodo esposto è molto più conveniente della minimizzazione diretta

dell'Eq. (2.33) perché ci permette di procedere alla minimizzazione dei parametri (punto 2) senza dover eseguire per ogni nuovo insieme $\{\alpha_j\}$ un VMC per calcolare la corrispondente energia $E(\{\alpha_j\})$ da sostituire nell'Eq. (2.33), riducendo significativamente l'onere computazionale; infatti, il numero delle iterazioni autoconsistenti e il relativo calcolo VMC è molto inferiore al numero di passi necessari alla procedura di minimizzazione.

2.3 Monte Carlo di Diffusione

Mentre nel VMC si esegue il calcolo del valor medio di operatori hermitiani su una funzione d'onda di prova Ψ_T nota a priori (mediante il campionamento dello spazio secondo il suo modulo quadro) scopo del DMC è il calcolo di valori medi sullo stato fondamentale Ψ_0 dell'hamiltoniano considerato.

Sia Ψ_T una funzione d'onda di prova costruita in modo tale da approssimare Ψ_0 (supponiamo che sia Ψ_T che Ψ_0 siano reali, vedere nota 1) e $\hat{A} = \hat{A}^+$ un qualsiasi operatore hermitiano, e cioè tale che

$$\langle \Psi_0 | \hat{A} | \Psi_T \rangle = (\langle \Psi_T | \hat{A} | \Psi_0 \rangle)^*. \quad (2.36)$$

Se \hat{A} commuta con l'operatore \hat{H} (cioè $[\hat{A}, \hat{H}] = 0$), esso possiede le stesse autofunzioni⁴

$$\hat{A}\Psi_0(\mathbf{R}) = a_0\Psi_0(\mathbf{R}) \quad (2.37)$$

e si ha

$$\frac{\langle \Psi_0 | \hat{A} | \Psi_T \rangle}{\langle \Psi_0 | \Psi_T \rangle} = \frac{(\langle \Psi_T | \hat{A} | \Psi_0 \rangle)^*}{\langle \Psi_0 | \Psi_T \rangle} = a_0 \frac{(\langle \Psi_T | \Psi_0 \rangle)^*}{\langle \Psi_0 | \Psi_T \rangle} = a_0; \quad (2.38)$$

risulta dunque

$$\frac{\langle \Psi_0 | \hat{A} | \Psi_0 \rangle}{\langle \Psi_0 | \Psi_0 \rangle} = \frac{\langle \Psi_0 | \hat{A} | \Psi_T \rangle}{\langle \Psi_0 | \Psi_T \rangle}. \quad (2.39)$$

Se \hat{A} non commuta con \hat{H} (cioè $[\hat{A}, \hat{H}] \neq 0$) definendo

$$\delta\Psi_T \equiv \Psi_0 - \Psi_T \quad (2.40)$$

⁴Nel caso di autostati degeneri è sempre possibile scegliere autofunzioni comuni per i due operatori[28].

si può scrivere

$$\begin{aligned}
 \langle \Psi_0 | \hat{A} | \Psi_0 \rangle &= \langle \Psi_T + \delta \Psi_T | \hat{A} | \Psi_T + \delta \Psi_T \rangle = \\
 &= \langle \Psi_T | \hat{A} | \Psi_T \rangle + 2 \langle \delta \Psi_T | \hat{A} | \Psi_T \rangle + O(\delta \Psi_T^2) = \\
 &= 2 \langle \Psi_0 | \hat{A} | \Psi_T \rangle - \langle \Psi_T | \hat{A} | \Psi_T \rangle + O(\delta \Psi_T^2)
 \end{aligned} \tag{2.41}$$

in cui $\langle \Psi_T | \hat{A} | \Psi_T \rangle$ si calcola mediante VMC come descritto nella sezione precedente. Dall'Eq. (2.41) si può notare che quanto più Ψ_T si avvicina a Ψ_0 tanto più $\delta \Psi_T$ è piccola e l'approssimazione è buona⁵. Approssimare $\langle \Psi_0 | \hat{A} | \Psi_0 \rangle$ solamente con $\langle \Psi_0 | \hat{A} | \Psi_T \rangle$ e cioè

$$\langle \Psi_0 | \hat{A} | \Psi_0 \rangle = \langle \Psi_0 | \hat{A} | \Psi_T + \delta \Psi_T \rangle = \langle \Psi_0 | \hat{A} | \Psi_T \rangle + O(\delta \Psi_T) \tag{2.42}$$

da origine a un errore più grande come si vede confrontando direttamente le Eq. (2.41) e (2.42).

Osserviamo che in entrambi i casi, $[\hat{A}, \hat{H}] = 0$ e $[\hat{A}, \hat{H}] \neq 0$, il calcolo di $\langle \Psi_0 | \hat{A} | \Psi_0 \rangle$ si può ridurre al calcolo di $\langle \Psi_0 | \hat{A} | \Psi_T \rangle$ che ora consideriamo in dettaglio e per fissare le idee poniamo $\hat{A} = \hat{H}$. È semplice dimostrare che per passare dall'energia variazionale dell'Eq. (2.2) all'energia E_0 dello stato fondamentale è sufficiente sostituire $|\Psi_T|^2$ con $\Psi_T \Psi_0$. Infatti

$$E_0 = \frac{\langle \Psi_0 | \hat{H} | \Psi_T \rangle}{\langle \Psi_0 | \Psi_T \rangle} = \int \frac{\Psi_0(\mathbf{R}) \hat{H} \Psi_T(\mathbf{R}) d\mathbf{R}}{\Psi_0(\mathbf{R}) \Psi_T(\mathbf{R}) d\mathbf{R}} = \int \frac{\Psi_0(\mathbf{R}) \Psi_T(\mathbf{R}) \frac{\hat{H} \Psi_T(\mathbf{R})}{\Psi_T(\mathbf{R})} d\mathbf{R}}{\Psi_0(\mathbf{R}) \Psi_T(\mathbf{R}) d\mathbf{R}}. \tag{2.43}$$

Supposto di poter campionare secondo $\Psi_T \Psi_0$ al posto di $|\Psi_T|^2$ l'integrale (2.43) può essere approssimato dalla somma finita

$$E_0 \simeq \frac{1}{N} \sum_{i=1}^N \frac{\hat{H} \Psi_T(\mathbf{R}_i)}{\Psi_T(\mathbf{R}_i)}, \tag{2.44}$$

entro gli errori statistici. Poiché $\Psi_T \Psi_0$ non è noto, è necessario trovare un metodo che permetta almeno di campionarla mediante un numero finito N di punti $\{\mathbf{R}_i\}$. A meno che Ψ_0 (e dunque anche Ψ_T) sia privo di nodi (sistemi bosonici o sistemi a due elettroni nello stato di singoletto di spin), sorgono immediatamente dei problemi relativi all'interpretazione statistica del segno della funzione d'onda. Questi problemi, a parte il caso di poche particelle[72],

⁵Risulta possibile correggere questa approssimazione come descritto nel Rif.[71].

fino ad oggi non sono stati completamente risolti ed è necessario introdurre opportune approssimazioni. In particolare, nell'approssimazione più diffusa, quella cosiddetta a nodi fissi[4], lo spazio delle configurazioni del sistema fermionico viene diviso in varie regioni γ_i limitate dalle superfici nodali di Ψ_T e in cui il segno di Ψ_T non cambia. In ogni regione si considera l'equazione agli autovalori

$$\hat{H}\phi^{(i)} = E^{(i)}\phi^{(i)} \quad (2.45)$$

e si cerca lo stato fondamentale bosonico $\phi_0^{(i)}$ che è la soluzione priva di nodi all'interno di γ_i con la condizione al contorno che la funzione d'onda $\phi^{(i)}$ si annulli sulla superficie di γ_i . Estendendo le funzioni $\phi_0^{(i)}$ per antisimmetria otteniamo delle funzioni d'onda $\Phi_0^{(i)}$ definite in tutto lo spazio e con la corretta simmetria fermionica: per il principio variazionale, le corrispondenti energie $E_0^{(i)}$ sono dei limiti superiori per quella dello stato fermionico in tutto lo spazio, e sono la stessa nel caso in cui le posizioni dei nodi di Ψ_T siano esatte[4]. Fra tutte le soluzioni di stato fondamentale $\phi_0^{(i)}$ dell'Eq. (2.45) in ognuna delle regioni γ_i , quella corrispondente all'energia $E_0^{(i)}$ più bassa sarà quella che, estesa per antisimmetria, meglio approssima Ψ_0 . Nell'approssimazione a nodi fissi, dunque, lo spazio è suddiviso in tante regioni γ_i in cui il prodotto $\Psi_T\phi_0^{(i)}$ non cambia segno e perciò in ognuna di queste è possibile eseguire un campionamento secondo $\Psi_T\phi_0^{(i)}$ e calcolare l'energia relativa $E_0^{(i)}$.

Poiché alla fine ci si riduce sempre a uno stato fondamentale senza nodi (anche se con opportune condizioni al contorno), per capire il funzionamento del DMC trascuriamo le condizioni al contorno (che si possono sempre introdurre in seguito) e, per fissare le idee su un sistema che studieremo nel seguito, consideriamo il caso di due elettroni nello stato di singoletto di spin (spin non interagenti) e a simmetria sferica. La funzione d'onda di stato fondamentale di tale sistema si fattorizza nel prodotto di una parte spaziale simmetrica, senza nodi in tutto lo spazio, e una parte di spin antisimmetrica.

Sviluppando Ψ_T in serie delle autofunzioni $\{\Psi_k\}$ di \hat{H} (con autovalori E_k), si ha

$$\Psi_T = \sum_{k=0}^{+\infty} a_k \Psi_k, \quad (2.46)$$

e, se $\Psi(\mathbf{R}, t)$ è la soluzione dell'equazione di Schrödinger

$$\begin{cases} i \frac{\partial \Psi}{\partial t} = \hat{H} \Psi \\ \Psi(\mathbf{R}, 0) = \Psi_T(\mathbf{R}) \end{cases} \quad , \quad (2.47)$$

risulta

$$\Psi(\mathbf{R}, t) = \sum_{k=0}^{+\infty} a_k \Psi_k(\mathbf{R}) e^{-i E_k t}. \quad (2.48)$$

Se introduciamo la funzione $\Phi(\mathbf{R}, t) \equiv \Psi(\mathbf{R}, -it) e^{E_T t}$, che risolve l'equazione

$$\begin{cases} \frac{\partial \Phi}{\partial t} = -(\hat{H} - E_T) \Phi \\ \Phi(\mathbf{R}, 0) = \Psi_T(\mathbf{R}) \end{cases} \quad , \quad (2.49)$$

per $t \gg (E_1 - E_0)^{-1}$ è possibile isolare Ψ_0 :

$$\Phi(\mathbf{R}, t) = a_0 \Psi_0(\mathbf{R}) e^{(E_T - E_0)t} \quad t \gg (E_1 - E_0)^{-1}. \quad (2.50)$$

E_T , detta energia di prova, permette di evitare problemi numerici, per tempi lunghi, dovuti alla presenza del fattore esponenziale. La funzione f definita da

$$f(\mathbf{R}, t) = \Psi_T(\mathbf{R}) \Phi(\mathbf{R}, t), \quad (2.51)$$

nel limite di $t \gg (E_1 - E_0)^{-1}$ si può scrivere come

$$f(\mathbf{R}, t) = a_0 \Psi_T(\mathbf{R}) \Psi_0(\mathbf{R}) e^{(E_T - E_0)t} \quad t \gg (E_1 - E_0)^{-1}, \quad (2.52)$$

e perciò, per tempi sufficientemente lunghi, è proporzionale al prodotto $\Psi_T \Psi_0$: campionare secondo la f per tempi lunghi equivale a campionare secondo $\Psi_T \Psi_0$ come desiderato. Inoltre dalle Eq. (2.51) e (2.49) segue che

$$f(\mathbf{R}, 0) = \Psi_T(\mathbf{R}) \Phi(\mathbf{R}, 0) = |\Psi_T(\mathbf{R})|^2, \quad (2.53)$$

e quindi campionare secondo $f(\mathbf{R}, 0)$ equivale a campionare secondo $|\Psi_T(\mathbf{R})|^2$ (VMC). Introducendo la funzione di Green per l'Eq. (2.49)

$$\tilde{G}(\mathbf{R}, \mathbf{R}', t) = \langle \mathbf{R} | \exp[-t(\hat{H} - E_T)] | \mathbf{R}' \rangle, \quad (2.54)$$

la funzione $\Phi(\mathbf{R}, t)$ si può calcolare a partire da $\Phi(\mathbf{R}, 0)$

$$\Phi(\mathbf{R}, t) = \int \tilde{G}(\mathbf{R}, \mathbf{R}', t) \Phi(\mathbf{R}', 0) d\mathbf{R}'. \quad (2.55)$$

La funzione di Green \tilde{G} è la soluzione dell'equazione

$$-(\hat{H} - E_T) \tilde{G}(\mathbf{R}, \mathbf{R}', t) = \frac{\partial \tilde{G}(\mathbf{R}, \mathbf{R}', t)}{\partial t}, \quad (2.56)$$

con la condizione iniziale

$$\tilde{G}(\mathbf{R}, \mathbf{R}', 0) = \delta(\mathbf{R} - \mathbf{R}'). \quad (2.57)$$

Purtroppo la conoscenza completa di \tilde{G} è equivalente alla soluzione dell'equazione di Schrödinger agli autovalori e quindi fuori portata⁶. Dall'Eq. (2.55) segue che

$$\Psi_T(\mathbf{R}) \Phi(\mathbf{R}, t) = \int \Psi_T(\mathbf{R}) \tilde{G}(\mathbf{R}, \mathbf{R}', t) \Psi_T^{-1}(\mathbf{R}') \Psi_T(\mathbf{R}') \Phi(\mathbf{R}', 0) d\mathbf{R}' \quad (2.59)$$

e usando l'Eq.(2.51) si ha

$$f(\mathbf{R}, t) = \int G(\mathbf{R}, \mathbf{R}', t) f(\mathbf{R}', 0) d\mathbf{R}', \quad (2.60)$$

in cui $G(\mathbf{R}, \mathbf{R}', t)$ è la funzione di Green relativa alla funzione f ed è legata alla $\tilde{G}(\mathbf{R}, \mathbf{R}', t)$ dalla relazione

$$G(\mathbf{R}, \mathbf{R}', t) = \Psi_T(\mathbf{R}) \tilde{G}(\mathbf{R}, \mathbf{R}', t) \Psi_T^{-1}(\mathbf{R}'). \quad (2.61)$$

Per campionare secondo f è necessario considerare f come una densità e $G(\mathbf{R}, \mathbf{R}', t)$ come probabilità (generalizzata) di transizione da \mathbf{R} a \mathbf{R}' nell'intervallo di tempo t : questa interpretazione richiede che f e la funzione di Green G siano definite positive. Come \tilde{G} anche G è sconosciuta, ma è possibile costruire un'espressione G_0 approssimata per intervalli temporali

⁶ \tilde{G} si può scrivere esplicitamente come

$$\tilde{G}(\mathbf{R}, \mathbf{R}', t) = \sum_n \Psi_n(\mathbf{R}) \Psi_n^*(\mathbf{R}') \exp[-t(E_n - E_T)] \quad (2.58)$$

dove $\{\Psi_n\}$ sono le autofunzioni e $\{E_n\}$ gli autovalori di \hat{H} .

Δt piccoli con un errore di ordine $O(\Delta t^2)$ (approssimazione a tempi corti): in questo modo l'Eq. (2.60) per $t \gg (E_1 - E_0)^{-1}$ può essere approssimata iterando M volte l'equazione

$$f(\mathbf{R}, n\Delta t) = \int G_0(\mathbf{R}, \mathbf{R}', \Delta t) f(\mathbf{R}', (n-1)\Delta t) d\mathbf{R}' \quad n = 1, \dots, M, \quad (2.62)$$

dove $M\Delta t \gg (E_1 - E_0)^{-1}$. Approssimare G con G_0 significa approssimare \hat{H} con l'hamiltoniano corrispondente a G_0 [73] e dunque commettere un errore nel calcolo dell'energia E_0 tramite l'Eq. (2.44): il risultato esatto, nel limite $\Delta t \rightarrow 0$, si ottiene ripetendo il calcolo per valori diversi di Δt ed estrapolando a $\Delta t = 0$ (vedere Sez. 2.4). Per costruire G_0 è necessario avere un'espressione esplicita per l'operatore \hat{H} , che nel nostro caso è del tipo

$$\hat{H} = -\frac{1}{2}\Delta + U(\mathbf{R}), \quad (2.63)$$

con $U(\mathbf{R})$ potenziale moltiplicativo. Così si può scrivere l'Eq. (2.49) in maniera più esplicita come

$$\frac{\partial \Phi(\mathbf{R}, t)}{\partial t} = \left[\frac{1}{2}\Delta - U(\mathbf{R}) + E_T \right] \Phi(\mathbf{R}, t) \quad (2.64)$$

e, moltiplicando primo e secondo membro per $\Psi_T(\mathbf{R})$, si ha

$$\Psi_T(\mathbf{R}) \frac{\partial \Phi(\mathbf{R}, t)}{\partial t} = \Psi_T(\mathbf{R}) \left[\frac{1}{2}\Delta - U(\mathbf{R}) + E_T \right] \Phi(\mathbf{R}, t). \quad (2.65)$$

Siccome $\Psi_T(\mathbf{R})$ non dipende da t e $U(\mathbf{R})$ è moltiplicativo si può scrivere

$$\begin{aligned} \frac{\partial [\Psi_T(\mathbf{R})\Phi(\mathbf{R}, t)]}{\partial t} &= \frac{1}{2}\Psi_T(\mathbf{R})\Delta\Phi(\mathbf{R}, t) - U(\mathbf{R})[\Psi_T(\mathbf{R})\Phi(\mathbf{R}, t)] \\ &\quad + E_T\Psi_T(\mathbf{R})\Phi(\mathbf{R}, t) \end{aligned} \quad (2.66)$$

o, tralasciando gli argomenti delle funzioni Ψ_T e Φ per semplicità di notazione,

$$\frac{\partial (\Psi_T\Phi)}{\partial t} = \frac{1}{2}\Psi_T\Delta\Phi - U(\mathbf{R})(\Psi_T\Phi) + E_T\Psi_T\Phi. \quad (2.67)$$

Esprimendo $\Psi_T\Delta\Phi$ nel modo seguente

$$\begin{aligned} \Psi_T\Delta\Phi &= \Delta(\Psi_T\Phi) - \Phi\Delta\Psi_T - 2\nabla\Phi\nabla\Psi_T \\ &= \Delta(\Psi_T\Phi) + \Phi\Delta\Psi_T - 2\nabla(\Phi\nabla\Psi_T) \\ &= \Delta(\Psi_T\Phi) + (\Psi_T\Phi)\frac{\Delta\Psi_T}{\Psi_T} - \nabla\left(\Psi_T\Phi\frac{\nabla\Psi_T}{\Psi_T}\right), \end{aligned} \quad (2.68)$$

l'Eq. (2.67) diventa

$$\begin{aligned} \frac{\partial(\Psi_T \Phi)}{\partial t} = & \frac{1}{2} \Delta(\Psi_T \Phi) + \frac{1}{2} (\Psi_T \Phi) \frac{\Delta \Psi_T}{\Psi_T} - \frac{1}{2} \nabla \left(\Psi_T \Phi \frac{\nabla \Psi_T}{\Psi_T} \right) \\ & - U(\mathbf{R})(\Psi_T \Phi) + E_T \Psi_T \Phi. \end{aligned} \quad (2.69)$$

Definita la “forza quantica”

$$\mathbf{F}^Q(\mathbf{R}) \equiv 2 \frac{\nabla \Psi_T(\mathbf{R})}{\Psi_T(\mathbf{R})}, \quad (2.70)$$

e l’“energia locale”

$$E_L(\mathbf{R}) \equiv \frac{\hat{H} \Psi_T(\mathbf{R})}{\Psi_T(\mathbf{R})}, \quad (2.71)$$

l'Eq. (2.69) diventa

$$\frac{\partial f}{\partial t} = \frac{1}{2} \Delta f - \frac{1}{2} \nabla [f \mathbf{F}^Q(\mathbf{R})] + [E_T - E_L(\mathbf{R})] f, \quad (2.72)$$

che è l'equazione di diffusione (generalizzata) per f . Per ricavare la funzione di Green $G(\mathbf{R}, \mathbf{R}', t)$ relativa all'Eq. (2.72) nell'approssimazione a tempi corti scriviamo l'Eq. (2.72) nella forma

$$\frac{\partial f(\mathbf{R}, t)}{\partial t} = (\hat{T} + \hat{U}) f(\mathbf{R}, t), \quad (2.73)$$

dove $\hat{T} = \hat{T}_2 + \hat{T}_1$ e gli operatori \hat{T}_2 , \hat{T}_1 , \hat{U} , detti rispettivamente di diffusione, di deriva o *drift* e di nascita-morte o *branching*, sono definiti come

$$\hat{T}_2 = \frac{1}{2} \Delta, \quad (2.74)$$

$$\hat{T}_1 = -\frac{1}{2} \nabla \mathbf{F}^Q(\mathbf{R}), \quad (2.75)$$

$$\hat{U} = E_T - E_L(\mathbf{R}). \quad (2.76)$$

Dati due operatori qualsiasi \hat{A} e \hat{B} che non commutano, cioè $[\hat{A}, \hat{B}] \neq 0$, la formula di Baker-Hausdorff[74] ci permette di scrivere

$$\exp(\hat{A}) \exp(\hat{B}) = \exp\left\{ \hat{A} + \hat{B} + \frac{1}{2} [\hat{A}, \hat{B}] + \frac{1}{12} [(\hat{A} - \hat{B}), [\hat{A}, \hat{B}]] + \dots \right\}, \quad (2.77)$$

e nel nostro caso si ha

$$\begin{aligned}\exp[t(\hat{T} + \hat{U})] &= \exp\left[t\left(\frac{\hat{U}}{2} + \hat{T} + \frac{\hat{U}}{2}\right)\right] = \\ &= \exp\left(\frac{1}{2}t\hat{U}\right) \exp(t\hat{T}) \exp\left(\frac{1}{2}t\hat{U}\right) + O(t^3).\end{aligned}\quad (2.78)$$

La funzione di Green per l'equazione di diffusione (2.73) è definita da

$$G(\mathbf{R}, \mathbf{R}', t) = \langle \mathbf{R} | \exp[t(\hat{T} + \hat{U})] | \mathbf{R}' \rangle, \quad (2.79)$$

e, tramite l'Eq. (2.78) e l'identità

$$\int |\mathbf{R}\rangle \langle \mathbf{R}| d\mathbf{R} = \mathbf{1}, \quad (2.80)$$

trascurando i termini del terzo ordine in t , arriviamo all'espressione valida per tempi corti

$$\begin{aligned}G_0(\mathbf{R}, \mathbf{R}', t) &= \iint \langle \mathbf{R} | \exp\left(\frac{t}{2}\hat{U}\right) | \mathbf{R}'' \rangle \langle \mathbf{R}'' | \exp(t\hat{T}) | \mathbf{R}''' \rangle \\ &\quad \times \langle \mathbf{R}''' | \exp\left(\frac{t}{2}\hat{U}\right) | \mathbf{R}' \rangle d\mathbf{R}'' d\mathbf{R}''',\end{aligned}\quad (2.81)$$

in cui compaiono le funzioni di Green di $\hat{U}/2$ e \hat{T} . Il calcolo di $\langle \mathbf{R} | \exp\left(\frac{t\hat{U}}{2}\right) | \mathbf{R}' \rangle$ è immediato:

$$\begin{aligned}\langle \mathbf{R} | \exp\left(\frac{t\hat{U}}{2}\right) | \mathbf{R}' \rangle &= \exp\left\{\frac{t}{2}[E_T - E_L(\mathbf{R})]\right\} \langle \mathbf{R} | \mathbf{R}' \rangle = \\ &= \exp\left\{\frac{t}{2}[E_T - E_L(\mathbf{R})]\right\} \delta(\mathbf{R} - \mathbf{R}').\end{aligned}\quad (2.82)$$

Sostituendo nell'Eq. (2.81) si ottiene:

$$\begin{aligned}G_0(\mathbf{R}, \mathbf{R}', t) &= \iint \exp\left\{\frac{t}{2}[E_T - E_L(\mathbf{R})]\right\} \delta(\mathbf{R} - \mathbf{R}'') \langle \mathbf{R}'' | \exp(t\hat{T}) | \mathbf{R}''' \rangle \\ &\quad \times \exp\left\{\frac{t}{2}[E_T - E_L(\mathbf{R}')] \right\} \delta(\mathbf{R}''' - \mathbf{R}') d\mathbf{R}'' d\mathbf{R}''' \\ &= \langle \mathbf{R} | \exp(t\hat{T}) | \mathbf{R}' \rangle \exp\left\{-t\left[\frac{E_L(\mathbf{R}') + E_L(\mathbf{R})}{2} - E_T\right]\right\}.\end{aligned}\quad (2.83)$$

L'elemento di matrice $\langle \mathbf{R} | \exp(t\hat{T}) | \mathbf{R}' \rangle$ è la funzione di Green per l'operatore \hat{T} , che nel limite di piccoli t , per l'Eq. (2.77), si può scrivere nel seguente modo

$$\begin{aligned}\langle \mathbf{R} | \exp(t\hat{T}) | \mathbf{R}' \rangle &= \langle \mathbf{R} | \exp(t\hat{T}_2) \exp(t\hat{T}_1) | \mathbf{R}' \rangle + O(t^2) = \\ &= \int \langle \mathbf{R} | \exp(t\hat{T}_2) | \mathbf{R}'' \rangle \langle \mathbf{R}'' | \exp(t\hat{T}_1) | \mathbf{R}' \rangle d\mathbf{R}'' + O(t^2).\end{aligned}\quad (2.84)$$

Il primo fattore dell'integrando della (2.84) è la funzione di Green per l'operatore di diffusione \hat{T}_2 o, in altri termini, per la particella libera in $3N_P$ dimensioni e, come noto, è una gaussiana:

$$\langle \mathbf{R} | \exp\left(\frac{t}{2}\Delta\right) | \mathbf{R}'' \rangle = \frac{1}{(2\pi t)^{3N_P/2}} \exp\left[-\frac{(\mathbf{R} - \mathbf{R}'')^2}{2t}\right]. \quad (2.85)$$

Il secondo fattore dell'integrando della (2.84) è la funzione di Green per l'operatore di deriva \hat{T}_1 che, come si può verificare direttamente sostituendo nell'equazione per la funzione di Green, risulta essere

$$\langle \mathbf{R}'' | \exp(t\hat{T}_1) | \mathbf{R}' \rangle = \delta\{\mathbf{R}'' - \mathbf{S}(t)\}, \quad (2.86)$$

con $\mathbf{S}(t)$ soluzione dell'equazione

$$\frac{d\mathbf{S}}{dt} = \frac{1}{2}\mathbf{F}^Q[\mathbf{S}(t)], \quad (2.87)$$

che soddisfa la condizione iniziale $\mathbf{S}(0) = \mathbf{R}'$. Un'espressione per piccoli t , corretta a meno di termini del secondo ordine, si ottiene usando l'espansione intorno a $t = 0$

$$\mathbf{S}(t) = \mathbf{R}' + \frac{1}{2}\mathbf{F}^Q(\mathbf{R}')t + O(t^2), \quad (2.88)$$

da cui segue che, per tempi corti,

$$\langle \mathbf{R}'' | \exp(t\hat{T}_1) | \mathbf{R}' \rangle = \delta\left\{\mathbf{R}'' - \mathbf{R}' - \frac{1}{2}\mathbf{F}^Q(\mathbf{R}')t\right\}. \quad (2.89)$$

Riunendo i risultati delle Eq. (2.83, 2.84, 2.85, 2.89) si ottiene un'espressione esplicita per la funzione di Green nell'approssimazione a tempi corti, corretta a meno di termini del secondo ordine in t :

$$\begin{aligned} G_0(\mathbf{R}, \mathbf{R}', t) &= \frac{1}{(2\pi t)^{\frac{3N_P}{2}}} \exp\left\{-\frac{\left[\mathbf{R} - \mathbf{R}' - \frac{1}{2}\mathbf{F}^Q(\mathbf{R}')t\right]^2}{2t}\right\} \\ &\times \exp\left\{-t\left[\frac{E_L(\mathbf{R}) + E_L(\mathbf{R}')}{2} - E_T\right]\right\} \geq 0. \end{aligned} \quad (2.90)$$

L'Eq. (2.72) descrive un processo di diffusione, deriva e nascita-morte che, per tempi piccoli, come si vede dalla forma della funzione di Green G_0 , possono essere considerati separatamente. È abbastanza naturale, allora, data una popolazione di configurazioni che campiona

$f(\mathbf{R}, (n-1)\Delta t)$, farla evolvere secondo tali processi in modo tale da soddisfare stocasticamente l'Eq. (2.62) e quindi campionare $f(\mathbf{R}, n\Delta t)$. Fatto questo, a partire dal campionamento di $|\Psi_T|^2$ per $t = 0$ (che si fa mediante VMC), siamo in grado di campionare $\Psi_T\Psi_0$ per $t \gg (E_1 - E_0)^{-1}$, iterando M volte l'Eq. (2.62), e quindi di stimare E_0 con l'Eq. (2.44).

2.3.1 Funzione di prova

La scelta di Ψ_T , a priori arbitraria, ha delle importanti conseguenze sull'efficienza e sulla velocità di convergenza del DMC. L'efficienza è massima quando è minimo il numero N di configurazioni necessarie a campionare $E_L(\mathbf{R})$, Eq. (2.44), per ottenere un'incertezza statistica inferiore alla soglia fissata. Nel limite in cui $\Psi_T = \Psi_0$, $E_L(\mathbf{R})$ è costante e uguale a E_0 (l'incertezza statistica è nulla), ed è sufficiente un'unica configurazione. In maniera simile a questo caso ideale, Ψ_T deve essere scelta in modo tale da rendere $E_L(\mathbf{R})$ la più vicina possibile a una costante, se si vuole abbattere l'incertezza statistica e massimizzare l'efficienza. Nel caso di singolarità per l'hamiltoniano \hat{H} (singolarità elettrone-nucleo o elettrone-elettrone), è necessario che Ψ_T soddisfi le condizioni di cuspidi[75, 69, 4, 70] e cioè deve comportarsi come la funzione d'onda esatta in modo tale che $E_L(\mathbf{R})$ non diverga in nessun punto⁷. Matematicamente, la condizione di cuspidi per ciò che riguarda l'interazione fra due elettroni di indici i e j si esprime come

$$\left. \frac{1}{\Psi_T} \frac{\partial \Psi_T}{\partial r_{ij}} \right|_{r_{ij}=0} = \begin{cases} \frac{1}{4} & i, j \text{ spin paralleli} \\ \frac{1}{2} & i, j \text{ spin antiparalleli} \end{cases} \quad (2.91)$$

dove r_{ij} è la distanza elettrone-elettrone; quella relativa all'interazione fra l'elettrone di indice i con il nucleo di carica Z (per il potenziale coulombiano $-Z/r_i$) come

$$\left. \frac{1}{\Psi_T} \frac{\partial \Psi_T}{\partial r_i} \right|_{r_i=0} = -Z, \quad (2.92)$$

in cui r_i è la distanza elettrone-nucleo. Osserviamo che nel caso di uno pseudopotenziale non esiste il problema della singolarità elettrone-nucleo. Ψ_T è anche all'origine della forza

⁷Osserviamo che se $E_L(\mathbf{R})$ diverge in un punto, l'espansione Eq. (2.90) per la funzione di Green non è corretta nell'intorno di tale punto.

quantica \mathbf{F}^Q e anche in questo caso può servire a migliorare l'efficienza, se è ragionevolmente simile a Ψ_0 . Infatti, l'effetto di \mathbf{F}^Q è quello di spostare le configurazioni verso le regioni in cui Ψ_T è massima. Se Ψ_T approssima ragionevolmente Ψ_0 , le regioni di massimo delle due funzioni coincidono e la regione in cui il prodotto $\Psi_T\Psi_0$ contribuisce in modo rilevante all'integrale (2.43) è ben localizzata e il campionamento efficiente.

Per quanto riguarda la scelta della forma esplicita da dare a Ψ_T nel caso generale rimandiamo alla letteratura[4, 69, 70], mentre nel caso del sistema da noi studiato si veda la Sez. 2.6.1.

2.3.2 Bilancio dettagliato

Torniamo alle Eq. (2.72) e (2.90) e consideriamo una popolazione di N configurazioni $\{\mathbf{R}_i\}$: nell'intervallo temporale Δt , a causa del processo di deriva, la popolazione evolve in un insieme $\{\mathbf{R}'_i\}$ tale che [Eq. (2.89)]

$$\mathbf{R}_i \rightarrow \mathbf{R}'_i = \mathbf{R}_i + \frac{1}{2}\mathbf{F}^Q(\mathbf{R}_i)\Delta t. \quad (2.93)$$

La diffusione introduce una dinamica stocastica con legge [Eq. (2.85)]

$$\mathbf{R}'_i \rightarrow \mathbf{R}''_i = \mathbf{R}'_i + \boldsymbol{\omega}\sqrt{\Delta t}, \quad (2.94)$$

dove $\boldsymbol{\omega}$ è un vettore stocastico le cui componenti hanno distribuzione normale standard[43]. Come nel VMC, l'evoluzione secondo diffusione e deriva viene subordinata alla seguente probabilità di transizione [che non dipende da $E_L(\mathbf{R})$ e cioè dal processo di nascita-morte]⁸

$$P(\mathbf{R} \rightarrow \mathbf{R}'') = \min \left\{ \frac{|\Psi_T(\mathbf{R}'')|^2 G_0(\mathbf{R}, \mathbf{R}'', \Delta t)}{|\Psi_T(\mathbf{R})|^2 G_0(\mathbf{R}'', \mathbf{R}, \Delta t)}, 1 \right\}, \quad (2.95)$$

e dunque vale l'equazione del bilancio dettagliato

$$\frac{P(\mathbf{R}'' \rightarrow \mathbf{R})}{P(\mathbf{R} \rightarrow \mathbf{R}'')} = \frac{|\Psi_T(\mathbf{R})|^2}{|\Psi_T(\mathbf{R}'')|^2} \frac{K(\mathbf{R} \rightarrow \mathbf{R}'')}{K(\mathbf{R}'' \rightarrow \mathbf{R})}, \quad (2.96)$$

⁸Diffusione e deriva (virtuali) e selezione delle configurazioni virtuali mediante la probabilità di transizione P , possono essere fatte in due modi diversi. Infatti, si può considerare un'intera configurazione per volta (e cioè tutti gli elettroni che ne fanno parte contemporaneamente) come nell'algoritmo di Umrigar[70], oppure un solo elettrone per volta, in successione, per tutti gli elettroni della configurazione come nell'algoritmo di Reynolds[4].

che assicura il campionamento asintotico di $|\Psi_T|^2$ e in cui K è il nucleo stocastico della dinamica data dai processi di deriva [Eq. (2.89)] e di diffusione [Eq. (2.85)]

$$K(\mathbf{R} \rightarrow \mathbf{R}'') = \frac{1}{(2\pi\Delta t)^{\frac{3N_P}{2}}} \exp \left\{ -\frac{\left[\mathbf{R}'' - \mathbf{R} - \frac{1}{2}\mathbf{F}^Q(\mathbf{R})\Delta t \right]^2}{2\Delta t} \right\}. \quad (2.97)$$

A questo punto, la popolazione iniziale $\{\mathbf{R}_i\}$ è diventata $\{\mathbf{R}_i^f\}$, dove

$$\mathbf{R}_i^f = \begin{cases} \mathbf{R}_i'' & \text{con probabilità } P(\mathbf{R}_i \rightarrow \mathbf{R}_i'') \\ \mathbf{R}_i & \text{con probabilità } 1 - P(\mathbf{R}_i \rightarrow \mathbf{R}_i'') \end{cases} \quad (2.98)$$

Osserviamo che il bilancio dettagliato non altera la distribuzione di equilibrio per i processi di deriva e diffusione ma piuttosto corregge l'errore introdotto da G_0 : l'Eq. (2.72) priva del termine di nascita-morte (ultimo termine a destra nel secondo membro) ha come soluzione stazionaria $f(\mathbf{R}, t) = |\Psi_T(\mathbf{R})|^2$ come si può verificare per sostituzione diretta. Inoltre esso garantisce, nel limite $\Psi_T \rightarrow \Psi_0$, il corretto campionamento di $|\Psi_0|^2$ qualunque sia il valore di Δt (pur avendo G approssimata da G_0), e perciò permette di ridurre l'errore dovuto a Δt finito (che scompare nel limite $\Delta t \rightarrow 0$).

Fino a questo punto il numero di configurazioni nella popolazione è rimasto inalterato, dato che diffusione e deriva da sole hanno una funzione di Green normalizzata a 1. Se ci fermassimo a questo punto, però, avremmo solamente un metodo alternativo al VMC per il campionamento di $|\Psi_T|^2$. Quello che fa la differenza è il processo di nascita-morte che, diversamente da diffusione e deriva, non influisce direttamente sulla dinamica delle configurazioni, bensì ne altera densità e numero (la funzione di Green relativa, Eq. (2.82), non è normalizzata a 1). La realizzazione del processo di nascita-morte, però, è diversa nei due algoritmi studiati e quindi li consideriamo separatamente. Cominciamo con quello di Reynolds *et al.*[4] nella prossima sezione.

2.3.3 L'algoritmo di Reynolds

Il processo di nascita-morte agisce introducendo dei pesi $\{w_i\}$ associati alle configurazioni $\{\mathbf{R}_i^f\}$

$$w_i = \exp \left\{ -\Delta t_{eff} \left[\frac{E_L(\mathbf{R}_i) + E_L(\mathbf{R}_i^f)}{2} - E_T \right] \right\} \quad (2.99)$$

Al posto di Δt è stato introdotto Δt_{eff} per tener conto del fatto che diffusione e deriva della popolazione sono ridotti dalla probabilità di transizione. Infatti, se $\langle(\Delta \mathbf{r})^2\rangle = 3\Delta t$ è la distanza quadratica media che un elettrone percorre durante il processo di diffusione nel tempo Δt , la distanza media realmente percorsa dall'intera popolazione è inferiore, dato che non tutte le configurazioni virtuali vengono alla fine accettate. Si può quindi definire Δt_{eff} per mezzo della relazione

$$\langle(\Delta \mathbf{r})^2\rangle_{accettate} = 3\Delta t_{eff}, \quad (2.100)$$

dove $\langle \rangle_{accettate}$ è la media presa su tutte le configurazioni elettroniche accettate, oppure, dato che $\langle(\Delta \mathbf{r})^2\rangle_{virtuali} \simeq 3\Delta t$ (entro la statistica), come

$$\Delta t_{eff} = \Delta t \frac{\langle(\Delta \mathbf{r})^2\rangle_{accettate}}{\langle(\Delta \mathbf{r})^2\rangle_{virtuali}}. \quad (2.101)$$

I pesi vengono fatti agire sulla popolazione replicando o sopprimendo configurazioni esistenti nel modo seguente. Se $0 \leq w_i < 1$, la configurazione i -esima viene soppressa con probabilità $(1 - w_i)$ (o equivalentemente mantenuta inalterata con probabilità w_i). Se $w_i \geq 1$, invece, detta \bar{w}_i la sua parte intera, la configurazione viene sostituita da \bar{w}_i sue repliche con probabilità $1 - (w_i - \bar{w}_i)$ e da $\bar{w}_i + 1$ repliche con probabilità $(w_i - \bar{w}_i)$. Le configurazioni sopprese vengono definitivamente eliminate dalla popolazione, mentre le repliche vengono integrate nella stessa come nuove configurazioni, l'una indipendente dall'altra. In questo modo, se la popolazione iniziale campionava $f(\mathbf{R}, (n-1)\Delta t)$, la nuova popolazione campegnerà $f(\mathbf{R}, n\Delta t)$ e avrà un numero di individui che può essere diverso da quella iniziale. Per evitare l'estinzione o l'esplosione della popolazione abbiamo a disposizione l'energia di prova E_T . Per mantenere costante il numero medio di individui, è necessario che i pesi w_i definiti dall'Eq. (2.99) fluttuino in media attorno a 1 e cioè deve risultare che il valor medio dell'argomento dell'esponenziale su tutte le configurazioni a tutti i passi sia approssimativamente uguale a zero. All'iterazione $(n+1)$ -esima si deve dunque avere

$$\frac{1}{n+1} \sum_{k=0}^n \frac{1}{N(k)} \sum_{i=1}^{N(k)} \left[\frac{E_L[\mathbf{R}_i^{(k)}] + E_L[\mathbf{R}_i^{(k)f}]}{2} - E_T(n+1) \right] \simeq 0 \quad (2.102)$$

dove $\{\mathbf{R}_i^{(k)}\}$ è l'insieme finale di $N(k)$ individui alla k -esima iterazione e cioè quello iniziale alla $(k+1)$ -esima che, dopo deriva, diffusione e transizione, evolve in $\{\mathbf{R}_i^{(k)f}\}$. Approssimando il valor medio di $E_L(\mathbf{R}_i^{(k)f})$ con quello di $E_L(\mathbf{R}_i^{(k)})$ il valore di E_T che ci permette di mantenere in media costante il numero di individui della popolazione (almeno per un certo numero di iterazioni) risulta essere

$$E_T(n+1) = \frac{1}{n+1} \sum_{k=0}^n \frac{1}{N(k)} \sum_{i=1}^{N(k)} E_L[\mathbf{R}_i^{(k)}] \quad n = 0, \dots, M, \quad (2.103)$$

e cioè la media cumulativa dell'energia locale. Osserviamo che, all'equilibrio, $E_T \simeq E_0$. Nonostante l'utilizzo accurato di E_T risulta necessario rinormalizzare periodicamente la popolazione a un numero prefissato di configurazioni [quello iniziale $N(0)$], replicando o eliminando a caso (con probabilità uniforme) dalla lista delle configurazioni esistenti il numero necessario di individui⁹.

2.3.4 L'algoritmo di Umrigar

Nel paragrafo precedente abbiamo visto come il processo di nascita-morte altera densità e normalizzazione della popolazione agendo direttamente sulla stessa e replicando o sopprimendo configurazioni esistenti. Un'alternativa a questo modo di procedere è quella di associare un peso w_i a ogni configurazione \mathbf{R}_i , che evolva con essa e che tenga conto delle variazioni di densità e normalizzazione introdotte dal processo di nascita-morte (alle configurazioni della popolazione iniziale è assegnato il peso 1). La popolazione $\{\mathbf{R}_i\}$, invece, è soggetta solo a diffusione e deriva subordinate alla probabilità di transizione P e dunque campiona $|\Psi_T|^2$ e non più f . È necessario, allora, modificare l'Eq. (2.44) per l'energia in modo da includere l'effetto dei pesi

$$E_0 \simeq \frac{\sum_{i=1}^N w_i E_L(\mathbf{R}_i^f)}{\sum_{i=1}^N w_i}, \quad (2.104)$$

dato che, nella procedura descritta, ogni configurazione assume un'importanza proporzionale al suo peso. È dunque la presenza dei pesi che permette di passare da una stima variazionale dell'energia [Eq. (2.104) con tutti i pesi uguali a 1] a quella di stato fondamentale. Il

⁹Tale operazione va eseguita con la frequenza più bassa possibile, per evitare di introdurre un errore sistematico.

vantaggio è una riduzione delle fluttuazioni statistiche rispetto a quelle che si hanno utilizzando l'algoritmo di Reynolds, in quanto il numero di individui della popolazione rimane inalterato. I pesi $\{w_i^f\}$ associati a $\{\mathbf{R}_i^f\}$ sono determinati dal processo di nascita-morte come

$$w_i^f = w_i \exp \left\{ -\Delta t_{eff} \left[\frac{E_L(\mathbf{R}_i) + E_L(\mathbf{R}_i^f)}{2} - E_T \right] \right\}. \quad (2.105)$$

Δt_{eff} è calcolato in maniera autoconsistente; partendo da $n = 0$ si effettuano M_1 iterazioni dell'Eq. (2.62) con $\Delta t_{eff} = \Delta t$ e alla fine si calcola

$$\Delta t'_{eff} = \Delta t \frac{\langle P(\Delta \mathbf{R})^2 \rangle}{\langle (\Delta \mathbf{R})^2 \rangle}, \quad (2.106)$$

dove il valor medio $\langle \rangle$ è preso rispetto a tutte le configurazioni virtuali per tutte le M_1 iterazioni, $\Delta \mathbf{R}$ è la distanza percorsa nel tempo Δt da una configurazione durante la diffusione e P è la sua probabilità di transizione, Eq. (2.95). Quindi si prosegue per altre M_1 iterazioni ponendo $\Delta t_{eff} = \Delta t'_{eff}$ e si calcola nuovamente $\Delta t'_{eff}$ secondo l'Eq. (2.106). Questa procedura si ripete finché Δt_{eff} e $\Delta t'_{eff}$ sono uguali (entro la statistica). M dovrà perciò essere tale da garantire sia il raggiungimento dell'autoconsistenza su Δt_{eff} ($M = JM_1$ con $J \simeq 2-3$), che la condizione $M\Delta t \gg (E_1 - E_0)^{-1}$. Osserviamo che nell'Eq. (2.106) il valor medio su tutti i passi virtuali pesato con la probabilità di transizione è equivalente al valor medio sui soli passi accettati (vedere algoritmo di Reynolds) ma è soggetto a una minor fluttuazione statistica. E_T viene scelto in modo tale da mantenere approssimativamente costante e pari a W_0 il peso totale W della popolazione. Poiché l'energia locale E_L all'iterazione $(n+1)$ -esima fluttua approssimativamente attorno alla sua media cumulativa $E_{cum}(n)$

$$E_{cum}(n) = \frac{\sum_{k=0}^n \sum_{i=1}^{N(k)} w_i^{(k)} E_L[\mathbf{R}_i^{(k)}]}{\sum_{k=0}^n W(k)}, \quad (2.107)$$

dove

$$W(n) = \sum_{i=1}^{N(n)} w_i^{(n)}, \quad (2.108)$$

l'argomento n indica l'iterazione a cui si riferiscono le varie grandezze, $\{\mathbf{R}_i^{(k)}\}$ è la popolazione alla fine della k -esima iterazione e $\{w_i^{(k)}\}$ sono i pesi relativi, in media risulta

$$W(n+1) \simeq W(n) \exp\{[E_T - E_{cum}(n)]\Delta t_{eff}\} \quad (2.109)$$

$$\ln \left[\frac{W(n+1)}{W(n)} \right] \simeq [E_T - E_{cum}(n)]\Delta t_{eff} \quad (2.110)$$

$$E_T \simeq E_{cum}(n) - \frac{1}{\Delta t_{eff}} \ln \left[\frac{W(n)}{W(n+1)} \right] \quad (2.111)$$

e siccome vogliamo che $W(n+1) \simeq W_0 = N(0)$ all'iterazione $(n+1)$ -esima poniamo

$$E_T(n+1) = E_{cum}(n) - \frac{1}{\Delta t_{eff}} \ln \left[\frac{W(n)}{W_0} \right] \quad n = 0, \dots, M, \quad (2.112)$$

Per evitare problemi numerici con i singoli pesi, non appena uno dei w_i supera una soglia prefissata w_{max} , si crea una copia della corrispondente configurazione \mathbf{R}_i e si assegna il peso $w_i/2$ a ciascuna delle due (identiche) configurazioni risultanti (operazione di *split*). Al contrario, per ogni coppia di configurazioni $\mathbf{R}_i, \mathbf{R}_j$ i cui pesi w_i, w_j sono inferiori a un certo w_{min} , si seleziona una delle due configurazioni

$$\begin{cases} \mathbf{R}_i & \text{con probabilità } \frac{w_i}{w_i + w_j} \\ \mathbf{R}_j & \text{con probabilità } \frac{w_j}{w_i + w_j} \end{cases}, \quad (2.113)$$

e le si assegna il peso $(w_i + w_j)$, mentre l'altra viene soppressa (operazione di *join*).

Le Eq. (2.104, 2.105, 2.107) possono essere sostituite da equazioni equivalenti, nel limite di $\Delta t \rightarrow 0$, che però sono affette da una minor incertezza statistica e per le quali l'errore dovuto a Δt finito è più piccolo[70]. Detti $p_i = P(\mathbf{R}_i \rightarrow \mathbf{R}_i'')$ e $q_i = 1 - p_i$, invece di considerare la configurazione finale \mathbf{R}_i^f , utilizziamo direttamente le configurazioni \mathbf{R}_i e \mathbf{R}_i'' , che diventano \mathbf{R}_i^f con probabilità q_i e p_i rispettivamente, e sostituiamo $E_L(\mathbf{R}_i^f)$ con $q_i E_L(\mathbf{R}_i) + p_i E_L(\mathbf{R}_i'')$. L'Eq. (2.105) diventa

$$w_i^f = w_i \exp \left(-\Delta t_{eff} \left\{ \frac{p_i}{2} [E_L(\mathbf{R}_i) + E_L(\mathbf{R}_i'')] + q_i E_L(\mathbf{R}_i) - E_T \right\} \right). \quad (2.114)$$

In modo simile l'Eq. (2.104) si trasforma in

$$E_0 = \frac{\sum_{i=1}^N w_i [q_i E_L(\mathbf{R}_i) + p_i E_L(\mathbf{R}_i'')]}{\sum_{i=1}^N w_i}, \quad (2.115)$$

e così anche l'Eq. (2.107)

$$E_{cum}(n) = \frac{\sum_{k=0}^n \sum_{i=1}^{N(k)} w_i^{(k)} \{q_i E_L[\mathbf{R}_i^{(k-1)}] + p_i E_L[\mathbf{R}_i^{(k-1)''}]\}}{\sum_{k=0}^n W(k)}. \quad (2.116)$$

2.4 Calcolo dei valori medi

Sia nel VMC che nel DMC, una volta che la popolazione delle configurazioni campiona le rispettive distribuzioni di equilibrio o asintotiche, è possibile procedere con il calcolo delle energie variazionale o di stato fondamentale tramite le Eq. (2.3,2.44). Se l'incertezza massima richiesta sulle quantità da calcolare è fissata a priori, è necessario utilizzare la media cumulativa dell'energia locale fatta su \bar{M} iterazioni successive della popolazione. Partendo da un insieme di configurazioni che campiona la distribuzione di equilibrio, alla fine di ogni iterazione si ottiene una nuova popolazione che ancora campiona la stessa distribuzione. Sia $\{\mathbf{R}_i^{(k)}\}$ la popolazione alla fine della k -esima iterazione, $N(k)$ il numero degli individui e $\{w_i^{(k)}\}$ gli eventuali pesi associati [con peso totale $W(k)$]. La media cumulativa nell'algoritmo di Reynolds per il DMC, è data da

$$\bar{E}_0 = \frac{1}{\bar{M}} \sum_{k=M+1}^{M+\bar{M}} \frac{1}{N(k)} \sum_{i=1}^{N(k)} E_L[\mathbf{R}_i^{(k)}], \quad (2.117)$$

(questo, e quanto detto nella prossima sezione, vale anche per il VMC con le sostituzioni $E_0 \rightarrow E$, $N(k) \rightarrow N$ e $\Delta t \rightarrow \lambda$) che, introducendo la media di configurazione alla fine dell'iterazione k -esima

$$E_C(k) = \frac{1}{N(k)} \sum_{i=1}^{N(k)} E_L[\mathbf{R}_i^{(k)}] \quad k = 1, \dots, M + \bar{M}, \quad (2.118)$$

si può scrivere anche come

$$\bar{E}_0 = \frac{1}{\bar{M}} \sum_{k=M+1}^{M+\bar{M}} E_C(k). \quad (2.119)$$

Dato che l'errore statistico è proporzionale a $1/\sqrt{\bar{M}}$, è possibile, pur di procedere sufficientemente a lungo, scendere sotto la soglia di incertezza richiesta. Dall'Eq. (2.103) per E_T , se $n > M$, viene tolto il contributo delle iterazioni di stabilizzazione e abbiamo

$$E_T(n+1) = \frac{1}{(n-M)} \sum_{k=M+1}^n \frac{1}{N(k)} \sum_{i=1}^{N(k)} E_L[\mathbf{R}_i^{(k)}], \quad (2.120)$$

con $M + 1 \leq n \leq M + \bar{M}$.

Quanto detto finora vale anche per l'algoritmo di Umrigar, pur di modificare conseguentemente le equazioni, ma contemporaneamente bisogna affrontare un problema che nel caso precedente risulta trascurabile. Anche se ogni generazione di configurazioni evolve da quella precedente secondo l'Eq. (2.62), l'influenza del peso totale della popolazione $W(n)$ nel calcolo di $E_T(n + 1)$, la quale a sua volta controlla $W(n + 1)$, produce un errore sistematico nel calcolo di E_0 e di E_{cum} , Eq. (2.115, 2.116), che è proporzionale a $1/W_0$ [76, 70]. Oltre a scegliere popolazioni in cui il numero medio di individui W_0 sia relativamente grande¹⁰, è possibile ridurre l'incidenza di tale errore entro l'incertezza statistica. Questo può esser fatto pesando in maniera diversa la popolazione delle configurazioni ad ogni iterazione [76, 70].

L'Eq. (2.115) va generalizzata come segue per il calcolo della media cumulativa

$$\bar{E}_0 = \frac{\sum_{k=M+1}^{M+\bar{M}} \Pi_k \sum_{i=1}^{N(k)} w_i^{(k)} \left\{ q_i E_L[\mathbf{R}_i^{(k-1)}] + p_i E_L[\mathbf{R}_i^{(k-1)''}] \right\}}{\sum_{k=M+1}^{M+\bar{M}} \Pi_k W(k)}, \quad (2.121)$$

dove

$$\Pi_k = \prod_{m=0}^{\min(k,L)-1} \exp[-\Delta t_{eff} E_T(k-m)] \quad k, L \geq 1, \quad (2.122)$$

$\Pi_0 = 1 \forall L$ e $\Pi_k = 1 \forall k$ se $L = 0$. L è un parametro importante del metodo di correzione e viene scelto una volta per tutte all'inizio. In linea di principio, l'errore sistematico viene completamente rimosso nel limite in cui $L = k$ e $\bar{M} \rightarrow +\infty$, ma in questo caso anche l'errore statistico diverge con \bar{M} . È dunque necessario limitare L a una costante in modo da ridurre l'errore sistematico il più possibile, senza per questo aumentare l'errore statistico. Valori ragionevoli di L sono dell'ordine del tempo di correlazione introdotto nella prossima sezione [70].

L'Eq. (2.107) per E_{cum} merita un discorso a parte dato che dobbiamo distinguere il caso delle M iterazioni di stabilizzazione della popolazione dalle \bar{M} iterazioni per il calcolo di \bar{E}_0 . Durante le prime M iterazioni si ha

$$E_{cum}(n) = \frac{\sum_{k=0}^n \Pi_k \sum_{i=1}^{N(k)} w_i^{(k)} \left\{ q_i E_L[\mathbf{R}_i^{(k-1)}] + p_i E_L[\mathbf{R}_i^{(k-1)''}] \right\}}{\sum_{k=0}^n \Pi_k W(k)}, \quad (2.123)$$

¹⁰Osserviamo che è possibile eliminare tale errore ripetendo i calcoli per diversi valori di W_0 ed estrapolando a $W_0^{-1} = 0$, ma questo risulta molto oneroso.

con $0 \leq n \leq M$, mentre nel seguito il contributo della fase di stabilizzazione viene trascurato

$$E_{cum}(n) = \frac{\sum_{k=M+1}^n \Pi_k \sum_{i=1}^{N(k)} w_i^{(k)} \{q_i E_L[\mathbf{R}_i^{(k-1)}] + p_i E_L[\mathbf{R}_i^{(k-1)''}]\}}{\sum_{k=M+1}^n \Pi_k W(k)}, \quad (2.124)$$

con $M+1 \leq n \leq M+\bar{M}$.

Per quanto riguarda Δt_{eff} , mentre nel metodo di Reynolds esso viene continuamente aggiustato mediante l'Eq. (2.101), in quello di Umrigar Δt_{eff} viene determinato durante le prime M iterazioni [Eq. (2.106)] e poi tenuto fisso.

Un'ultima considerazione riguarda l'estrapolazione per $\Delta t = 0$. Una volta ripetuto il calcolo di \bar{E}_0 e della relativa incertezza statistica, per differenti valori di Δt , si esegue un'interpolazione dei risultati ottenuti mediante polinomi in potenze intere di $\sqrt{\Delta t}$ a partire da Δt (le stime dell'errore dovuto all'approssimazione a tempi corti indicano un andamento di questo tipo[70, 73]), e si estrapolano risultato e incertezza a $\Delta t = 0$.

2.5 Correlazione e calcolo delle incertezze

Poiché le configurazioni evolvono l'una nell'altra in piccoli intervalli temporali Δt (diversamente non riusciremmo ad approssimare bene la funzione di Green G mediante G_0) e dunque la posizione di ogni configurazione dipende in larga misura dalla posizione della configurazione precedente, i valori di E_L e di conseguenza quelli di E_C sono correlati, e a configurazioni successive corrispondono energie molto simili (le stesse considerazioni valgono anche nel caso del VMC, pur di fare le opportune sostituzioni come spiegato nella sezione precedente). Per quanto detto, l'uso della varianza dell'energia locale (per semplicità di notazione consideriamo solo l'algoritmo di Reynolds)

$$\sigma^2 = \frac{1}{\bar{M}-1} \sum_{k=M+1}^{M+\bar{M}} [E_C(k) - \bar{E}_0]^2 \quad (2.125)$$

per calcolare l'errore statistico su \bar{E}_0 , per mezzo della formula $\sigma/\sqrt{\bar{M}}$, dà luogo a una sottostima in quanto tale formula è valida solo nel caso di quantità indipendenti. Per superare questo inconveniente, si divide l'intervallo temporale che si estende da $(M+1)\Delta t$ a $(M+\bar{M})\Delta t$ in B blocchi di M_B passi ciascuno, essendo M_B un numero intero. Per ognuno

di questi si calcola la media di E_C , detta energia di blocco

$$E_B(j) = \frac{1}{M_B} \sum_{k=1}^{M_B} E_C[M + (j-1)M_B + k] \quad j = 1, \dots, B, \quad (2.126)$$

dove $E_C[M + (j-1)M_B + k]$ è l'energia media di tutta la popolazione alla fine della k -esima iterazione nel j -esimo blocco. Più è grande M_B , più è grande l'intervallo temporale $M_B \Delta t$ che separa ogni blocco e più indipendenti saranno tra loro le $E_B(j)$. Viceversa, a parità di passi totali di media, più è piccolo M_B , più grande sarà il numero di blocchi B e più significativa la statistica. Per quanto detto, a noi interessa trovare il valore minimo di M_B per cui valori successivi di $E_B(j)$ risultano non correlati in modo da avere il più gran numero di blocchi possibile. Per fare questo si calcola la varianza dell'energia di blocco

$$\sigma_B^2 = \frac{1}{B-1} \sum_{j=1}^B [E_B(j) - \bar{E}_0]^2, \quad (2.127)$$

e il tempo di correlazione

$$T_{corr} = M_B \left(\frac{\sigma_B}{\sigma} \right)^2. \quad (2.128)$$

Per poter stimare correttamente T_{corr} dall'Eq. (2.128) i blocchi non devono essere correlati e cioè deve risultare $M_B \gg T_{corr}$. Infatti, a partire da $M_B = 1$ si ha

$$\sigma_B^2 = \sigma^2 \quad \Rightarrow \quad T_{corr} = 1 \quad (M_B = 1), \quad (2.129)$$

ma all'aumentare di M_B la correlazione fra i blocchi è minore di quella fra i singoli passi e dunque T_{corr} nell'Eq. (2.128) aumenta

$$\sigma_B^2 > \frac{\sigma^2}{M_B} \quad \Rightarrow \quad T_{corr} > 1 \quad (M_B > 1). \quad (2.130)$$

T_{corr} si stabilizza e cioè resta costante entro le fluttuazioni statistiche una volta raggiunta l'indipendenza fra i blocchi: ciò avviene quando la lunghezza di ogni blocco è sostanzialmente maggiore del tempo di correlazione, ovvero per $M_B \gg T_{corr}$. Il valore di M_B attorno a cui T_{corr} si stabilizza, è il numero minimo di passi che deve avere ogni blocco perché non si abbia correlazione fra le $E_B(j)$. Il calcolo di $E_B(j)$ e di σ_B^2 ha anche lo scopo di verificare se

lo stato di equilibrio è stato effettivamente raggiunto. Una deviazione sistematica dal valor medio indicherebbe una situazione ancora in fase di stabilizzazione e non di equilibrio.

Un metodo alternativo che ci permette di stimare quanto deve essere grande M_B perché i blocchi non siano correlati è il calcolo dell'autocorrelazione lineare delle energie di blocco

$$C_L(M_B) = \frac{\frac{1}{B-1} \sum_{i=j}^{B-1} [E_B(j) - \bar{E}_0][E_B(j+1) - \bar{E}_0]}{\frac{1}{B-1} \sum_{j=1}^B [E_B(j) - \bar{E}_0]^2}, \quad (2.131)$$

che dipende da M_B tramite l'Eq. (2.126). I valori di M_B per cui l'autocorrelazione lineare risulta molto minore di 1 indicano indipendenza fra i vari blocchi.

Una volta raggiunta l'indipendenza statistica fra blocchi successivi, la nostra stima di E_0 è

$$E_0 = \bar{E}_0 \pm \frac{\sigma_B}{\sqrt{B}}. \quad (2.132)$$

2.6 Risultati per sistemi a due elettroni

Come anticipato nella Sez. 2.3, il sistema che abbiamo studiato utilizzando le tecniche Monte Carlo esposte è costituito da due elettroni in un campo esterno descritto da uno pseudopotenziale a norma conservata[20] nello stato di singoletto di spin e a simmetria sferica. Con gli pseudopotenziali abbiamo a disposizione degli hamiltoniani efficaci in cui gli unici gradi di libertà sono quelli di valenza, e quindi, in linea di principio, particolarmente interessanti per i motivi esposti nella Sez. 2.1. L'ostacolo principale all'uso degli pseudopotenziali a norma conservata nel DMC (e per certi versi anche nel VMC), come discusso nel primo capitolo, è il fatto che si tratta di operatori non locali e perciò non utilizzabili senza ulteriori elaborazioni (VMC e DMC) e approssimazioni (DMC). Nonostante questo, Fahy, Wang e Louie[5] e Mitáš *et al.*[7], rispettivamente nel caso del VMC e del DMC, hanno dimostrato che tale strada è percorribile.

Originariamente ideati e costruiti all'interno di una teoria di campo medio a singola particella, la Teoria del Funzionale Densità nell'approssimazione di densità locale per la de-

scrizione dell'interazione elettrone-elettrone[23], è naturale chiedersi se sia legittimo portare gli pseudopotenziali in un ambito in cui l'interazione elettrone-elettrone viene trattata esattamente (QMC). Ecco quindi che anche lo studio del più semplice sistema possibile in cui è presente l'interazione elettrone-elettrone, quello a due elettroni, può dare informazioni preziose che aiutino a chiarire il problema della portabilità, in particolar modo se portato avanti in maniera sistematica per buona parte della Tavola Periodica. Poiché all'interno della Teoria del Funzionale Densità, nella descrizione del sistema di sola valenza da noi considerato, si fa uso della sola componente $\ell = 0$ dello pseudopotenziale a norma conservata per la descrizione dell'interazione core-valenza, anche nel nostro hamiltoniano a molti corpi i due elettroni di valenza sono immersi nello stesso potenziale esterno. Di conseguenza il potenziale agisce in maniera moltiplicativa esattamente come nei metodi QMC presentati in precedenza. Inoltre, non è necessario introdurre l'approssimazione a nodi fissi in quanto lo stato fondamentale del nostro sistema è privo di nodi spaziali.

Tutti i risultati presentati in questo capitolo, a meno che non vi siano ulteriori specificazioni, sono espressi in unità atomiche e sono relativi al VMC e al DMC con un numero iniziale di configurazioni $N = 400$, con un numero di passi di equilibratura $M = 10^4$ (VMC e DMC nella versione di Reynolds) e $M = 2 \times 10^4$ (DMC nella versione di Umrigar), con un numero di passi di media $\bar{M} = 10^5$ divisi in $B = 500$ blocchi di $M_B = 200$ passi ciascuno. Gli altri parametri usati per il DMC sono $w_{min} = 0.5$, $w_{max} = 2.0$, $L = 200$ e $J = 2$ (algoritmo di Umrigar).

2.6.1 Hamiltoniano e funzione di prova

Come anticipato, l'hamiltoniano \hat{H} per il nostro sistema è del tipo

$$\hat{H} = -\frac{1}{2} \sum_{i=1}^2 \Delta_i + \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} + \sum_{i=1}^2 v(r_i), \quad (2.133)$$

dove $v(r)$ è la componente $\ell = 0$ dello pseudopotenziale a norma conservata per l'atomo considerato[40].

La funzione di prova Ψ_T è stata scelta della forma[4, 70]

$$\Psi_T(\mathbf{r}_1, \mathbf{r}_2) = \frac{1}{4\pi} \exp\left(\frac{ar_{12}}{1 + br_{12}}\right) \frac{\chi(r_1)}{r_1} \frac{\chi(r_2)}{r_2} \quad (2.134)$$

dove $\mathbf{r}_1, \mathbf{r}_2$ sono le coordinate dei due elettroni, $r_1 = |\mathbf{r}_1|$, $r_2 = |\mathbf{r}_2|$ e $r_{12} = |\mathbf{r}_1 - \mathbf{r}_2|$ è la distanza relativa. $\chi(r)$ è la parte radiale della funzione d'onda di stato fondamentale per un elettrone nel potenziale esterno considerato (in assenza dell'altro elettrone). Il termine esponenziale, detto fattore di Jastrow, è stato introdotto per tener conto del fatto che i due elettroni sono interagenti. Dei due parametri presenti nel fattore di Jastrow, come vedremo, a viene fissato dalle condizioni di cuspidità per $r_{12} = 0$ (singolarità elettrone-elettrone) mentre b è un parametro variazionale che andrà ottimizzato. La simmetria complessiva di Ψ_T è sferica in accordo con quella del sistema che intendiamo studiare. Pur essendo molto semplice, questa scelta di Ψ_T risulta abbastanza efficace come testimoniano i risultati per l'energia variazionale ottenuti con il VMC (prossima sezione). La condizione di cuspidità nel punto di singolarità elettrone-elettrone (l'unica, dato che $v(r)$ è finito ovunque), è data dalla relazione (vedere Sez. 2.3.1)

$$\frac{1}{\Psi_T} \frac{\partial \Psi_T}{\partial r_{12}} \Big|_{r_{12}=0} = \frac{1}{2} \quad (2.135)$$

ed è facile verificare che è soddisfatta pur di prendere $a = 1/2$; in questo modo l'energia locale è finita ovunque.

2.6.2 Tempo di correlazione e autocorrelazione.

Tramite il tempo di correlazione T_{corr} e la funzione di autocorrelazione lineare C_L (Sez. 2.5) possiamo stimare sia nel caso del VMC che del DMC il numero minimo M_B di passi che deve contenere ogni blocco perché le energie di blocco siano indipendenti fra loro e l'errore statistico risulti calcolato correttamente tramite la formula (2.132). In particolare questo è il valore per cui T_{corr} si stabilizza e C_L è molto minore di 1. Come sappiamo la dinamica stocastica del VMC dipende dall'ampiezza massima λ del passo in ogni direzione (Sez. 2.2) e quella del DMC dall'intervallo temporale Δt (Sez. 2.3). λ nel VMC ha un ruolo analogo a Δt nel DMC in quanto sia λ che Δt determinano l'ampiezza dello spostamento. Nel

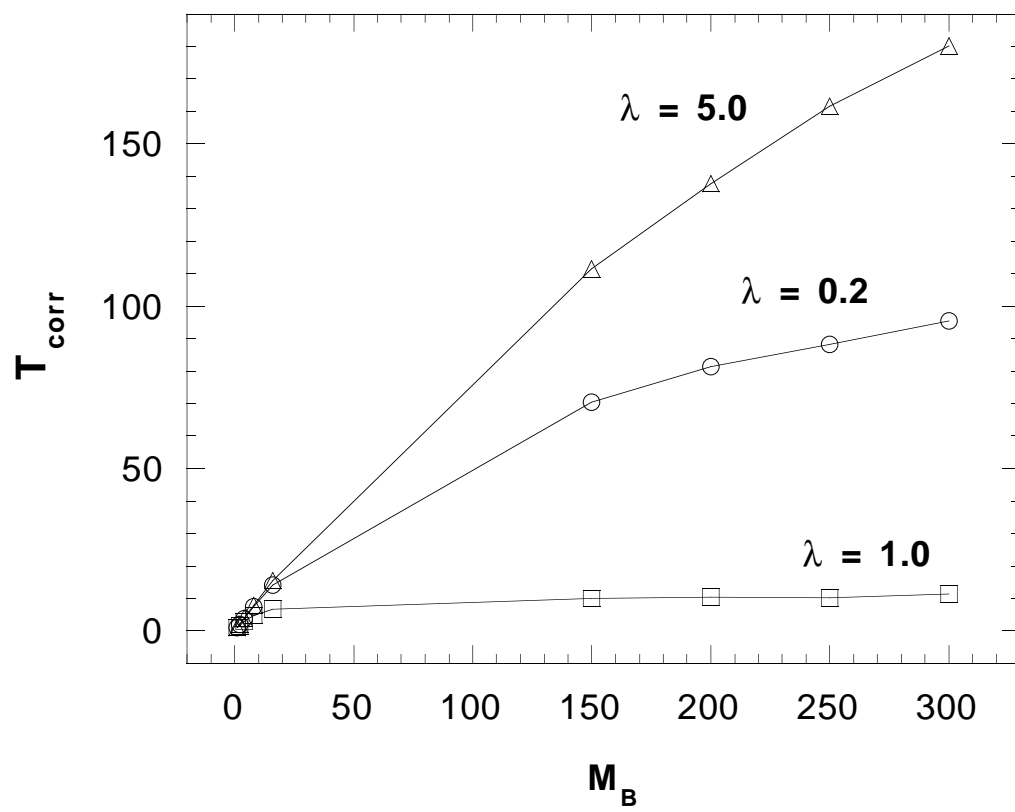


Figura 2.1: Valori di T_{corr} in funzione di M_B per $\lambda = 0.2$ (\circ), per $\lambda = 1.0$ (\square) e $\lambda = 5.0$ (\triangle) nel caso dello ione Si^{2+} per una simulazione VMC (vedere testo).

caso del VMC potremmo pensare che quanto più grande è λ , tanto più piccolo sarà il valore di M_B necessario a raggiungere l'indipendenza fra le energie di blocco in quanto la regione di spazio accessibile in un passo della dinamica aumenta con λ . In realtà non è così, perché se λ è troppo grande la dinamica sposta buona parte delle configurazioni fuori dalla regione d'interesse, la probabilità di transizione rifiuta quasi sempre il movimento e il valore dell'energia rimane invariato. Viceversa, se λ è troppo piccolo, anche se il passo viene quasi sempre accettato le energie fra i passi successivi sono molto simili fra loro. Perciò, in entrambi i casi M_B deve essere molto grande perché si raggiunga l'indipendenza. Il valore di λ è arbitrario e abbiamo visto con quale criterio sia possibile scegliere quello più opportuno; qui vogliamo studiare quale influenza ha la scelta di λ sulla correlazione e in Fig. 2.1 riportiamo T_{corr} in funzione di M_B per i valori $\lambda = 0.2$, $\lambda = 1.0$ (che è quello ottimale dato dalla procedura della Sez. 2.2) e $\lambda = 5.0$ nel caso dello ione Si^{2+} . Come previsto, il valore migliore di λ , e cioè quello per il quale l'indipendenza si raggiunge più velocemente in quanto T_{corr} si stabilizza prima¹¹, è proprio quello intermedio $\lambda = 1.0$.

Il caso del DMC è simile, ma la scelta di Δt non è arbitraria come quella di λ dato che l'approssimazione della funzione di Green G tramite G_0 è valida soltanto a tempi corti e abbiamo bisogno di usare Δt piccoli e diversi per correggere l'errore dovuto a tale approssimazione. In Fig. 2.2 sono mostrati i risultati ottenuti per T_{corr} e C_L in funzione di M_B per $\Delta t = 0.05$ nel caso dello ione Cl^{5+} ; osserviamo che T_{corr} si stabilizza intorno a quel valore di M_B per il quale C_L diventa molto minore di 1. Questo dimostra che il tempo di correlazione e l'autocorrelazione lineare forniscono informazioni equivalenti sulla correlazione, e nel caso del Cl^{5+} il valore minimo di M_B per cui si ha indipendenza tra i blocchi è circa 100. Un andamento qualitativamente simile si è ottenuto anche nel caso del VMC. È da osservare che in generale il valore minimo di M_B tale che non si abbia correlazione dipende, oltre che da λ o Δt , dall'elemento chimico studiato (e quindi dal suo pseudopotenziale) e dalla funzione di prova Ψ_T tramite l'energia locale E_L . In realtà, come abbiamo scritto nell'introduzione,

¹¹È possibile ridurre in maniera rilevante i tempi di correlazione modificando l'algoritmo di Metropolis come discusso da Umrigar[77].

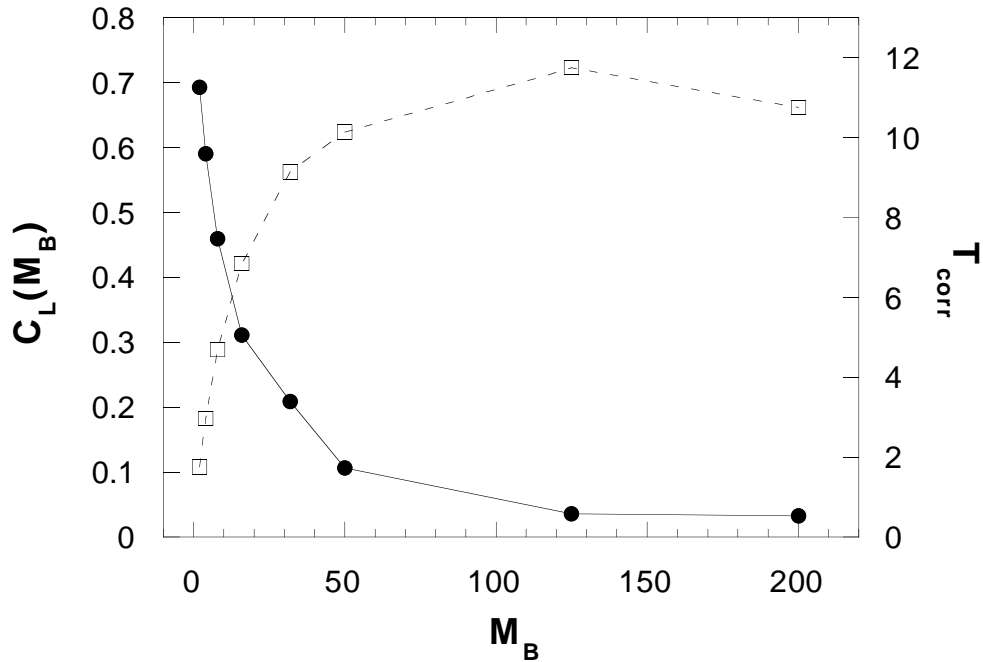


Figura 2.2: Valori di T_{corr} (□) e di C_L (●) in funzione di diversi valori di M_B nel caso dello ione Cl^{5+} con $\Delta t = 0.05$ per una simulazione DMC.

abbiamo scelto un valore di M_B che andasse bene per tutti gli elementi chimici studiati e per tutti i Δt ($M_B = 200$), in modo da avere la stessa statistica ($B = 500$) in tutti i casi.

2.6.3 Risultati per il VMC

Mediante la tecnica di minimizzazione della varianza (Sez. 2.2.1) abbiamo ottimizzato il parametro b nella funzione di prova Ψ_T e con un VMC standard abbiamo calcolato la corrispondente energia variazionale E . Per quanto riguarda b abbiamo eseguito un unico calcolo autoconsistente sull'energia (3 iterazioni). Una volta determinato b , con un calcolo più lungo abbiamo determinato l'energia variazionale associata ($N = 400$, $M = 10^4$, $\bar{M} = 10^5$, $M_B = 200$, $B = 500$). I risultati ottenuti sono riassunti nelle Tab. (2.1-2.8) e sono espressi in unità atomiche[11]. Il fine di queste simulazioni è quello di costruire un'approssimazione ragionevole per lo stato fondamentale Ψ_0 e, avendo a che fare con un sistema semplice per il quale l'approssimazione a nodi fissi non è necessaria, una funzione di prova a un solo parametro è perfettamente adeguata. I numeri in parentesi sono gli errori statistici e si

atomo	Z	b (Bohr ⁻¹)	\bar{E} (Ha)
He	2	0.26375	-2.87942(20)
Be	4	0.1575	-1.003829(19)
Mg	12	0.155	-0.840111(15)
Ca	20	0.125	-0.662179(13)
Sr	38	0.13	-0.616583(13)
Ba	56	0.125	-0.558502(14)

Tavola 2.1: Energie variazionali per gli atomi neutri (He e gruppo IIA). Z è il numero atomico, b è il parametro variazionale nel fattore di Jastrow, \bar{E} l'energia variazionale (vedere testo).

ione	Z	b (Bohr ⁻¹)	\bar{E} (Ha)
H ⁻	1	0.1625	-0.48856(11)
Li ⁻	3	0.095	-0.217657(15)
Na ⁻	11	0.105	-0.211612(15)
K ⁻	19	0.08625	-0.1801188(91)
Rb ⁻	37	0.095	-0.1738555(87)
Cs ⁻	55	0.0875	-0.1614594(80)

Tavola 2.2: Energie variazionali per gli ioni negativi (gruppo IA). Z è il numero atomico, b è il parametro variazionale nel fattore di Jastrow, \bar{E} l'energia variazionale (vedere testo).

ione	Z	b (Bohr ⁻¹)	\bar{E} (Ha)
B ⁺	5	0.21375	-2.297189(25)
Al ⁺	13	0.195	-1.741988(26)
Ga ⁺	31	0.22	-1.914496(31)
In ⁺	49	0.255	-1.728965(40)
Tl ⁺	81	0.375	-1.830987(87)

Tavola 2.3: Energie variazionali per gli ioni positivi (gruppo IIIB). Z è il numero atomico, b è il parametro variazionale nel fattore di Jastrow, \bar{E} l'energia variazionale (vedere testo).

riferiscono alle ultime cifre: per esempio $-2.886342(31) = -2.886342 \pm 0.000031$.

Come si vede, sono stati considerati quasi tutti gli atomi e ioni a due elettroni di valenza il cui stato fondamentale ha simmetria sferica. Sono invece stati esclusi dallo studio gli elementi di transizione e le terre rare.

ione	Z	b (Bohr ⁻¹)	\bar{E} (Ha)
C ²⁺	6	0.27625	-4.090335(34)
Si ²⁺	14	0.2325	-2.886342(31)
Ge ²⁺	32	0.245	-2.961067(33)
Sn ²⁺	50	0.230	-2.622140(38)
Pb ²⁺	82	0.475	-2.702624(85)

Tavola 2.4: Energie variazionali per gli ioni positivi (gruppo IVB). Z è il numero atomico, b è il parametro variazionale nel fattore di Jastrow, \bar{E} l'energia variazionale (vedere testo).

ione	Z	b (Bohr ⁻¹)	\bar{E} (Ha)
N ³⁺	7	0.33875	-6.384061(46)
P ³⁺	15	0.28875	-4.263398(37)
As ³⁺	33	0.295	-4.158949(44)
Sb ³⁺	51	0.2675	-3.623054(45)
Bi ³⁺	83	0.575	-3.67865(13)

Tavola 2.5: Energie variazionali per gli ioni positivi (gruppo VB). Z è il numero atomico, b è il parametro variazionale nel fattore di Jastrow, \bar{E} l'energia variazionale (vedere testo).

ione	Z	b (Bohr ⁻¹)	\bar{E} (Ha)
O ⁴⁺	8	0.3825	-9.176360(50)
S ⁴⁺	16	0.345	-5.869896(47)
Se ⁴⁺	34	0.28875	-5.504630(46)
Te ⁴⁺	52	0.28	-4.727198(49)
Po ⁴⁺	84	0.725	-4.72655(15)

Tavola 2.6: Energie variazionali per gli ioni positivi (gruppo VIB). Z è il numero atomico, b è il parametro variazionale nel fattore di Jastrow, \bar{E} l'energia variazionale (vedere testo).

ione	Z	b (Bohr ⁻¹)	\bar{E} (Ha)
F ⁵⁺	9	0.445	-12.471017(57)
Cl ⁵⁺	17	0.35125	-7.719827(57)
Br ⁵⁺	35	0.35125	-6.990580(52)
I ⁵⁺	53	0.2925	-5.933266(53)
At ⁵⁺	85	0.750	-5.87197(10)

Tavola 2.7: Energie variazionali per gli ioni positivi (gruppo VIIB). Z è il numero atomico, b è il parametro variazionale nel fattore di Jastrow, \bar{E} l'energia variazionale (vedere testo).

ione	Z	b (Bohr ⁻¹)	\bar{E} (Ha)
Ne ⁶⁺	10	0.47	-16.266114(66)
Ar ⁶⁺	18	0.42	-9.76278(10)
Kr ⁶⁺	36	0.38875	-8.617309(74)
Xe ⁶⁺	54	0.3425	-7.235291(60)
Rn ⁶⁺	86	0.350	-7.127372(63)

Tavola 2.8: Energie variazionali per gli ioni positivi (gruppo VIII). Z è il numero atomico, b è il parametro variazionale nel fattore di Jastrow, \bar{E} l'energia variazionale (vedere testo).

2.6.4 Risultati per il DMC

Tutti i risultati di questa sezione sono stati ottenuti con il DMC nella versione di Reynolds, utilizzando i valori calcolati nella sezione precedente per il parametro b della funzione di prova. Per ogni elemento il calcolo di \bar{E}_0 [$N(0) = 400$, $M = 10^4$, $\bar{M} = 10^5$, $M_B = 200$, $B = 500$] è stato effettuato per diversi valori di Δt (di norma 6), e il risultato è stato estrapolato a $\Delta t = 0$ come descritto nella Sez. 2.4 per correggere l'errore dovuto all'approssimazione a tempi corti. In Fig. 2.3 è mostrato il caso di Si²⁺: \bar{E}_0 è riportato in funzione di Δt assieme alla curva di interpolazione, scelta della forma $\alpha + \beta\Delta t^3 + \gamma\Delta t^{3.5} + \delta\Delta t^4$, ed è mostrata l'estrapolazione a $\Delta t = 0$. Tutti gli altri risultati, già estrapolati a $\Delta t = 0$, sono mostrati nelle Tab. (2.9-2.16) assieme al corrispondente valore sperimentale (calcolato come somma delle successive energie di ionizzazione o affinità elettroniche per i singoli elettroni[78, 79]) e sono espressi in unità atomiche[11]. L'errore sul risultato sperimentale è di 4 unità sull'ultima cifra, quello sul risultato DMC è mostrato in parentesi accanto al risultato stesso.

In Fig. 2.4, le differenze relative e assolute fra risultati DMC e sperimentali permettono di tracciare un primo bilancio. Considerando le differenze relative, per quasi tutti gli elementi studiati, esse cadono nella fascia compresa fra -1% e 1% , esclusi Mg, Ga, Sb e Cs (fra -2% e 2%), i metalli alcalini da Na a Rb (fra 2% e 4%) e I, Xe, Po, At e Rn (fra 3% e 14% : i risultati non sono mostrati in figura in quanto il risultato sperimentale è dubbio come discusso più avanti). Analizzando invece le differenze assolute e fissando una fascia fra -0.5 e $+0.5$ eV/elettrone (e cioè fra -1 e 1 eV), osserviamo che sono quasi tutte incluse

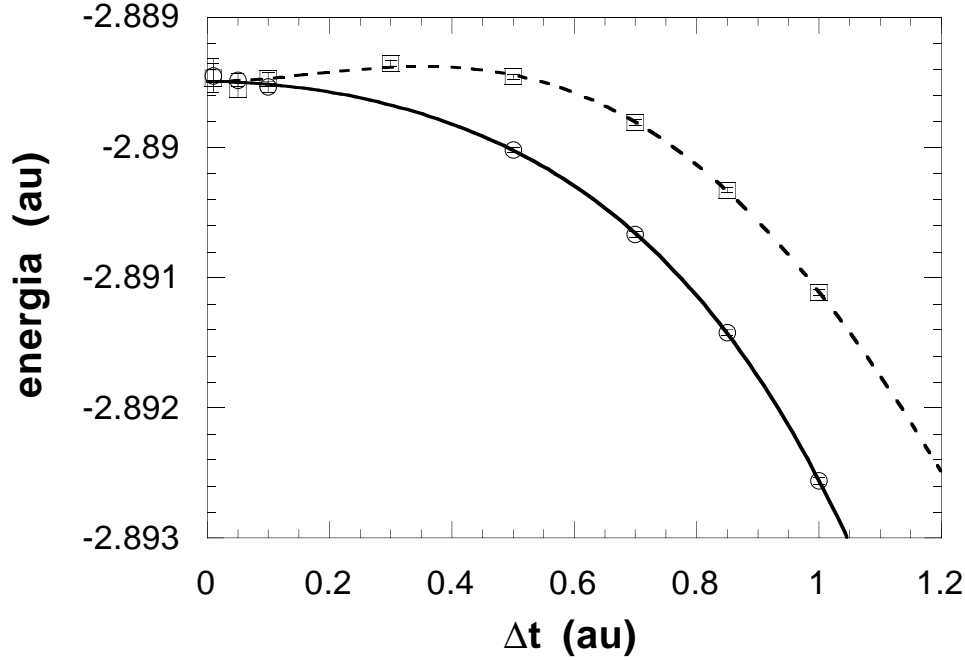


Figura 2.3: Energie DMC per lo ione Si^{2+} in funzione del passo temporale Δt . I punti (○) e la curva di interpolazione continua $\alpha + \beta\Delta t^3 + \gamma\Delta t^{3.5} + \delta\Delta t^4$ si riferiscono ai risultati ottenuti con l'algoritmo di Reynolds. I valori calcolati per i parametri sono $\alpha = -2.889503(15)$, $\beta = -0.0157(19)$, $\gamma = 0.0254(40)$ e $\delta = -0.0128(21)$ con l'incertezza in parentesi. Il corrispondente valore di χ^2 è 0.69. Il valore dell'energia estrapolato a $\Delta t = 0$ è $\bar{E}_0 = -2.889503(15)$ e questa è la stima DMC per l'energia di stato fondamentale. I punti (□) e la curva di interpolazione tratteggiata $\alpha_1 + \beta_1\Delta t^2 + \gamma_1\Delta t^3 + \delta_1\Delta t^4$ si riferiscono ai risultati ottenuti con l'algoritmo di Umrigar: $\alpha_1 = -2.889491(32)$, $\beta_1 = 0.00292(66)$, $\gamma_1 = -0.0064(14)$, $\delta_1 = 0.00184(82)$, $\chi^2 = 4.12$ e $\bar{E}_0 = -2.889491(32)$ (vedere testo).

atomo	Z	E_{exp} (Ha)	\bar{E}_0 (Ha)
He	2	-2.90331	-2.908532(59)
Be	4	-1.01182	-1.005706(13)
Mg	12	-0.83351	-0.8421513(94)
Ca	20	-0.66090	-0.6642001(57)
Sr	38	-0.61463	-0.618948(19)
Ba	56	-0.55918	-0.561024(13)

Tavola 2.9: Energie di stato fondamentale per gli atomi neutri (He e gruppo IIA). Z è il numero atomico, \bar{E}_0 l'energia DMC e E_{exp} quella sperimentale (vedere testo).

ione	Z	E_{exp} (Ha)	\bar{E}_0 (Ha)
H ⁻	1	-0.52743	-0.526053(79)
Li ⁻	3	-0.22086	-0.2215563(86)
Na ⁻	11	-0.20899	-0.2156833(52)
K ⁻	19	-0.17796	-0.182415(13)
Rb ⁻	37	-0.17136	-0.1763871(77)
Cs ⁻	55	-0.16043	-0.1636144(33)

Tavola 2.10: Energie di stato fondamentale per gli ioni negativi (gruppo IA). Z è il numero atomico, \bar{E}_0 l'energia DMC e E_{exp} quella sperimentale (vedere testo).

ione	Z	E_{exp} (Ha)	\bar{E}_0 (Ha)
B ⁺	5	-2.31829	-2.2988316(83)
Al ⁺	13	-1.73732	-1.744489(10)
Ga ⁺	31	-1.8823	-1.917627(23)
In ⁺	49	-1.72351	-1.735069(15)
Tl ⁺	81	-1.8470	-1.845536(18)

Tavola 2.11: Energie di stato fondamentale per gli ioni positivi (gruppo IIIB). Z è il numero atomico, \bar{E}_0 l'energia DMC e E_{exp} quella sperimentale (vedere testo).

ione	Z	E_{exp} (Ha)	\bar{E}_0 (Ha)
C ²⁺	6	-4.12985	-4.092171(12)
Si ²⁺	14	-2.88971	-2.889503(15)
Ge ²⁺	32	-2.9374	-2.964900(32)
Sn ²⁺	50	-2.61787	-2.627146(16)
Pb ²⁺	82	-2.7289	-2.7290989(63)

Tavola 2.12: Energie di stato fondamentale per gli ioni positivi (gruppo IVB). Z è il numero atomico, \bar{E}_0 l'energia DMC e E_{exp} quella sperimentale (vedere testo).

ione	Z	E_{exp} (Ha)	\bar{E}_0 (Ha)
N ³⁺	7	-6.44436	-6.385991(12)
P ³⁺	15	-4.2774	-4.267741(10)
As ³⁺	33	-4.1439	-4.164267(31)
Sb ³⁺	51	-3.68	-3.629541(18)
Bi ³⁺	83	-3.723	-3.703604(13)

Tavola 2.13: Energie di stato fondamentale per gli ioni positivi (gruppo VB). Z è il numero atomico, \bar{E}_0 l'energia DMC e E_{exp} quella sperimentale (vedere testo).

ione	Z	E_{exp} (Ha)	\bar{E}_0 (Ha)
O ⁴⁺	8	-9.26126	-9.178216(18)
S ⁴⁺	16	-5.9067	-5.875234(20)
Se ⁴⁺	34	-5.51	-5.509899(15)
Te ⁴⁺	52	-4.757	-4.734385(23)
Po ⁴⁺	84	-4.92	-4.764977(30)

Tavola 2.14: Energie di stato fondamentale per gli ioni positivi (gruppo VIB). Z è il numero atomico, \bar{E}_0 l'energia DMC e E_{exp} quella sperimentale (vedere testo).

ione	Z	E_{exp} (Ha)	\bar{E}_0 (Ha)
F ⁵⁺	9	-12.5809	-12.473091(12)
Cl ⁵⁺	17	-7.7990	-7.724793(15)
Br ⁵⁺	35	-7.041	-6.997540(17)
I ⁵⁺	53	-6.9	-5.940821(23)
At ⁵⁺	85	-6.21	-5.912081(15)

Tavola 2.15: Energie di stato fondamentale per gli ioni positivi (gruppo VIIB). Z è il numero atomico, \bar{E}_0 l'energia DMC e E_{exp} quella sperimentale (vedere testo).

ione	Z	E_{exp} (Ha)	\bar{E}_0 (Ha)
Ne ⁶⁺	10	-16.4034	-16.268025(25)
Ar ⁶⁺	18	-9.84054	-9.768896(49)
Kr ⁶⁺	36	-8.71	-8.625198(35)
Xe ⁶⁺	54	-8.4	-7.244511(29)
Rn ⁶⁺	86	-7.64	-7.137842(15)

Tavola 2.16: Energie di stato fondamentale per gli ioni positivi (gruppo VIII). Z è il numero atomico, \bar{E}_0 l'energia DMC e E_{exp} quella sperimentale (vedere testo).

atomo	Z	E_{exp}^{EA} (eV)	\bar{E}^{EA} (eV)
H	1	0.684(2)	0.754209(3)
Li	3	0.5447(2)	0.6180(5)
Na	11	0.5657(1)	0.547930(25)
K	19	0.5650(4)	0.50147(10)
Rb	37	0.5779(2)	0.48592(2)
Cs	55	0.5558(9)	0.471630(25)

Tavola 2.17: Affinità elettroniche per H e i metalli alcalini. Z è il numero atomico, \bar{E}^{EA} è il risultato DMC, E_{exp}^{EA} il risultato sperimentale[78] (vedere testo).

tranne quelle per C, N, Ar, Br, Sb (fra 1 e 2 eV), O, F, Ne, Cl, Kr (fra 2 e 4 eV), I, Xe, Po, At e Rn (oltre 4 eV). La ragione della discrepanza fra risultati sperimentali e teorici va ricercata nell'approssimazione di pseudopotenziale (separazione fra core e valenza, rilassamento del core) e nel trattamento approssimato dell'interazione elettrone-elettrone alla base dello schema DFT-LDA in cui gli pseudopotenziali vengono generati. Fra gli elementi i cui pseudopotenziali manifestano la peggior portabilità, ci sono i metalli alcalini da Na a Cs. Come già osservato e discusso da Focher *et al.*[24], questo è dovuto alla separazione fra core e valenza che in questi elementi non è netta. In conseguenza di ciò gli pseudopotenziali standard per questi elementi[20, 40] manifestano una cattiva trasferibilità all'interno della stessa DFT-LDA a causa della non linearità del funzionale di scambio e correlazione[52]; ma, mentre in DFT-LDA è possibile superare questa difficoltà utilizzando la correzione di core non lineare introdotta da Louie, Froyen e Cohen[52], una generalizzazione ai metodi QMC non pare possibile. Un ulteriore problema nel caso dei metalli alcalini nasce dal fatto che DFT-LDA non è in grado di descrivere gli ioni negativi (essi risultano non legati) e questo si riflette nel grosso errore con cui le affinità elettroniche, e dunque l'energia del sistema a due elettroni, vengono riprodotte dagli pseudopotenziali (Tab. 2.17). Un altro gruppo che mostra pessima portabilità, e per cui le differenze relative e assolute fra risultato DMC e sperimentale risultano molto più grandi di tutte le altre, è costituito da Po, At e Rn (differenze relative fra 3% e 7%, differenze assolute fra 0.15 Ha e 0.51 Ha),

e da I e Xe (differenze relative fra 13% e 14%, differenze assolute fra 0.9 Ha e 1.2 Ha), ma in questi casi avanziamo seri dubbi sulla validità degli unici risultati sperimentali da noi trovati in letteratura[79], che sono abbastanza datati e non riportati in pubblicazioni più recenti[78]. Per quanto riguarda i rimanenti elementi, le differenze relative rimangono contenute nella fascia $\pm 2\%$, e la portabilità, che dipende anche dalla precisione richiesta nelle particolari applicazioni, va valutata considerando le differenze assolute (si arriva fino a circa -1.8 eV/elettrone nel caso del Ne).

I risultati ottenuti sono in parziale accordo con quanto trovato da Focher *et al.*[24], per i corrispondenti sistemi a un solo elettrone di valenza. Le discordanze più significative riguardano gli elementi della prima e della terza riga come mostrato in Fig. 2.5: in entrambi i casi le differenze fra gli errori assoluti per elettrone (calcolati come differenza fra risultato sperimentale e teorico) per i sistemi a uno e due elettroni aumentano con il numero atomico Z . Nel caso della prima riga l'errore per il sistema a due elettroni è approssimativamente lo stesso di quello del corrispondente sistema a un elettrone per Li e Be ed è più grande a partire da B fino a Ne. Nel caso della terza riga l'errore è approssimativamente lo stesso per Rb e Sr, più piccolo per Ga, Ge, As e Se, più grande per Br e Kr. Ricordando gli intervalli introdotti in precedenza osserviamo che nel caso di due elettroni per Ga, Ge e As l'errore assoluto scende all'interno della fascia ± 0.5 eV/elettrone, mentre per C, N, O, Cl, Br e Kr esso aumenta (in modulo) ed esce da tale fascia. Tutte queste considerazioni indicano che l'interazione elettrone-elettrone (due elettroni interagenti invece di uno solo) introduce differenze significative, e perciò un'analisi quantitativa della portabilità non può limitarsi al solo caso di uno o anche due elettroni.

Per chiarire meglio quale sia la possibile sorgente dell'errore di portabilità (cattiva descrizione dell'interazione elettrone-elettrone in DFT-LDA e/o cattiva approssimazione di pseudopotenziale), in Fig. 2.6 abbiamo confrontato le energie di legame dei due elettroni di valenza per l'atomo full-core e per l'atomo di sola valenza in DFT-LDA, mentre in Fig. 2.7 abbiamo messo a confronto le energie di legame DMC e DFT-LDA per l'atomo di sola valenza. Il primo confronto dovrebbe fornirci informazioni sulla bontà dell'approssimazione

di pseudopotenziale e il secondo sulla bontà dell'approssimazione di densità locale nell'atomo di sola valenza. Osservando le figure è possibile fare delle considerazioni generali; in Fig. 2.6 l'errore assoluto aumenta all'interno dello stesso periodo (questo è dovuto al progressivo allontanamento dalla configurazione di riferimento per lo pseudopotenziale che è l'atomo neutro) come in Fig. 2.7, ma rimane approssimativamente lo stesso per tutti i periodi al contrario dell'andamento di Fig. 2.7 in cui vi è una diminuzione con l'aumento del periodo. In entrambe le figure (eccetto He in Fig. 2.6) le differenze assolute sono positive e dunque sia le energie di legame DFT-LDA per l'atomo full-core che quelle DMC per l'atomo di sola valenza sono in valore assoluto più grandi di quelle DFT-LDA per l'atomo di sola valenza: vi è perciò una compensazione degli errori introdotti dalle due diverse approssimazioni nel passare dall'atomo DFT-LDA full-core a quello DMC di sola valenza che, però, diminuisce al crescere del periodo. Inoltre, al di là delle possibili compensazioni fra gli errori di origine diversa, osserviamo che nel passaggio dall'atomo DFT-LDA full-core a quello DMC di sola valenza la principale fonte d'errore è nell'approssimazione di pseudopotenziale piuttosto che in quella di densità locale (per gli elementi della prima riga le due approssimazioni introducono errori simili) anche se, in generale, l'approssimazione di pseudopotenziale migliora più l'atomo si trova vicino alla configurazione di riferimento: questo effetto può essere rilevante per gli elementi di fine periodo che nel nostro caso sono fortemente ionizzati e dunque molto distanti da tale configurazione. Naturalmente, l'analisi delle Fig. 2.6 e 2.7 non permette un'estrapolazione dell'errore fra risultato DMC ed esperimento (Fig. 2.4) ma ci fornisce ugualmente una stima ragionevole del suo ordine di grandezza.

Vale la pena osservare che nel DMC gli ioni negativi risultano legati, pur avendo energie in disaccordo con l'esperimento per i motivi sopra esposti, mentre non lo sono nell'ambito originale in cui gli pseudopotenziali vengono generati, e cioè la Teoria del Funzionale Densità in approssimazione locale. Questa è una testimonianza diretta del fatto che gli pseudopotenziali hanno, seppure con gradazioni molto diverse fra loro, una certa identità fisica, indipendentemente da come sono stati generati e dall'approssimazione usata per descrivere l'interazione elettrone-elettrone. D'altro canto, per gli elementi che presentano la maggior discrepanza

fra teoria ed esperimento, emerge l'esigenza di avere a disposizione pseudopotenziali che non diano solo risultati in accordo qualitativo con l'esperimento. A questo proposito sarebbe auspicabile poter generare tali operatori direttamente all'interno del DMC o di altri schemi realmente a molti corpi, in cui l'interazione elettrone-elettrone viene trattata esattamente. Molto recentemente qualche progresso è stato fatto anche in questa direzione[10].

2.6.5 Confronto fra diversi DMC

Nel caso di Si^{2+} abbiamo messo a confronto i due algoritmi studiati per il DMC e cioè quello di Reynolds *et al.*[4] e quello più recente di Umrigar *et al.*[70]. In entrambi i casi abbiamo posto $N(0) = W_0 = 400$, $\bar{M} = 10^5$, $M_B = 200$, $B = 500$ e, nel primo, $M = 10^4$, mentre nel secondo $M = 2M_1 = 2 \cdot 10^4$. In Fig. 2.3 sono riportate le energie \bar{E}_0 in funzione di Δt per entrambi i metodi, assieme alle due curve di interpolazione. Le estrapolazioni a $\Delta t = 0$ sono, nel primo caso $\bar{E}_0 = -2.889503(15)$, e nel secondo $\bar{E}_0 = -2.889491(32)$. Osserviamo che, nel limite di $\Delta t \rightarrow 0$, i due metodi danno lo stesso risultato (entro la statistica), mentre, per Δt finito, l'algoritmo di Umrigar dà risultati la cui dipendenza da Δt è meno marcata (per un confronto più completo vedere il prossimo capitolo). A parità di parametri di esecuzione le incertezze statistiche e i tempi di esecuzione risultano sostanzialmente gli stessi per entrambe le tecniche.

Per quanto riguarda la correzione dell'errore sistematico nell'algoritmo di Umrigar, abbiamo posto $L = M_B = 200$, ma, per uno dei valori di Δt , un calcolo fatto prima con $L = 200$ e poi con $L = 0$ non ha evidenziato alcuna differenza (entro la statistica); sembra ragionevole ipotizzare che la semplicità del sistema considerato, che si riflette positivamente sulla stabilità della popolazione e che quindi introduce solo una piccola correzione a E_{cum} nel secondo membro dell'Eq. (2.112), dia origine a un errore sistematico ben al di sotto dell'incertezza statistica.

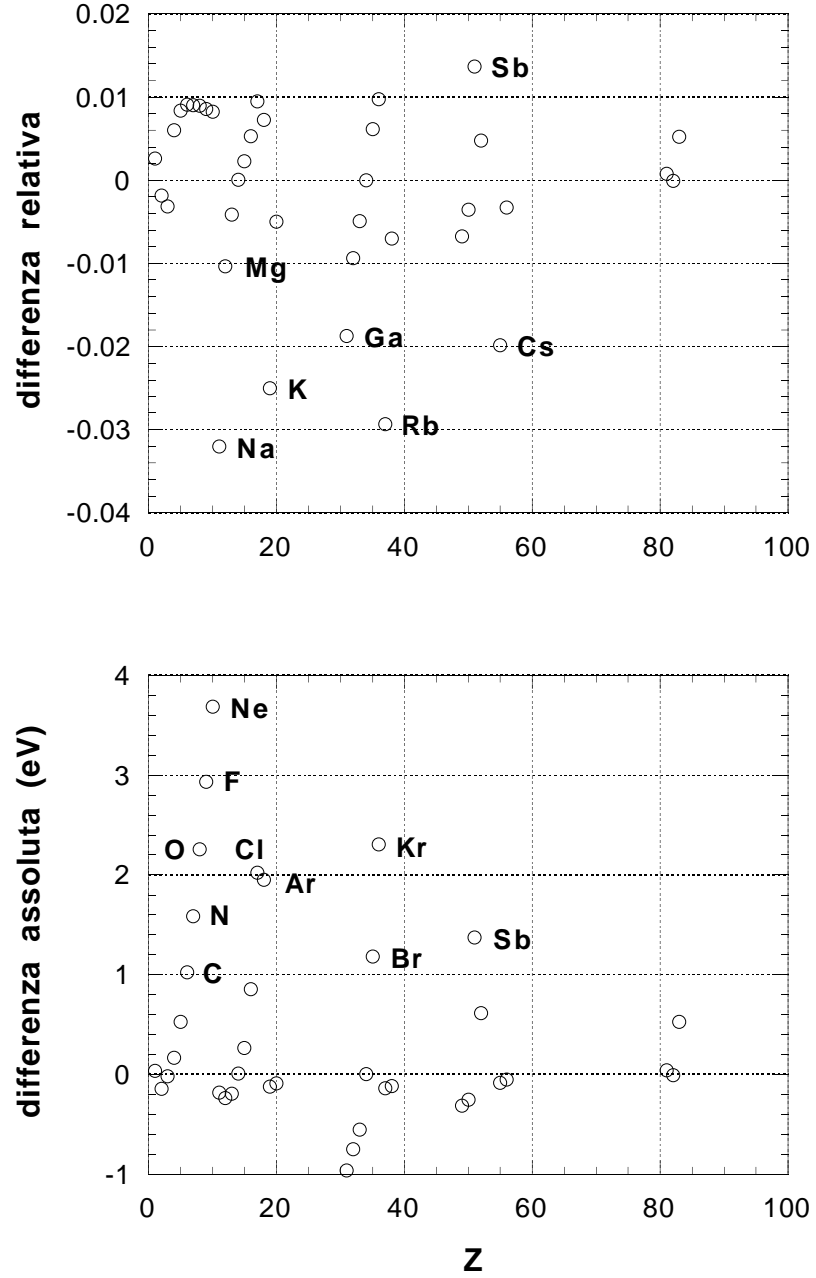


Figura 2.4: Confronto fra le energie di legame sperimentali (E_{exp}) e DMC (\bar{E}_0) dei due elettroni di valenza. In funzione del numero atomico Z sono riportate le differenze relative $(|E_{exp}| - |\bar{E}_0|)/|E_{exp}|$ (riquadro superiore) e assolute $|E_{exp}| - |\bar{E}_0|$ (riquadro inferiore) come discusso nel testo.

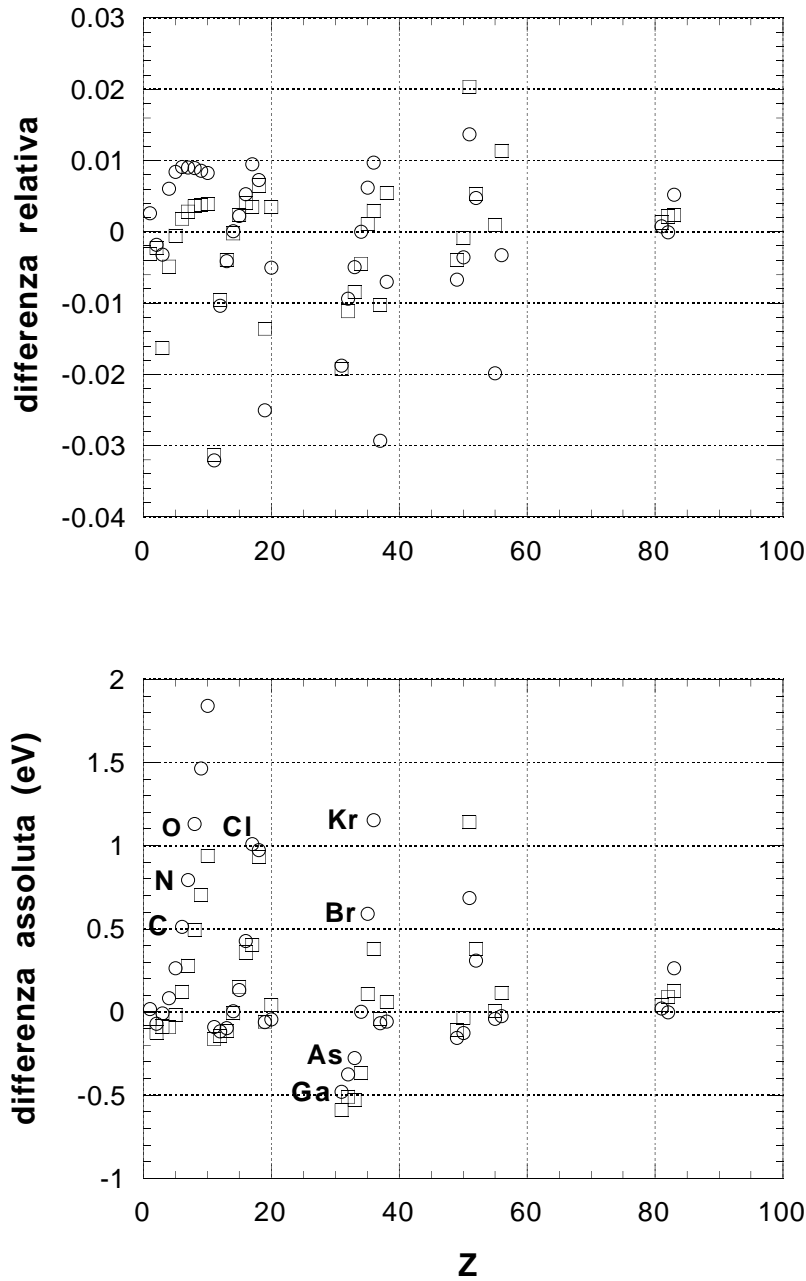


Figura 2.5: Confronto fra le energie di legame sperimentali (E_{exp}) e DMC (\bar{E}_0) di uno e due elettroni di valenza. In funzione del numero atomico Z sono riportate le differenze relative $(|E_{exp}| - |\bar{E}_0|)/|E_{exp}|$ (riquadro superiore) e assolute $|E_{exp}| - |\bar{E}_0|$ (riquadro inferiore) per il sistema a due elettroni [simbolo (\circ)] e a un elettrone[24] [simbolo (\square)]. Le energie sono riferite al singolo elettrone.

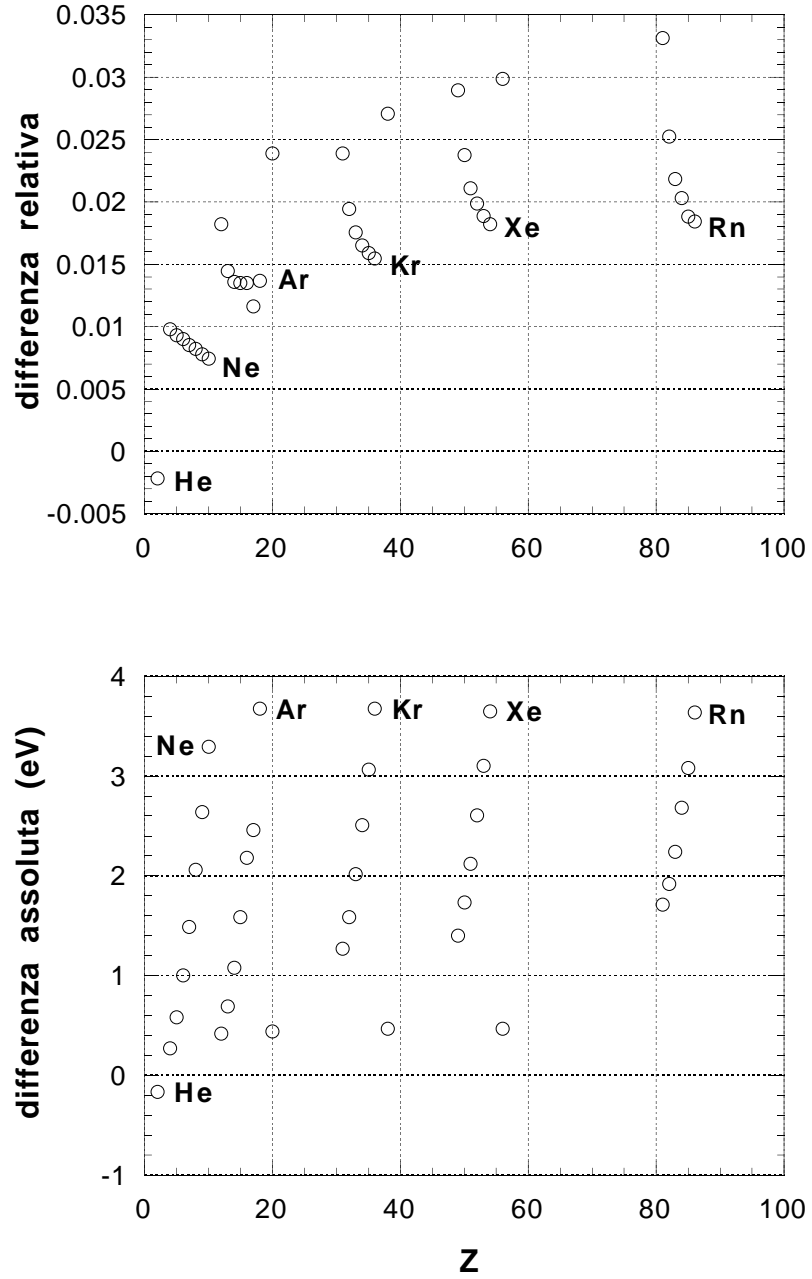


Figura 2.6: Confronto fra le energie di legame dei due elettroni di valenza per l'atomo full-core (E_{LDA}^{FC}) e quello di sola valenza (E_{LDA}) in DFT-LDA. In funzione del numero atomico Z sono riportate le differenze relative ($(|E_{LDA}^{FC}| - |E_{LDA}|) / |E_{LDA}^{FC}|$) (riquadro superiore) e assolute $|E_{LDA}^{FC}| - |E_{LDA}|$ (riquadro inferiore) come discusso nel testo.

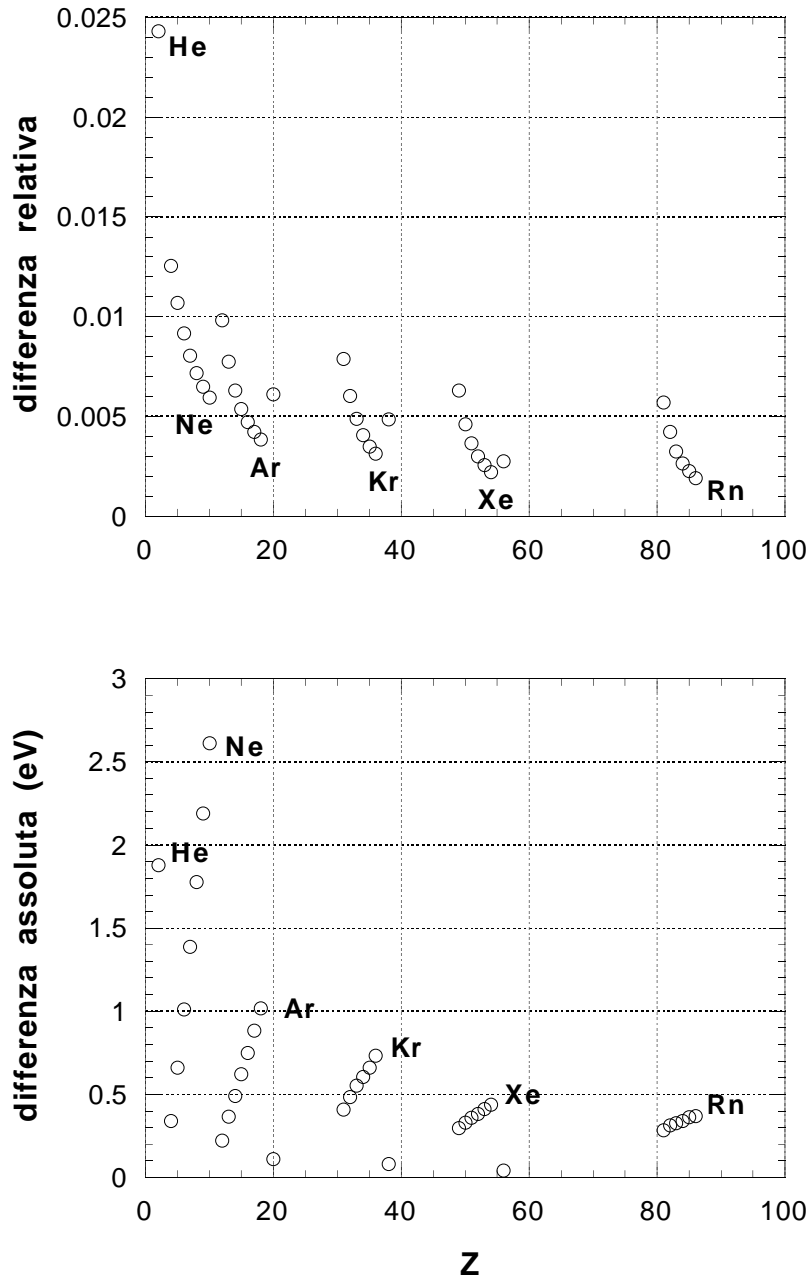


Figura 2.7: Confronto fra le energie di legame DMC (\bar{E}_0) e DFT-LDA (E_{LDA}) per i due elettroni nell'atomo di sola valenza. In funzione del numero atomico Z sono riportate le differenze relative $(|\bar{E}_0| - |E_{LDA}|)/|\bar{E}_0|$ (riquadro superiore) e assolute $|\bar{E}_0| - |E_{LDA}|$ (riquadro inferiore) come discusso nel testo.

2.7 Conclusioni

L'onere computazionale delle simulazioni Monte Carlo Variazionale e di Diffusione può essere fortemente ridotto mediante l'utilizzo di hamiltoniani di sola valenza al posto degli hamiltoniani full-core permettendo così un allargamento del campo d'indagine, finora limitato ad atomi e molecole leggere, a sistemi più complessi quali i solidi. Dato l'elevato grado di accuratezza che i metodi QMC possono raggiungere, emerge l'esigenza fondamentale di avere a disposizione hamiltoniani di sola valenza altrettanto accurati. Purtroppo non sono ancora disponibili operatori generati direttamente in ambito QMC o all'interno di altri metodi realmente a molti corpi e, di conseguenza, in letteratura si fa ricorso agli hamiltoniani di sola valenza generati all'interno di teorie di campo medio autoconsistente quali DFT-LDA e Hartree-Fock che trattano in maniera approssimata l'interazione elettrone-elettrone. L'impiego "indiscriminato" di tali operatori è in contrasto con i requisiti di accuratezza accennati in precedenza ed è a del tutto insoddisfacente anche da un punto di vista teorico di principio: la giustificazione della bontà dell'approssimazione introdotta avviene solo a posteriori.

Per cercare di far luce sul tipo e sulla qualità dell'approssimazione che si introduce, abbiamo considerato in dettaglio un sistema semplice costituito da due soli elettroni di valenza (nello stato fondamentale a simmetria sferica e di singoletto di spin) e descritto dalla componente $\ell = 0$ degli pseudopotenziali a norma conservata standard, per tutti gli elementi *sp* della Tavola Periodica. Pur non essendo in grado di dare una risposta univoca e completa, questa analisi mette in luce chiaramente quali sono le principali fonti di approssimazione connesse con l'uso degli pseudopotenziali standard e quale l'ordine di grandezza dell'errore introdotto, e dovrebbe permettere di stimare a priori l'importanza dell'approssimazione introdotta mediante l'uso di uno pseudopotenziale convenzionale in una simulazione QMC.

2.8 Ringraziamenti

Un ringraziamento particolare va a L. Mitáš per i suoi preziosi insegnamenti sulle simulazioni QMC e a G. B. Bachelet per gli innumerevoli suggerimenti e consigli. Ringraziamo P. Focher per averci messo a disposizione i risultati per i sistemi a un elettrone e V. Fiorentini per le utili discussioni.

3 Calcolo parallelo

3.1 Introduzione

L'esigenza di ridurre l'onere computazionale dei Metodi Monte Carlo Variazionale e di Diffusione, che rappresenta il fattore limitante per l'impiego di queste tecniche nello studio di sistemi "complessi" nonostante l'utilizzo di hamiltoniani di sola valenza introduca un sostanziale miglioramento, trova una soluzione naturale nell'utilizzo di nuove metodologie computazionali quali il calcolo parallelo grazie alle favorevoli caratteristiche proprie dei metodi QMC. Sulla base delle considerazioni precedenti, abbiamo deciso di affrontare il problema dello sviluppo di programmi per le simulazioni VMC e DMC su calcolatori paralleli, al fine di verificare la convenienza del calcolo parallelo rispetto a quello convenzionale o "sequenziale" o "seriale" come strumento in grado di permettere l'estensione delle simulazioni QMC a un numero sempre maggiore di sistemi fisici interessanti. Per fare questo e mettere a punto uno schema di parallelizzazione il più generale possibile, ci siamo concentrati su un problema di prova semplice ma significativo, il sistema a due elettroni di valenza studiato nel capitolo precedente, che pone le stesse problematiche computazionali di sistemi più complessi senza però introdurre ulteriori complicazioni di natura fisica. Infatti, per arrivare a una costruzione generale, il procedimento seguito nello sviluppo dei programmi paralleli deve naturalmente essere indipendente dal problema di prova scelto, ma i risultati specifici devono essere quelli di un caso noto per poter verificare la correttezza dell'intera procedura.

Una considerazione importante riguarda la scelta dell'architettura parallela in quanto questa influenza in maniera rilevante la costruzione dei programmi. Nel nostro caso la disponibilità in loco di un certo numero di stazioni di lavoro collegate in rete ci ha fornito la base hardware per un'architettura parallela distribuita realizzata coordinando le varie stazioni per mezzo di un opportuno software di gestione. Tale architettura è risultata del tutto adeguata per le simulazioni QMC da noi considerate e ha il grosso pregio di avere costi contenuti, grande versatilità e larga diffusione.

3.2 Calcolo parallelo

Le difficoltà principali legate allo sviluppo del software per calcolatori paralleli sono sia teoriche che pratiche. Rispetto al modello di programmazione sequenziale delle architetture convenzionali in cui le istruzioni vengono eseguite una dopo l'altra nel preciso ordine stabilito dal programmatore¹, nelle architetture parallele questo vincolo di sequenzialità viene meno nel senso che è possibile eseguire più istruzioni contemporaneamente. Questo fatto, pur portando a vantaggi rilevanti quali il drastico abbattimento dei tempi di attesa per il risultato di un calcolo, mette il programmatore di fronte a tutta una serie di nuovi problemi dovuti alle innumerevoli condizioni e situazioni in cui il programma può incorrere e che devono essere affrontate correttamente. Ciò è conseguenza del fatto che il risultato non è raggiunto sequenzialmente un passo dopo l'altro, ma attraverso un insieme di gruppi di istruzioni concorrenti o processi o *task* che hanno la possibilità di interagire mediante lo scambio di informazioni e il cui ordine di esecuzione non è rigidamente fissato. È dunque necessario identificare all'interno del problema analizzato i possibili task, provvedere alla loro creazione, alla loro corretta sincronizzazione, allo scambio delle informazioni fra di essi e, infine, alla loro terminazione.

A livello pratico, i problemi di cui abbiamo parlato sopra possono avere soluzioni diverse a seconda del tipo di hardware utilizzato rendendo così il software sviluppato per una certa applicazione su uno specifico calcolatore totalmente inutilizzabile su calcolatori diversi. Uno dei grossi sforzi tutt'ora in corso nel campo dello sviluppo dei linguaggi di programmazione, è volto al superamento del problema della portabilità del software da un calcolatore all'altro mediante la definizione di modelli di programmazione astratti (che poi vengono realizzati per esempio come estensioni di linguaggi comunemente utilizzati quali C o FORTRAN) che ben si adattino al più gran numero possibile di architetture parallele diverse e che dunque siano in larga misura indipendenti dall'hardware specifico. In questo modo l'utente è fortemente avvantaggiato in quanto è in grado di sviluppare programmi indipendenti dalle specificità

¹I calcolatori vettoriali per alcuni aspetti fanno eccezione e si avvicinano di più a quelli paralleli.

dell'hardware e in grado di funzionare su architetture diverse.

Nella maggior parte dei problemi non tutte le operazioni sono indipendenti le une dalle altre e perciò si ha comunicazione fra i processori. Fra i programmi paralleli, e cioè strutturati per utilizzare contemporaneamente più processori, possiamo distinguere quelli chiamati debolmente accoppiati in quanto richiedono poca o addirittura nulla comunicazione fra i processori e quelli detti fortemente accoppiati perché richiedono molta comunicazione. Inoltre, le operazioni di coordinamento di più task distribuiti fra più processori, al fine di eseguire uno stesso programma, comportano un maggiore sforzo computazionale e maggiori tempi morti rispetto al corrispondente programma sequenziale. È dunque importante, come vedremo nella Sez. 3.5, tenere in debita considerazione questi aspetti durante la stesura del software se non si vuole ridurre, anche drasticamente, l'efficienza di calcolo e cioè il guadagno reso possibile dall'aumentato numero di processori a causa della comunicazione e della sincronizzazione fra i vari processi.

Per sfruttare al meglio un calcolatore parallelo è necessario selezionare fra i vari problemi che si vogliono risolvere quelli per cui è conveniente il calcolo parallelo ed individuare bene le parti del calcolo che sono parallelizzabili. Parallelizzare un programma è conveniente solo se la frazione di calcolo che va eseguita in modo sequenziale è molto minore di quella che si può eseguire in parallelo e se il tempo di calcolo è molto maggiore di quello speso per la comunicazione fra i vari processi. Una semplice formula relativa al tempo di comunicazione T_C di un processo con un altro è [80]

$$T_C = T_L + B \times N_B \quad (3.1)$$

dove T_L è il tempo di latenza e cioè il tempo che passa fra l'inizio della trasmissione e l'inizio della ricezione del messaggio (dovuto alle caratteristiche fisiche del mezzo e alle modalità di comunicazione), B è il tempo di trasmissione di un byte e N_B è la lunghezza del messaggio in byte.

Fra le architetture parallele possiamo distinguere i calcolatori paralleli massivi o calcolatori paralleli veri e propri, in cui tutti i processori stanno nello stesso contenitore e sono

collegati tramite circuiti interni, e i calcolatori a parallelismo distribuito o reti di calcolatori o *cluster*[81] che sono formati da un insieme di stazioni di lavoro o *workstation* indipendenti che comunicano tramite una connessione software su un mezzo fisico detto rete. La differenza fondamentale fra i due tipi di architetture appena descritti è che nei calcolatori paralleli massivi, grazie alle loro connessioni hardware interne, i vari processori comunicano fra loro molto più velocemente che nelle reti di calcolatori e che i calcolatori paralleli massivi hanno in genere molti più processori di un cluster.

Come vedremo nel prossimo paragrafo esistono architetture di tipo SIMD e MIMD[80, 82]. I calcolatori SIMD (*Single Instruction, Multiple Data*) consistono di un insieme di processori identici con la propria memoria locale che eseguono simultaneamente la stessa istruzione su dati diversi. I calcolatori MIMD (*Multiple Instruction, Multiple Data*) sono formati da un insieme di processori di solito identici (fanno eccezione i cluster), in genere più potenti di quelli che costituiscono i SIMD, in grado di eseguire istruzioni diverse su dati diversi in maniera asincrona. Mentre i calcolatori paralleli massivi possono essere sia di tipo SIMD che di tipo MIMD, i cluster possono essere solo di tipo MIMD.

3.3 Architetture parallele SIMD e MIMD

Poiché, come appena detto, nei calcolatori SIMD tutti i processori eseguono in maniera sincrona la stessa istruzione, ognuno di questi non può essere usato contemporaneamente da più utenti per programmi diversi e sono vantaggiosi esclusivamente per problemi la cui soluzione si ottiene eseguendo le stesse operazioni contemporaneamente su tanti dati diversi. Nei calcolatori SIMD il calcolo è sincrono nel senso che tutti i processori iniziano a eseguire contemporaneamente la stessa istruzione e tutti devono aver terminato prima che ognuno possa eseguire l'istruzione successiva. Questa peculiare caratteristica di funzionamento rende poco flessibili i calcolatori SIMD se confrontati con i calcolatori MIMD che, come scritto nella sezione precedente, sono asincroni. Nei calcolatori SIMD i processori comunicano per mezzo di linee dedicate la cui velocità tipica è di circa 25 Mbyte/s (valore massimo per l'architettura APE100).

I calcolatori paralleli MIMD si possono dividere a loro volta in due categorie a seconda dell'organizzazione della memoria: i calcolatori a memoria condivisa e i calcolatori a memoria distribuita. Le architetture a memoria condivisa sono formate da vari processori (da due a qualche decina) e una grande memoria comune, tutti collegati tramite un unico circuito interno detto *bus* (la cui velocità tipica è compresa fra 100 Mbyte/s e 1000 Mbyte/s). La comunicazione con la memoria è tanto più veloce quanto più è piccolo il numero di processori che usano la connessione hardware di comunicazione contemporaneamente, in quanto su di essa può scrivere un solo processore alla volta. Inoltre, in questo caso, è molto importante l'ordine con cui i vari processori scrivono e leggono i dati in memoria. I calcolatori MIMD a memoria distribuita sono formati da tanti processori, ognuno con la propria memoria locale, che comunicano per mezzo di linee dedicate o via rete nel caso di cluster. Nel caso di una rete di calcolatori, la comunicazione fra i processori è in genere molto più lenta rispetto a quella degli altri calcolatori MIMD, specialmente se la rete non è dedicata, perciò la strategia di programmazione è differente.

Esempi di architetture SIMD sono APE100 (INFN), CM-2 (Thinking Machines Corporation). Esempi di architetture MIMD a memoria condivisa sono i calcolatori Power Series (Silicon Graphics) e SPARCcenter 2000 (Sun Microsystems). Esempi di architetture MIMD a memoria distribuita sono iPSC/860 Parallel Supercomputer (Intel), Paragon (Intel), CM-5 (Thinking Machines) e le reti di workstation.

3.4 Architetture parallele distribuite

Con architettura parallela distribuita[81] si intende un insieme omogeneo o eterogeneo di due o più calcolatori collegati fra loro mediante un mezzo fisico detto rete (connessione hardware) e che comunicano fra loro utilizzando un qualche tipo di protocollo di rete (connessione software). Il lavoro che può compiere un cluster è eterogeneo, nel senso che può essere utilizzato sia per programmi puramente seriali che impegnano una sola macchina che per uno o più programmi paralleli che usano più calcolatori; inoltre le varie macchine possono eseguire i task a loro più adatti. In genere, in un cluster ogni macchina ha più utenti e

dovrà lavorare a turno per ciascuno di essi cosicché nessun utente sfrutterà l'intera potenza del cluster. È perciò opportuno sviluppare i programmi in modo da sfruttare la potenza disponibile nella maniera più efficiente possibile.

Dal punto di vista del calcolo parallelo, l'efficienza di un cluster dipende fortemente dalle proprietà della rete che collega i suoi computer e tramite la quale i processi di un programma parallelo comunicano. Il fattore più limitante è la velocità della rete. La maggior parte dei cluster, oggi, usa reti di comunicazione Ethernet che è relativamente lenta con una velocità massima di trasmissione di 10 Mbit/s o FDDI (*Fiber Distributed Data Interface*) che è una rete a fibra ottica con una velocità massima di trasmissione di 100 Mbit/s. Con entrambi i tipi di rete citati la comunicazione fra i processi è completamente seriale, infatti solo una macchina alla volta può scrivere i dati sulla rete mentre tutte le macchine possono leggerli contemporaneamente. La differenza di velocità fra due tipi diversi di rete si nota specialmente quando si spediscono grossi messaggi: più è lungo il messaggio che si vuole spedire più è grande la differenza fra i tempi impiegati sulle due reti.

Una considerazione importante riguarda la struttura dei messaggi che i vari task si scambiano. Ogni messaggio comprende alcune informazioni di controllo, oltre ai dati veri e propri. Dal punto di vista della trasmissione dei dati veri e propri, i dati di controllo introducono un ritardo che sommato a quello intrinseco della rete da origine alla latenza discussa nella Sez. 3.2. In altri termini possiamo dire che la latenza è il tempo richiesto per spedire un messaggio in cui la parte di dati veri e propri ha lunghezza zero. Per quanto detto, se si vogliono ridurre i tempi morti è meglio spedire un unico messaggio lungo piuttosto che suddividerlo in tanti messaggi più corti. Infatti il numero dei dati di controllo non varia al variare delle dimensioni del messaggio. La velocità di trasmissione dipende quindi dal tipo di rete, dal protocollo di rete e dalla lunghezza del messaggio. Nel caso del PVM e cioè del software di gestione del cluster che noi abbiamo usato (Sez. 3.6), la Tab. 3.1 riporta i tempi in secondi impiegati dai due diversi tipi di rete sopra descritti per spedire messaggi di diversa lunghezza (la lunghezza si riferisce ai dati)[83]. Da questa si può notare che la latenza, e cioè il tempo impiegato per spedire un messaggio di lunghezza zero, è la stessa

lunghezza del messaggio	Ethernet (s)	FDDI (s)
0 byte (latenza)	0.0012	0.0012
1 Kbyte	0.0032	0.0025
1 Mbyte	1.2112	0.6657

Tavola 3.1: Tempi di trasmissione su diversi tipi di rete nel caso del PVM (vedere testo).

in entrambe le reti. Calcolando le velocità in Mbit/s otteniamo (in parentesi è indicata la lunghezza)

- Ethernet: 2.44 Mbit/s (1 Kbyte) e 6.61 Mbit/s (1 Mbyte);
- FDDI: 3.12 Mbit/s (1 Kbyte) e 12.02 Mbit/s (1 Mbyte).

Si vede come, per entrambe le reti, la velocità effettiva si avvicina maggiormente a quella massima per messaggi lunghi e che la differenza di velocità fra le due reti è tanto più grande quanto più lungo è il messaggio.

I motivi per cui spesso si preferisce un cluster a una macchina a parallelismo massivo sono vari: (1) disponibilità in loco di un insieme di workstation e impossibilità economica di comprare una macchina parallela massiva; (2) il tempo speso ad aspettare il proprio turno in una macchina parallela massiva molto carica supera il tempo richiesto per eseguire un programma su un cluster di workstation più lente (il numero di utenti per macchina parallela massiva è in genere molto più elevato di quello per workstation e dunque per cluster); (3) su una macchina parallela massiva è difficile programmare applicazioni che hanno bisogno di più calcolatori specializzati in compiti diversi.

Un esempio di architettura parallela distribuita commercializzata come tale è il cluster SP1 della IBM, che può essere formato da varie decine di stazioni di lavoro.

3.5 Limite di velocità

Calcoleremo, ora, il possibile aumento di velocità in un calcolo eseguito da P processori uguali dedicati che lavorano contemporaneamente sullo stesso problema. L'ipotesi più semplice è che se il problema è composto da n operazioni elementari il cui tempo di esecuzione è

lo stesso, e se ognuno dei P processori può eseguire indipendentemente un'uguale quantità di queste operazioni, allora l'intero problema sarà risolto nel tempo necessario a eseguire n/P operazioni. In questo modo l'aumento di velocità, dato dal rapporto fra tempo impiegato da un solo processore e tempo impiegato da tutti P i processori assieme, è uguale al numero di processori. Nella maggior parte dei problemi le parti intrinsecamente sequenziali, la comunicazione e spesso la sincronizzazione fra i processori fanno sì che l'aumento di velocità non scali linearmente con il numero di processori o che scali linearmente ma che la costante di proporzionalità sia inferiore a uno. Secondo Gene Amdahl[80] tutti i programmi sono composti da una frazione di calcolo puramente sequenziale s che non può essere eseguita in parallelo e da una frazione parallela p , tali che $s + p = 1$. In tal caso, nell'ipotesi che il numero di operazioni sequenziali e parallele rimanga lo stesso indipendentemente dal numero P di processori che vengono utilizzati, l'aumento di velocità S è

$$S = \frac{s + p}{s + \frac{p}{P}} = \frac{1}{s + \frac{p}{P}}, \quad (3.2)$$

dove il numeratore rappresenta il tempo impiegato a svolgere l'intero calcolo con un solo processore e il denominatore quello speso con P processori. Dall'Eq. (3.2) si nota che l'aumento di velocità è al più $1/s$ per $P \rightarrow \infty$ e cioè, anche con un numero infinito di processori, l'aumento di velocità è comunque limitato dalla frazione di calcolo puramente sequenziale. Gustafson[80] suggerì una legge alternativa a quella di Amdahl. Siano s' e p' le frazioni di calcolo rispettivamente sequenziale e parallelo per il programma eseguito su un calcolatore parallelo con P processori, tali che $s' + p' = 1$. Un unico processore per eseguire lo stesso programma impiega un tempo pari a $s' + p'P$ e dunque l'aumento di velocità S è dato da

$$S = \frac{s' + p'P}{s' + p'} = P(1 - s') + s' \quad (3.3)$$

Osserviamo che se il numero di operazioni sequenziali e parallele non varia al variare di P , allora la frazione parallela p' diminuisce all'aumentare del numero dei processori mentre quella sequenziale s' aumenta avvicinandosi a 1. Questo significa che S non aumenta

indefinitamente per $P \rightarrow +\infty$, come potrebbe sembrare a prima vista dall'Eq. (3.3), in accordo con la legge di Amdahl.

3.6 Macchina Parallela Virtuale

Se si vuole utilizzare un insieme di stazioni di lavoro per il calcolo parallelo è indispensabile avere un software di gestione che permetta a questi calcolatori collegati in rete, anche con architetture diverse, di essere visti e usati dall'utente come un unico calcolatore parallelo MIMD a memoria distribuita. Con il termine “macchina virtuale” si intende l'architettura appena descritta, i cui computer membri sono detti “ospiti”. L'ideale sarebbe un software semplice, che vada bene per tutti i tipi di architetture (ci limitiamo, però, ad architetture che funzionano sotto lo stesso sistema operativo, e cioè UNIX, e con lo stesso protocollo di rete, e cioè TCP/IP, che ormai costituiscono uno standard[84]), che sia utilizzabile usando linguaggi di programmazione standard quali FORTRAN e C, che abbia un'alta tolleranza agli errori dovuti a malfunzionamenti dei calcolatori e che sia utilizzabile anche su calcolatori paralleli massivi. Fra tutti i software disponibili[85] abbiamo scelto PVM (*Parallel Virtual Machine*), attualmente nella versione 3.3[86], perché ha tutte le caratteristiche ottimali appena descritte. PVM, sviluppato nell'ambito di un'iniziativa congiunta dei Laboratori Nazionali dell'Università del Tennessee e dell'Università di Emory (USA), è del tipo “*message-passing*” perché si basa sull'esplicito scambio di messaggi fra un processo e l'altro. Nella Tab. 3.2, tratta dal manuale d'uso[87], elenchiamo le architetture per le quali PVM è stato sviluppato.

PVM tollera eterogeneità a livello di programmi, architetture, e rete. A livello di programmi perché consente alle varie parti di cui si compone il programma parallelo di utilizzare l'ospite più adatto alla loro esecuzione. A livello di architetture perché se due o più calcolatori usano differenti rappresentazioni per i dati PVM esegue la conversione in modo appropriato. A livello di rete perché PVM permette agli ospiti della macchina virtuale di essere collegati con diversi tipi di rete (Ethernet, FDDI, ecc.). Inoltre, PVM opera in ambienti in cui nè la rete, nè i processori dei calcolatori individuali sono dedicati.

PVM_ARCH	Architettura	Note
AFX8	Alliant FX/8	
ALPHA	DEC Alpha	DEC OSF-1
BAL	Sequent Balance	DYNIX
BFLY	BBN Butterfly TC2000	
BSD386	80386/486 Unix box	BSDI
CM2	Thinking Machines CM2	Sun front-end
CM5	Thinking Machines CM5	
CNVX	Convex C-series	
CNVXN	Convex C-series	native mode
CRAY	C-90, YMP	UNICOS
CRAY2	Cray-2	
CRAYSMP	Cray S-MP	
DGAV	Data General Aviion	
E88K	Encore 88000	
HP300	HP-9000 model 300	HPUX
HPPA	HP-9000 PA-RISC	
I860	Intel iPSC/860	link -lrpc
IPSC2	Intel iPSC/2 386 host	SysV
KSR1	Kendall Square KSR-1	OSF-1
LINUX	80386/486 LINUX box	LINUX
MASPAR	MASPAR host	
MIPS	MIPS 4680	
NEXT	NeXT	
PGON	Intel Paragon	link -lrpc
PMAX	DECstation 3100, 5100	Ultrix
RS6K	IBM/RS6000	AIX
RT	IBM RT	
SGI	Silicon Graphics	IRIX 4.x
SGI5	Silicon Graphics	IRIX 5.1
SUN3	Sun 3	SunOS 4.2
SUN4	Sun 4, SPARCstation	SunOS 4.2
SUN4SOL2	Sun 4, SPARCstation	Solaris 2.2
SYMM	Sequent Symmetry	
U370	IBM 370	AIX
UVAX	DEC MicroVAX	

Tavola 3.2: Architetture per le quali è stato sviluppato PVM 3.3[87].

PVM è composto di due parti con funzioni diverse. La prima parte è un programma chiamato *pvm* che viene eseguito su tutti gli ospiti che fanno parte della macchina virtuale e che si occupa di coordinare e gestire la comunicazione fra i vari processori (per l'effettivo utilizzo di PVM rimandiamo all'esauriente manuale relativo[87]). La seconda parte è una libreria di funzioni d'interfaccia fra PVM e i programmi dell'utente. Tutti i programmi che vogliono usare PVM devono far riferimento a questa libreria. La libreria PVM è una libreria software che può essere vista sia come estensione del FORTRAN che del C. Essa contiene tutte le funzioni o *routine* che l'utente può usare per gestire la macchina virtuale²:

1. routine per modificare la configurazione della macchina virtuale aggiungendo o eliminando (*pvm-addhost*, *pvm-delhost*) ospiti;
2. routine per accedere alla macchina virtuale (*pvm-myid*) e uscirne (*pvm-exit*);
3. routine per generare (*pvm-spawn*) e terminare (*pvm-kill*) task sulla macchina virtuale attraverso la rete;
4. routine per avere informazioni sui task attivi (*pvm-pstat*), sullo stato di un ospite (*pvm-mstat*), sulla configurazione della macchina virtuale (*pvm-config*), ecc.;
5. routine per la comunicazione e la sincronizzazione fra i task: routine per creare il *buffer*³ di spedizione (*pvm-init**send*), per compattare (*pvm-pack*) e spedire (*pvm-send*, *pvm-mcast*) messaggi, per riceverli (*pvm-nrecv*, *pvm-recv*) e ancora per espanderli (*pvm-unpack*), per verificare se il messaggio richiesto è arrivato (*pvm-probe*) e per avere informazioni su un messaggio che si trova in un buffer specificato (*pvm-bufinfo*).
6. routine per segnalare errori (*pvm-perror*, *pvm-serror*).

Tutte le routine PVM restituiscono una condizione d'errore se qualche errore è stato rilevato durante l'esecuzione.

²Ogni routine ha bisogno di vari parametri relativi alla funzione che la particolare routine deve svolgere, ma che omettiamo per semplicità.

³Con il termine *buffer* si intende l'area di memoria della macchina in cui vengono messi i messaggi pronti per la spedizione o per la ricezione.

I programmi che vengono sviluppati per l'utilizzo con PVM sono formati da vari task o processi che vengono distribuiti fra i vari ospiti. A qualsiasi punto dell'esecuzione di un programma i processi possono avere relazioni, comunicare e sincronizzarsi fra loro. Qualsiasi task può spedire uno o più messaggi a qualsiasi altro task; PVM garantisce che i messaggi vengono ricevuti nello stesso ordine con cui sono stati spediti. Il modello di comunicazione è formato da una routine di spedizione asincrona (**pvm-send**), da una di ricezione asincrona bloccante (**pvm-recv**), e da una di ricezione non bloccante (**pvm-nrecv**). Una volta spedito un messaggio tramite la routine di spedizione asincrona, questa si può riutilizzare appena il buffer di spedizione è libero (non appena il messaggio è stato spedito) indipendentemente dallo stato del ricevente. Se per ricevere un messaggio si usa la routine di ricezione asincrona bloccante, questa può essere riutilizzata solo dopo che il messaggio richiesto è arrivato, se invece utilizziamo la routine di ricezione non bloccante, essa è riutilizzabile immediatamente fornendoci il messaggio desiderato se è arrivato, o informandoci che questo non è ancora giunto. Inoltre, le routine di ricezione possono essere predisposte per accettare messaggi di tipo specificato o di qualunque tipo. In aggiunta a questo tipo di comunicazione punto a punto e cioè solamente fra due o più processi specifici, esiste una routine che permette di spedire lo stesso messaggio a tutti i task (**pvm-mcast**).

Il modo di organizzare i programmi che è stato utilizzato in questo lavoro è il modello di programma “*master/slave*” in cui la ripartizione dei compiti è asimmetrica e cioè è diversa a seconda del task: esistono un task detto *master* e più task detti *slave*. Il processo *master* è quello principale; esso ha svariati compiti: organizza e controlla il lavoro, esegue la parte prettamente seriale del calcolo, genera e dirige un certo numero di processi *slave* che eseguono la parte parallela del calcolo sui dati forniti dal *master*, esegue dei controlli per sapere se gli *slave* stanno eseguendo in modo corretto il loro compito, elimina i processi *slave* quando questi hanno terminato il loro compito o in caso di errore, richiede agli *slave* i risultati parziali o finali dei calcoli che poi riorganizza per fornire il risultato richiesto.

3.7 VMC e DMC sulla macchina virtuale

Il metodo Monte Carlo Variazionale e il metodo Monte Carlo di Diffusione, come abbiamo visto rispettivamente nelle Sez. 2.2, 2.3 e 2.4, sono formati da una parte di equilibratura e da una parte di media immediatamente successiva. Data una popolazione iniziale di N configurazioni, durante la fase di equilibratura esse evolvono per M passi per raggiungere il campionamento desiderato; durante la fase di media esse si spostano per altri \bar{M} passi, campionando sempre la distribuzione di equilibrio in modo tale da ridurre l'errore statistico nel calcolo dei valori medi (vedere Sez. 2.4). Se P è il numero di processori disponibili esistono due modi distinti per parallelizzare il nostro problema:

1. il **metodo del sistema distribuito** che consiste nel dividere la popolazione in parti uguali da distribuire su ogni processore e a cui far compiere l'intero numero di passi;
2. il **metodo delle repliche** che consiste nel distribuire una copia dell'intera popolazione su ogni processore e a cui far compiere solo una parte dell'intero cammino in modo che la somma del numero di passi eseguiti da ogni processore sia uguale al numero totale di passi richiesti.

In questo lavoro sono stati modificati i programmi sequenziali esistenti per il VMC e il DMC nella versione di Umrigar in modo tale da poter essere utilizzati con PVM ed eseguiti in parallelo su una rete di calcolatori. Sia per il VMC che per il DMC si è sviluppato un programma master/slave, con due processi *slave* diversi: uno per l'equilibratura (che abbiamo chiamato *slaves*), ed uno per la media (che abbiamo chiamato *slavem*). Poiché la tecnica usata per parallelizzare i due programmi è simile, nella prossima sezione descriveremo il procedimento seguito nel caso particolare del DMC.

3.7.1 Parallelizzazione del DMC

Il programma DMC che abbiamo parallelizzato è quello nella versione di Umrigar e d'ora in poi, ogni volta che parleremo di DMC, ci riferiamo a quest'ultimo (vedere Sez. 2.3.4). Descriviamo ora l'organizzazione generale del programma master/slave che abbiamo realizzato

e vediamo quali sono le differenze significative con il programma sequenziale.

1. Il processo *master*, nello stesso modo del programma seriale, legge i dati iniziali dai file e, a partire da questi, prepara i dati da inviare agli *slaves* (Sez. 2.3.2,2.3.4,2.4).

È importante sottolineare che in generale, per comodità ed economia dello spazio disco, solo uno degli ospiti della macchina virtuale può accedere ai file di dati e quindi è su questo ospite che viene fatto eseguire il processo *master*. Inoltre, le operazioni iniziali appena descritte rappresentano una frazione trascurabile del tempo di esecuzione e non avrebbe senso eseguirle in parallelo.

2. Il *master* legge da un file il numero P di processi di equilibratura (*slaves*) che si vogliono utilizzare, con il nome degli ospiti su cui gli *slaves* devono essere eseguiti, e genera i relativi P processi *slaves* per eseguire in modo parallelo la fase di equilibratura. Poiché nella fase di equilibratura la popolazione deve eseguire tutti gli M passi per raggiungere la distribuzione di equilibrio è indispensabile procedere mediante il **metodo del sistema distribuito** e ricominciare dall'inizio se uno o più *slaves* dovessero terminare in anticipo a causa di qualche errore.

3. Il *master* spedisce a ognuno dei P *slaves* un numero di walker pari a N/P assieme a tutti i dati necessari⁴ per iniziare la fase di equilibratura (Sez. 2.3.2,2.3.4,2.4).

4. Ogni *slaves* attende finché i dati iniziali non sono arrivati (dunque li riceve in maniera bloccante), e poi spedisce al *master* un messaggio che comunica se la ricezione è avvenuta in modo corretto o meno. Si usa questo tipo di ricezione poiché gli *slaves* non possono procedere nei calcoli senza i dati iniziali provenienti dal *master*. I dati iniziali vengono successivamente organizzati dagli *slaves* che iniziano la fase di equilibratura. Sia nel programma seriale che in quello parallelo, gli M passi di equilibratura vengono divisi in B blocchi di M_B passi ciascuno (Sez. 2.5).

⁴È importante sottolineare che sia nella fase di equilibratura che di media ogni processo *slaves* e *slavem*, rispettivamente, ha utilizzato numeri casuali differenti per evitare di far compiere lo stesso cammino a walker diversi.

5. Il *master* attende la conferma dell'avvenuta ricezione in maniera non bloccante; se dopo un certo numero di tentativi la conferma non è ancora giunta, o se arriva un messaggio d'errore, il *master* termina tutti i processi *slaves* e termina l'esecuzione con un messaggio d'errore. In tal caso è necessario ricominciare dall'inizio. La ricezione da parte del *master* avviene in maniera non bloccante, in quanto altrimenti rischieremmo di attendere all'infinito se qualche risposta non arrivasse (*deadlock*).
6. Durante l'equilibratura ogni *slaves* fa evolvere i suoi N/P walker (nel programma sequenziale evolvono tutti N assieme) M volte secondo la dinamica data dal processo di deriva [Eq. (2.93)] e di diffusione [Eq. (2.94)] subordinati alla probabilità di transizione [Eq. (2.95)] che decide se accettare o meno il movimento (Sez. 2.3.2).
7. A ogni passo viene aggiornato il peso di ogni configurazione secondo il processo di nascita-morte [Eq. (2.114)] e mediante il processo di split e join applicato separatamente alla popolazione presente su ogni nodo (nel caso sequenziale lo split e join è applicato a tutta la popolazione contemporaneamente) si evitano fluttuazioni incontrollate dei pesi associati ai singoli individui (come descritto nella Sez. 2.3.4).
8. Nel programma seriale, alla fine di ogni passo (punti 6 e 7), si aggiorna l'energia di prova E_T secondo la formula data dall'Eq. (2.112) utilizzando l'Eq. (2.123) per E_{cum} , per mantenere approssimativamente costante il peso totale della popolazione (vedere Sez. 2.3.4). Nel programma parallelo l'aggiornamento dell'energia di prova E_T ad ogni passo implicherebbe una grossa comunicazione e problemi di sincronizzazione (e dunque una diminuzione di efficienza), dato che per il calcolo di E_T il *master* ha bisogno dei dati relativi a tutta la popolazione che è divisa fra i vari processori. Per questo, alla fine di ogni passo, e cioè ogni volta che tutta la sua popolazione di N/P individui evolve secondo deriva, diffusione e nascita-morte, ogni *slaves* dovrebbe spedire al *master* i dati necessari per aggiornare E_T e mettersi in attesa (sincronizzazione). Il *master*, una volta ricevuti i dati del corrispondente passo da tutti gli *slaves*, dovrebbe calcolare la nuova energia di prova e comunicarla a tutti gli *slaves* che così potrebbero proseguire

con il passo successivo. Per evitare lunghe attese dovute alla continua comunicazione, si è fatto in modo di aggiornare l'energia di prova E_T dopo ogni blocco di M_B passi anziché dopo ogni singolo passo, e poichè si è sperimentato che così si potrebbe avere un'esplosione o un'estinzione della popolazione (dal punto di vista del calcolatore) in quanto il peso totale viene normalizzato solo all'inizio del blocco, si è ampliato il processo di split e join rispetto al programma seriale. Lo scopo è quello di mantenere la popolazione su ogni processore sempre normalizzata a N/P , lasciandone inalterato il peso totale: se, dopo aver eseguito le operazioni descritte nel punto 7, il numero di configurazioni su un certo nodo è aumentato si cercano le coppie con peso più piccolo su quel nodo (e non quelle con peso inferiore a w_{min} come nel join tradizionale) e si “uniscono” come descritto dall'Eq. (2.113); se è diminuito, si cercano i walker con peso più grande (e non quelli con peso superiore a w_{max} come nello split tradizionale) e se ne crea una copia sempre nel modo descritto nella Sez. 2.3.4. Il fatto di mantenere costante il numero di individui su ogni nodo favorisce il bilanciamento del carico dei vari processori, se questi hanno la stessa velocità. Alla fine di ogni blocco ogni *slaves* spedisce al *master* i dati necessari per aggiornare E_T (ricezione bloccante). A questo punto il processo *master* possiede tutti i dati relativi al blocco considerato, aggiorna E_T tramite le stesse formule utilizzate nel programma seriale e, se il totale dei passi compiuti sino a quel momento dalla popolazione è più piccolo del valore stabilito M , spedisce a tutti gli *slaves* il nuovo valore di E_T e l'autorizzazione a procedere con un nuovo blocco di passi. Tutti gli *slaves*, dopo aver ricevuto il nuovo valore di E_T (ricezione bloccante), proseguono con il blocco successivo.

9. Ogni M_1 passi (vedere Sez. 2.3.4) viene aggiornato il valore di Δt_{eff} tramite l'Eq. (2.106). Nel programma parallelo questa operazione viene eseguita dal processo *master* dopo aver ricevuto in modo bloccante i dati necessari da tutti gli *slaves*.
10. Alla fine degli M passi il *master* riceve da tutti gli *slaves*, in modo bloccante, i dati finali e la configurazione finale di walker. Gli *slaves* terminano l'esecuzione e il *master*

procede al calcolo delle quantità necessarie per la fase di media (Sez. 2.3.2,2.3.4,2.4).

11. Per la fase di media, come nel punto 2, il *master* genera un numero stabilito di processi *slave*, questa volta chiamati *slavem*. Poichè una volta all'equilibrio abbiamo bisogno di \bar{M} campionamenti indipendenti della distribuzione asintotica possiamo procedere con il **metodo delle repliche** che è più efficiente del metodo del sistema distribuito poichè non si hanno problemi di sincronizzazione.
12. Il *master* spedisce a ogni *slavem* una copia dell'intera popolazione di N walker assieme a tutti gli altri dati necessari per la fase di media (vedere nota 4 e Sez. 2.3.2,2.3.4,2.4).
13. Gli *slavem* procedono come gli *slaves* nel punto 4 per i passi di media. A causa della correlazione fra passi successivi i campionamenti indipendenti dell'energia sono \bar{M}/M_B e cioè pari al numero B di blocchi. Poiché $M_B \gg T_{corr}$ (Sez. 2.5) i valori dell'energia di blocco non sono correlati e cioè il blocco è sufficientemente lungo da permettere il corretto campionamento dello spazio e da eliminare la dipendenza dalla particolare popolazione iniziale che è la stessa per tutti gli *slavem*.
14. Durante la fase di media tutto procede in modo simile alla fase di equilibratura ma con le seguenti differenze:
 - (a) poiché abbiamo bisogno di B campionamenti indipendenti dell'energia (di blocco) è sufficiente che ogni *slavem* esegua solo una parte (arbitraria) dei blocchi necessari e che alla fine di ogni blocco avverta il *master*; una volta che i blocchi eseguiti da tutti gli *slavem* raggiungono il totale B sarà il *master* a segnalare la fine dell'esecuzione. In questo modo tutti gli *slavem* eseguono un numero di blocchi proporzionale alla loro velocità relativa (dovuta alle caratteristiche e al carico del processore) e terminano approssimativamente assieme dando luogo a un bilanciamento ottimale del carico.
 - (b) ogni processo *slavem* possiede l'intera popolazione ed è in grado di aggiornare esso stesso, senza alcun problema di sincronizzazione, l'energia di prova E_T a ogni passo come avviene nel programma seriale. Per questo non si ha bisogno di rinormalizzare

la popolazione a ogni passo come nella fase di equilibratura e il processo di split e join rimane uguale a quello del programma seriale.

(c) Dato che non è necessario procedere alla sincronizzazione alla fine di ogni blocco ogni *slavem* procede immediatamente con il blocco successivo.

(d) Come nel programma seriale il valore di Δt_{eff} non viene aggiornato periodicamente ma viene posto uguale al valore calcolato alla fine della fase di equilibratura.

(e) La fase di media ha un alto livello di tolleranza agli errori. Come già scritto, ogni *slavem* possiede l'intera popolazione di N walker, e se uno o più *slavem* interrompe in anticipo l'esecuzione non è necessario ricominciare tutto dall'inizio (come invece si dovrebbe fare nella fase di equilibratura se si interrompesse l'esecuzione di uno *slaves*) ma è sufficiente che con gli *slavem* attivi si raggiunga il numero totale \bar{M} di passi desiderati. È dunque opportuno che ogni *slavem*, alla fine di ogni blocco di media, spedisca al *master* i dati relativi al blocco appena terminato in modo da non perdere i risultati da esso ottenuti sino a quel momento in caso di interruzione. Perciò, periodicamente e in maniera non bloccante, il *master* controlla se uno qualunque degli *slavem* ha spedito i dati relativi ad un blocco di media e, in caso affermativo, effettua una ricezione. Se il *master*, dopo un certo numero di controlli, non riceve nulla, blocca tutti gli *slavem* e ne genera di nuovi.

15. Una volta che sono stati fatti tutti gli \bar{M} passi, il *master* calcola le grandezze finali quali il valor medio dell'energia tramite l'Eq. (2.121), l'incertezza statistica tramite la varianza data dall'Eq. (2.127), il tempo di correlazione, Eq. (2.128), e sia il *master* che gli *slavem* terminano l'esecuzione (Sez. 2.4 e 2.5).

3.7.2 Parallelizzazione del VMC

Come già detto, il programma VMC parallelo è strutturato in maniera analoga al programma DMC parallelo. La differenza sta nel fatto che, mentre nel DMC viene aggiornata l'energia di prova E_T , nel VMC aggiorniamo il valore di λ per garantire la massima efficienza di

campionamento (vedere Sez. 2.2).

3.8 Risultati per sistemi a due elettroni

Questa e le prossime sezioni sono dedicate alla presentazione dei risultati ottenuti. In particolare analizzeremo in dettaglio i risultati ottenuti con i programmi VMC e DMC paralleli su una rete di calcolatori e li confronteremo con quelli ottenuti con i programmi seriali per verificarne l'equivalenza (Sez. 3.8.1, 3.8.2 e 3.8.3); paragoneremo i tempi impiegati per valutare se i programmi paralleli sono più o meno convenienti di quelli sequenziali (Sez. 3.8.4); analizzeremo passo per passo il programma DMC parallelo e studieremo la dipendenza dei tempi di calcolo dal tipo di rete (Sez. 3.8.5); infine confronteremo i nostri tempi con quelli impiegati da un calcolatore parallelo massivo, APE100[14] (Sez. 3.8.6). Il sistema considerato è lo stesso del capitolo precedente (vedere Sez. 2.6 e 2.6.1); i risultati ottenuti, a meno che non vi siano ulteriori specificazioni, sono espressi in unità atomiche e sono relativi ai parametri iniziali riportati nella Sez. 2.6. Nel caso del programma parallelo, riferendoci a un numero P di processori, sarà sempre sottinteso che consideriamo P task di tipo *slave* (di equilibratura o di media) uno per processore. Il task *master* è eseguito su uno dei P processori, contemporaneamente a un task *slave*, in quanto il suo onere computazionale è trascurabile.

3.8.1 Risultati delle simulazioni VMC

L'equivalenza dei programmi VMC sequenziale e parallelo si è verificata confrontando i risultati ottenuti per il valor medio della funzione d'onda di prova Ψ_T , Eq. (2.134), sull'hamiltoniano \hat{H} , Eq. (2.133), per tutti gli elementi chimici del terzo periodo della Tavola Periodica. Come si vede dalla Tab. 3.3, le energie coincidono entro le barre d'errore e l'equivalenza risulta provata. Nel caso del Si^{2+} abbiamo considerato in dettaglio anche i tempi di esecuzione per i due programmi. Nella Tab. 3.4 sono elencati i tempi di esecuzione per la fase di equilibratura (T_e), di media (T_m), ed il tempo totale (T) per il programma seriale e per quello parallelo, con l'utilizzo di un numero di processori $P = 1$ e $P = 4$. I

atomo o ione	Z	b (Bohr $^{-1}$)	\bar{E} (Ha)	\bar{E}^{PVM} (Ha)
Na $^{-}$	11	0.105	-0.211612(15)	-0.211589(15)
Mg	12	0.155	-0.840111(15)	-0.840109(15)
Al $^{+}$	13	0.195	-1.741988(26)	-1.741953(25)
Si $^{2+}$	14	0.2325	-2.886342(31)	-2.886353(33)
P $^{3+}$	15	0.28875	-4.263398(37)	-4.263369(38)
S $^{4+}$	16	0.345	-5.869896(47)	-5.869876(50)
Cl $^{5+}$	17	0.35125	-7.719827(57)	-7.719844(53)
Ar $^{6+}$	18	0.42	-9.76278(10)	-9.762679(77)

Tavola 3.3: Energie variazionali di stato fondamentale per tutti gli atomi (ioni) del III periodo in cui sono stati considerati solo due elettroni di valenza. Z è il numero atomico, b è il parametro variazionale nel fattore di Jastrow, \bar{E} l'energia variazionale calcolata con il programma sequenziale, \bar{E}^{PVM} l'energia variazionale calcolata con il programma parallelo (vedere testo). I valori delle energie \bar{E} e i diversi parametri b utilizzati sono quelli ottenuti da Menchi *et al.*[11, 88].

programma	$T_e(s)$	$T_m(s)$	$T(s)$
sequenziale	303	3917	4220
parallelo ($P=1$)	400	3949	4349
parallelo ($P=4$)	117	1003	1120

Tavola 3.4: Tempi di calcolo per il programma VMC sequenziale e parallelo con un numero di processori $P = 1$ e $P = 4$ nel caso del Si $^{2+}$ (vedere testo).

calcolatori utilizzati sono delle Sun SPARCstation 10 modello 512 e la velocità di ogni processore è di circa 20 Mflop⁵. I tempi di esecuzione riportati qui e nel seguito si riferiscono all'esecuzione in modo dedicato e cioè con ogni processore completamente dedicato al calcolo VMC o DMC. Osserviamo, come previsto sulla base delle considerazioni fatte nella Sez. 3.2, che il programma parallelo per $P = 1$ è più oneroso di quello sequenziale, specialmente nella fase di equilibratura: complessivamente, però, l'onere computazionale aggiuntivo è solo del 3%. Per quanto riguarda il confronto fra $P = 1$ e $P = 4$ nel caso del programma parallelo, osserviamo che i risultati sono più che soddisfacenti: quadruplicando il numero di processori otteniamo un tempo di calcolo più piccolo di un fattore 3.9.

⁵1 Mflop equivale a 10^6 operazioni in virgola mobile per secondo.

Δt	\bar{E}_0 (Ha)	\bar{E}_0^{PVM} (Ha)
0.01	-2.88947(11)	-2.88936(11)
0.05	-2.889557(56)	-2.889543(61)
0.1	-2.889469(43)	-2.889514(41)
0.3	-2.889351(26)	-2.889366(25)
0.5	-2.889455(19)	-2.889419(26)
0.7	-2.889807(20)	-2.889762(21)
0.85	-2.890328(20)	-2.890288(22)
1.0	-2.891112(23)	-2.891066(24)
0.0	-2.889491(32)	-2.889518(33)

Tavola 3.5: Energie di stato fondamentale per lo ione Si^{2+} calcolate mediante il DMC nella versione di Umrigar in funzione del passo temporale Δt . \bar{E}_0 è l'energia calcolata con il programma sequenziale, \bar{E}_0^{PVM} l'energia calcolata in parallelo (vedere testo). Il valore corrispondente al passo $\Delta t = 0$ è ottenuto per estrapolazione come spiegato nel testo. I numeri in parentesi sono gli errori statistici e si riferiscono alle ultime cifre.

3.8.2 Risultati delle simulazioni DMC

Una simulazione DMC, a causa dell'approssimazione a tempi corti, è composta di vari calcoli in cui si utilizzano passi temporali Δt differenti. Alla fine del calcolo abbiamo differenti valori dell'energia corrispondenti ai diversi passi Δt e possiamo eseguire un'interpolazione (mediante il metodo dei minimi quadrati come descritto nella Sez. 2.4) e contemporaneamente un'estrapolazione a $\Delta t = 0$: il risultato è la stima dell'energia di stato fondamentale corretta dall'errore introdotto dall'approssimazione a tempi corti. Per essere certi del corretto funzionamento del programma parallelo, nel caso dello ione Si^{2+} , abbiamo confrontato sia l'estrapolazione finale che le energie ottenute per i diversi Δt con i corrispondenti risultati forniti dal programma sequenziale. Come si vede nella Tab. 3.5 e nella Fig. 3.1, i risultati “paralleli” coincidono con quelli “sequenziali” entro l'errore statistico e lo stesso vale per le due curve di interpolazione, confermando la perfetta equivalenza fra i relativi programmi. Nella didascalia della Fig. 3.1 sono riportate le equazioni delle curve di interpolazione (Sez. 2.4), i valori ottenuti per i parametri, così come il valore del χ^2 che misura la bontà delle curve di interpolazione.

Come nel caso del VMC consideriamo i tempi di esecuzione riportati in Tab. 3.6 per

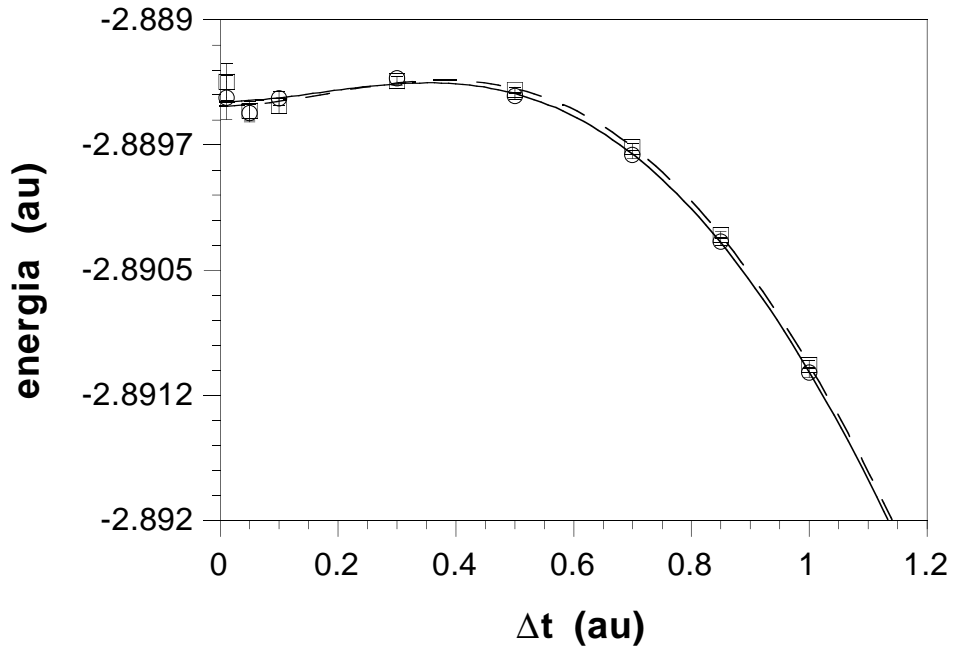


Figura 3.1: Energie DMC nella versione di Umrigar per lo ione Si^{2+} in funzione di Δt . I punti (○) e la curva d'interpolazione continua $\alpha + \beta\Delta t^2 + \gamma\Delta t^3 + \delta\Delta t^4$ si riferiscono ai risultati ottenuti con il programma seriale. I valori calcolati per i parametri sono $\alpha = -2.889491(32)$, $\beta = 0.00292(65)$, $\gamma = -0.0064(14)$, $\delta = 0.00184(81)$ con l'incertezza in parentesi. Il corrispondente valore di χ^2 è 4.12. Il valore dell'energia estrapolato a $\Delta t = 0$ è $\bar{E}_0 = -2.889491(32)$ e questa è la stima DMC per l'energia di stato fondamentale. I punti (□) e la curva d'interpolazione tratteggiata $\alpha_1 + \beta_1\Delta t^2 + \gamma_1\Delta t^3 + \delta_1\Delta t^4$ si riferiscono ai risultati ottenuti con il programma parallelo: $\alpha_1 = -2.889518(33)$, $\beta_1 = 0.00353(73)$, $\gamma_1 = -0.0073(16)$, $\delta_1 = 0.00224(92)$, $\chi^2 = 2.93$ e $\bar{E}_0 = -2.889518(33)$.

programma	T_e (s)	T_m (s)	T (s)
seriale	1400	6981	8381
parallelo ($P=1$)	1499	7010	8509
parallelo ($P=4$)	477	1783	2260

Tavola 3.6: Tempi di calcolo per il programma DMC nella versione di Umrigar sequenziale e parallelo per $P = 1$ e $P = 4$ processori, nel caso del Si^{2+} con passo temporale $\Delta t = 0.85$ (vedere testo).

il Si^{2+} . T_e è il tempo impiegato per eseguire tutta la fase di equilibratura, T_m quello relativo alla fase di media e T il tempo totale per il programma seriale e quello parallelo con l'utilizzo di un numero di processori $P = 1$ e $P = 4$. I calcolatori utilizzati sono gli stessi della sezione precedente. Di nuovo osserviamo che il programma parallelo con $P = 1$ è meno efficiente di quello sequenziale specialmente nella fase di equilibratura, anche se il sovraccarico complessivo è solo del 1.5%. Nel caso del programma parallelo eseguito con 1 e 4 processori, anche per il DMC i risultati sono molto buoni: il tempo di esecuzione si riduce di un fattore 3.8 passando da $P = 1$ a $P = 4$. Un'analisi più approfondita della dipendenza del tempo di calcolo dal numero di processori nel caso del DMC è presentata nella Sez. 3.8.4.

I risultati presentati in questa sezione si riferiscono a simulazioni DMC realizzate esclusivamente mediante l'algoritmo di Umrigar. Nella prossima sezione confronteremo invece i risultati per un certo numero di diversi elementi ottenuti da Menchi *et al.*[11, 88] mediante l'algoritmo di Reynolds (programma sequenziale) con quelli da noi ottenuti mediante l'algoritmo di Umrigar (programma parallelo), per vedere quali sono le differenze fra i due algoritmi.

3.8.3 Confronto fra diversi DMC

Nel caso di tutti gli atomi e gli ioni con due elettroni di valenza del terzo periodo della Tavola Periodica, abbiamo messo a confronto i due algoritmi per il DMC e cioè quello di Reynolds *et al.*[4] (risultati ottenuti da Menchi *et al.*[11, 88] con un programma sequenziale)

atomo o ione	\bar{E}_0 (Reynolds)	\bar{E}_0 (Umrigar)
Na ⁻	-0.2156833(91)	-0.215675(25)
Mg	-0.8421513(85)	-0.842127(17)
Al ⁺	-1.744489(15)	-1.744461(21)
Si ²⁺	-2.889503(15)	-2.889518(33)
P ³⁺	-4.267741(18)	-4.267698(41)
S ⁴⁺	-5.872117(37)	-5.875085(33)
Cl ⁵⁺	-7.724793(17)	-7.724851(54)
Ar ⁶⁺	-9.768896(45)	-9.768735(66)

Tavola 3.7: Confronto fra i valori dell'energia di stato fondamentale calcolati con gli algoritmi di Reynolds e di Umrigar (vedere testo).

e quello di Umrigar *et al.*[70]. Per quanto riguarda la versione di Umrigar, si è usato il programma parallelo che, come abbiamo verificato nella precedente sezione, è equivalente a quello seriale. In Fig. 3.2 e 3.3 sono riportate le energie \bar{E}_0 in funzione di Δt per entrambi i metodi, assieme alle curve di interpolazione. Notiamo che i due metodi, per valori finiti di Δt , a causa della diversità degli algoritmi, danno origine a due curve differenti la cui estrapolazione a zero (vedere Tab. 3.7) coincide entro gli errori statistici: questo dimostra che i due algoritmi sono equivalenti. Più nel dettaglio si può osservare che per grandi valori di Δt , la curva relativa all'algoritmo di Umrigar è sempre più vicina al risultato estrapolato a $\Delta t = 0$ di quella relativa all'algoritmo di Reynolds (la prima si trova sempre fra la seconda e la retta orizzontale che passa per il valore estrapolato). Inoltre, la curva di Reynolds è sempre sotto al valore corretto, quella di Umrigar invece, al crescere del passo temporale Δt (partendo da $\Delta t = 0$), si trova prima un po' sopra al valore esatto, poi incomincia a scendere, lo interseca e infine sta sotto.

Secondo gli autori, il metodo di Umrigar dovrebbe avere minori fluttuazioni statistiche rispetto a quello di Reynolds, perché diversamente da quest'ultimo non interviene direttamente sulla dimensione della popolazione con il processo di nascita-morte, sopprimendo o creando dei walker, ma interviene sui pesi (vedere Sez. 2.3.4); inoltre il valore di Δt_{eff} viene mantenuto costante durante la fase di media. In realtà, probabilmente a causa della sem-

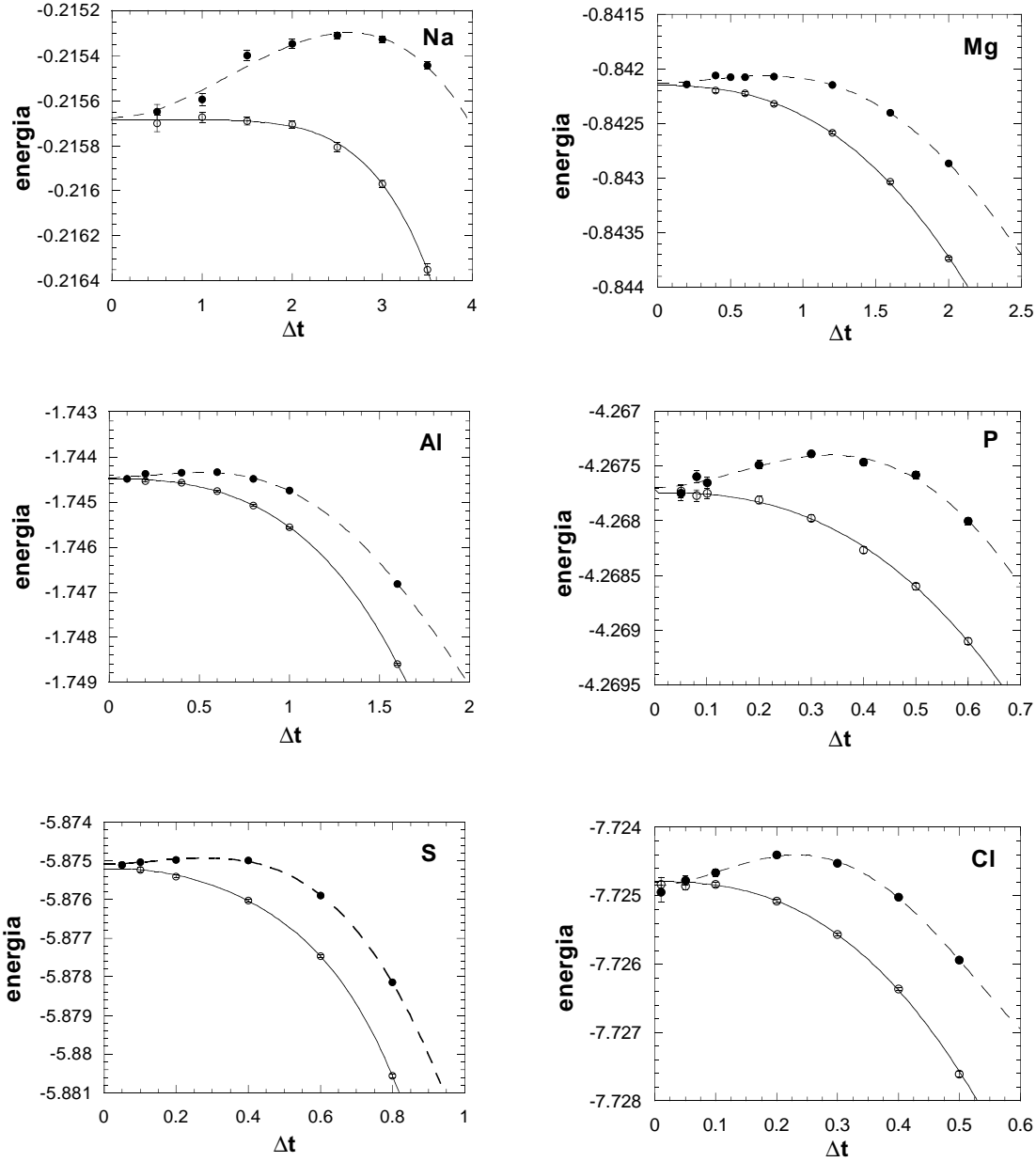


Figura 3.2: Energie DMC per gli ioni Na^- , Mg , Al^+ , P^{3+} , S^{4+} e Cl^{5+} in funzione di Δt . I punti (\circ) e la curva d'interpolazione continua si riferiscono ai risultati ottenuti con l'algoritmo di Reynolds. I punti (\bullet) e la curva d'interpolazione tratteggiata si riferiscono ai risultati ottenuti con l'algoritmo di Umrigar. Il valore dell'energia estrapolato a $\Delta t = 0$ è riportato in Tab. 3.7. Le unità di misura sono unità atomiche.

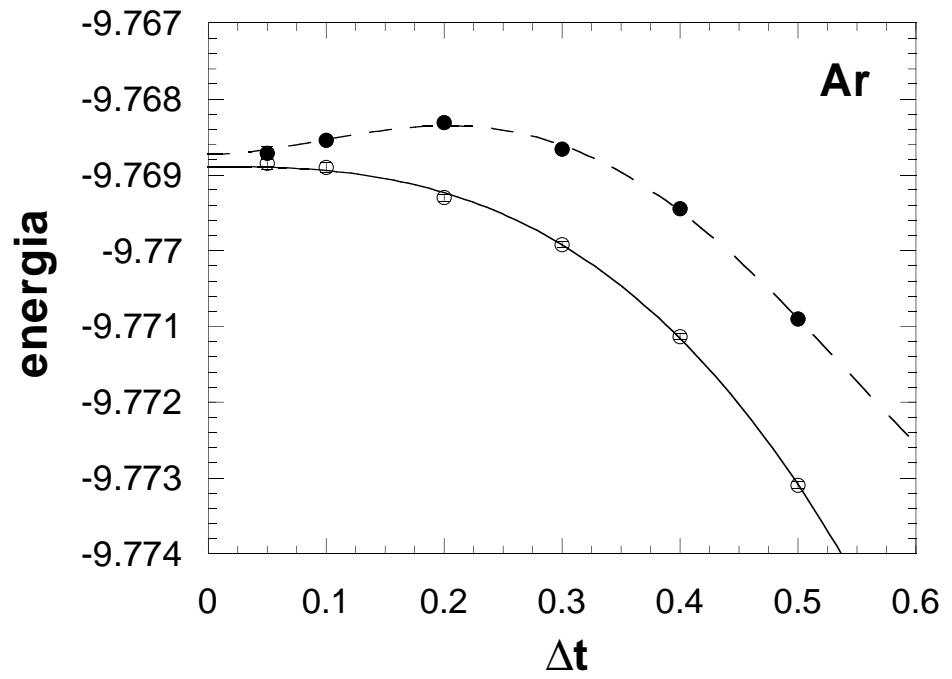


Figura 3.3: Energie DMC per lo ione Ar^{6+} in funzione di Δt . I punti (○) e la curva d'interpolazione continua si riferiscono ai risultati ottenuti con l'algoritmo di Reynolds. I punti (●) e la curva d'interpolazione tratteggiata si riferiscono ai risultati ottenuti con l'algoritmo di Umrigar. Il valore dell'energia estrapolato a $\Delta t = 0$ è riportato in Tab. 3.7. Le unità di misura sono unità atomiche.

plicità del nostro sistema a due elettroni, a parità di parametri di esecuzione, le incertezze statistiche risultano sostanzialmente le stesse per entrambe le tecniche. Nell'algoritmo di Umrigar abbiamo sperimentato che il valore dell'energia, calcolato tramite il metodo di correzione dell'errore sistematico dovuto al meccanismo di controllo della popolazione (vedere Sez. 2.4), non dipende, entro la statistica, dal valore di L : possiamo presumere che tale errore sia molto più piccolo di quello statistico e dunque invisibile, anche in questo caso per la semplicità del sistema trattato. Infatti, se la funzione d'onda di prova è sufficientemente buona (e questo di solito è il caso per un sistema semplice come il nostro), le fluttuazioni dell'energia locale sono piccole. Di conseguenza anche le fluttuazioni nel peso della popolazione sono piccole, e lo stesso vale per la correzione a E_T data dall'ultimo termine a destra nell'Eq. (2.112) che introduce un errore sistematico trascurabile. Per quanto riguarda la curva di interpolazione delle energie ottenute con l'algoritmo di Umrigar, per i sistemi da noi considerati è stato sufficiente scegliere un polinomio in potenze intere di Δt a partire da Δt^2 (e fino a Δt^4)[13], invece dell'andamento più generale discusso alla fine della Sez. 2.4 e in accordo con l'andamento quadratico osservato da Umrigar *et al.*[70]. L'interpolazione delle energie ottenute con l'algoritmo di Reynolds è stata invece effettuata con un polinomio in potenze intere di $\sqrt{\Delta t}$ a partire da $(\sqrt{\Delta t})^5$ [88].

3.8.4 Tempi di calcolo ed efficienza

Per valutare quanto sia conveniente risolvere il nostro problema in parallelo anziché in modo seriale, al variare del numero di processori, abbiamo eseguito diverse volte lo stesso calcolo utilizzando un numero diverso di processori uguali e dedicati. Nel caso del DMC abbiamo impiegato prima un solo processore, poi due, tre e quattro. Così, in Fig. 3.4, si è costruito un grafico di T_1/T_P in funzione del numero P di processori, in cui T_1 è il tempo impiegato da un processore e T_P è il tempo impiegato da P processori per eseguire la fase di equilibratura, di media, ed entrambe. La curva migliore che si può ottenere interpolando i punti è, nel limite teorico, una retta passante per l'origine e pendenza unitaria. In pratica, dalla Fig. 3.4 vediamo che la retta teorica (linea continua) interpola solo i punti relativi alla fase di media

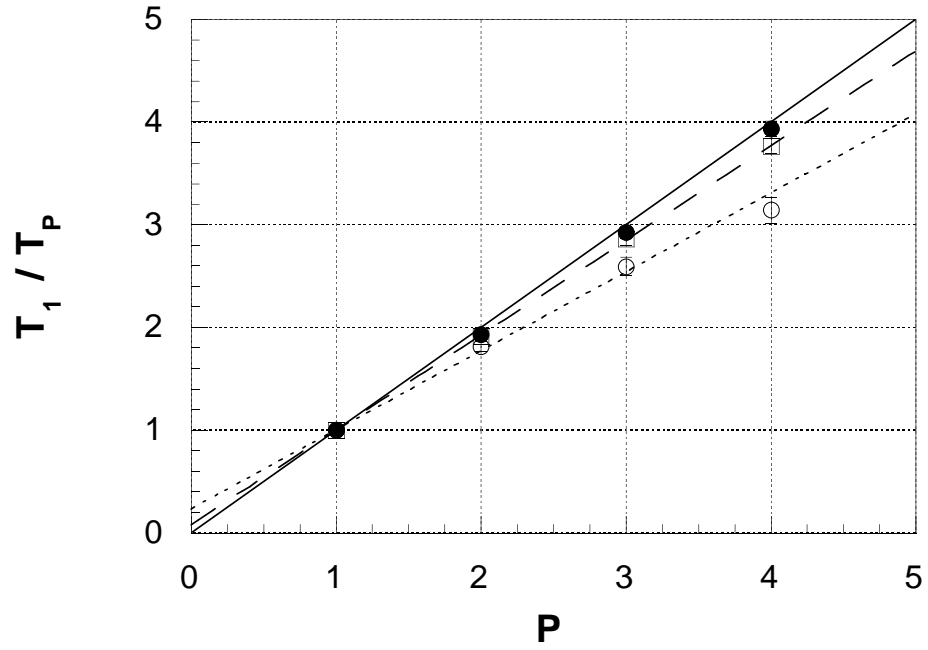


Figura 3.4: Inverso dei tempi di calcolo normalizzati a T_1 in funzione del numero di processori. Il limite teorico è identificato dalla retta in tratto continuo che passa per l'origine e ha equazione $T_1 = P T_P$. I punti (●) sono relativi alla fase di media. I punti (○) e la retta d'interpolazione $\alpha + (1 - \alpha)P$ (tratteggio corto) si riferiscono alla fase di equilibratura. Il valore calcolato per il parametro è $\alpha = 0.231(26)$ con l'incertezza in parentesi. La frazione di tempo speso in calcolo seriale è approssimativamente il 23%. I punti (□) e la curva d'interpolazione $\alpha_1 + (1 - \alpha_1)P$ (tratteggio lungo) si riferiscono al tempo totale: $\alpha_1 = 0.078(17)$. In questo caso la frazione seriale è circa l'8% (vedere anche il testo). Le barre d'errore sono state stimate come discusso nel testo.

e infatti questa è la parte in cui si ha meno comunicazione e quindi più indipendenza fra i processi. Per quanto riguarda i punti relativi alla fase di equilibratura, abbiamo interpolato con una retta passante per il nostro punto di riferimento di coordinate $(1, 1)$ ottenendo, così, una retta che non passa per l'origine e che non ha pendenza unitaria. Questo è dovuto al fatto che nella fase di equilibratura i processi non sono indipendenti e si ha necessità di sincronizzazione. L'intercetta di questa retta con l'asse delle ordinate è una stima della frazione del tempo di equilibratura speso nell'eseguire operazioni in modo seriale e quindi una misura dell'efficienza del programma: più la frazione seriale è vicina a zero più elevata sarà l'efficienza. Infatti, dalla generica equazione

$$y = \alpha + \beta x \quad (3.4)$$

dove $y = T_1/T_P$ e $x = P$, imposto il passaggio per il punto di riferimento $(1, 1)$, si ottiene $\beta = 1 - \alpha$, da cui

$$\frac{T_1}{T_P} = \alpha + (1 - \alpha)P \quad (3.5)$$

che è proprio la legge di Gustafson, Eq. (3.3). Osserviamo che la frazione seriale α (rapporto fra il tempo speso in operazioni seriali e il tempo totale) è di circa il 23% e questo valore è compatibile con tutti i punti $P = 2, 3, 4$ (entro le barre d'errore, vedere più avanti come sono stimate). Il fatto che α sia approssimativamente indipendente da P (almeno per valori di P piccoli), in apparente contrasto con quanto detto nella Sez. 3.5, si spiega osservando qual'è l'origine della maggior parte delle operazioni seriali per P piccoli. Ogni task esegue un numero di operazioni che non è costante a causa della natura stocastica dell'algoritmo e quindi, pur iniziando nello stesso momento, i diversi task *slaves* terminano i blocchi di M_B passi in istanti diversi. Nell'intervallo di tempo compreso fra la fine del blocco considerato da parte del primo e dell'ultimo *slaves* alcuni di essi sono inattivi (in attesa di ricevere il valore aggiornato di E_T). Parte del calcolo di ogni blocco è quindi eseguita in modo seriale e, supposto che questa parte sia in media una frazione costante dell'intero calcolo (la parte seriale di ogni blocco è soggetta a fluttuazioni in quanto lo sono le operazioni che ogni singolo

task esegue) arriviamo a giustificare una frazione seriale approssimativamente indipendente da P . Osserviamo che, a causa delle fluttuazioni nel numero di operazioni, il nostro caso è diverso da quello considerato nella Sez. 3.5 in cui abbiamo supposto costante tale numero e quindi il contrasto sopra accennato è solo apparente. Anche i punti relativi al tempo totale sono stati interpolati con la retta di Eq. (3.5) ottenendo una frazione seriale di circa l'8% dovuta al contributo della fase di equilibratura che è circa il 20% del totale.

Aumentando di molto il numero P di processori ci aspettiamo una diminuzione dell'efficienza che si traduce in un aumento della frazione seriale con P e dunque una deviazione dall'andamento rettilineo dell'Eq. (3.5). Questo è dovuto all'aumento nel numero delle operazioni di sincronizzazione e comunicazione sia da parte del programma che da parte del software di gestione (PVM) che implicano anche un aumento del carico della rete con conseguente rallentamento della velocità di trasmissione.

Un discorso simile a quello finora fatto per il DMC è valido anche per il VMC date le analogie fra i due programmi paralleli. Le frazioni seriali dedotte dai soli tempi per $P = 1$ e $P = 4$ riportati in Tab. 3.4 sono di circa il 19% per la fase di equilibratura e di circa il 4% per i tempi complessivi.

Per quanto riguarda i tempi di calcolo in generale c'è un aspetto importante da considerare: come già discusso il numero di operazioni eseguite da ogni processore è diverso per la natura stocastica dell'algoritmo. In particolare, sia per il VMC che per il DMC, i tempi di calcolo sono influenzati dai numeri casuali utilizzati che sono differenti in ogni processo e da cui dipende il numero di passi accettati. Così, il numero di passi accettati varia da processore a processore causando delle fluttuazioni statistiche nei tempi di calcolo a cui, nel caso del DMC, si aggiungono anche quelle dovute alla variazione del numero di individui della popolazione. Per quanto detto, anche se tutti i processori sono uguali e dedicati, non impiegano lo stesso tempo per eseguire lo stesso calcolo. Sia per il VMC che per il DMC, la barra d'errore sui tempi di calcolo è stata stimata eseguendo in modo seriale più calcoli (brevi) con gli stessi dati iniziali, eccetto i numeri casuali, per i quali sono state utilizzate sequenze differenti. Si è calcolato il valor medio del tempo speso in tutti questi calcoli e la

relativa deviazione standard e il rapporto fra i due è stato assunto come incertezza relativa su tutti i tempi di calcolo. Abbiamo ottenuto, così, un'incertezza relativa di circa l'1% per il VMC e di circa il 2% per il DMC.

3.8.5 Analisi dell'esecuzione del programma parallelo sulla macchina virtuale

In questa sezione vogliamo analizzare in dettaglio l'effettivo funzionamento del programma parallelo in termini di evoluzione temporale dei vari task che lo costituiscono. In particolare tale analisi ci ha permesso di studiare la dipendenza dei tempi di calcolo dal tipo di rete utilizzata (Ethernet o FDDI) come vedremo a fine sezione. Il cluster utilizzato è formato da tre stazioni di lavoro di cui due Digital Alpha AXP 3000 (*ritz* e *savoy*) e un HP Apollo 9000/735 (*georgesv*). Di grande utilità nell'analisi del funzionamento del programma parallelo, per seguire l'attività dei vari task costituenti, è stato il pacchetto grafico XPVM[89]. XPVM è in grado di seguire l'attività della macchina virtuale in tempo reale e visualizzare sotto forma grafica lo stato di esecuzione di tutti i task in esecuzione al passare del tempo. Le Fig. 3.5-3.9 sono state ottenute con XPVM unitamente a PVM per l'esecuzione del programma DMC e utilizzando la rete Ethernet; le Fig. 3.10, 3.11 sono state ottenute con XPVM unitamente a PVM per l'esecuzione del programma DMC e utilizzando la rete FDDI; nel seguito discuteremo il loro significato in termini di funzionamento del programma parallelo. In ognuna delle Fig. 3.5-3.11 le barre orizzontali rappresentano l'evoluzione in funzione del tempo (da sinistra a destra) dei task concorrenti che costituiscono il programma parallelo. Il loro stato è rappresentato dai diversi colori (tonalità di grigio) e il cambiamento di stato è indicato da una linea nera verticale attraverso la barra. Il verde (grigio scuro) indica che il task sta eseguendo la sua parte di programma vero e proprio e cioè non vi è nessuna chiamata alla libreria PVM; non necessariamente l'esecuzione avviene con il processore dedicato, la frazione del tempo disponibile dipende dal carico della macchina e non è visibile mediante XPVM. Il colore giallo (grigio chiaro) indica che il task sta eseguendo routine interne alla libreria PVM, mentre il colore bianco indica una situazione di attesa

di dati dovuta a una ricezione bloccante. Le linee sottili rosse (nere) che connettono due diverse barre indicano lo scambio di messaggi (dati) fra task, e più precisamente iniziano nel momento in cui un messaggio è stato spedito da un task (operazione di *send*) e terminano nel momento in cui è stato ricevuto da un altro task (operazione di *receive*, non viene indicato il momento in cui il messaggio viene immagazzinato nel buffer). Analizziamo ora, passo per passo, le operazioni del programma DMC che causano il cambiamento di stato dei task con riferimento alle figure.

1. Il processo *master* viene mandato in esecuzione su una macchina qualsiasi del cluster, nel caso specifico **ritz**.
2. Il *master*, dopo aver letto dai file i dati iniziali, entra a far parte della macchina virtuale, risulta visibile tramite XPVM (in Fig. 3.5 indicato dalla barra con etichetta **ritz:wmc_master**) e genera i processi *slave* di equilibratura (*slaves*).
3. Dopo l'ingresso nella macchina virtuale, anche gli *slaves* sono visibili in Fig. 3.5 (etichette **ritz:wmc_slaves**, ecc.). Essi, dopo le operazioni iniziali sono pronti a ricevere in modo bloccante dal *master* i dati iniziali. Questo tipo di ricezione fa sì che i task rimangano in attesa (zone bianche a sinistra nelle barre orizzontali in Fig. 3.5) finché non arrivano i dati.
4. Dopo che ogni *slaves* ha ricevuto i dati iniziali, spedisce al *master* la conferma dell'avvenuta ricezione (linee sottili dal *master* agli *slaves* e viceversa in Fig. 3.5) e inizia a eseguire i calcoli del primo blocco di equilibratura.
5. Dalle Fig. 3.5 e 3.6 possiamo vedere che ogni *slaves*, alla fine di un blocco di M_B passi, spedisce al *master* i dati (linee sottili verticali dagli *slaves* al *master*) necessari per aggiornare l'energia di prova E_T . Fatto questo, prima di iniziare i calcoli relativi al blocco successivo, attende, dal *master*, il nuovo valore di E_T .
6. Il *master* riceve in maniera bloccante i risultati dei calcoli di blocco dagli *slaves* (Fig. 3.5 e 3.6). Solo quando ha i dati di tutti gli *slaves* relativi allo stesso blocco, può aggiornare

E_T e spedirlo agli *slaves* (linee sottili verticali dal *master* agli *slaves*). Si può osservare che gli *slaves* ricevono nello stesso momento il valore aggiornato di E_T e iniziano contemporaneamente i vari blocchi di passi. Questo punto e quello precedente si ripetono finché tutti i walker non compiono M passi.

7. Raggiunto l'equilibrio tutti gli *slaves* spediscono al *master* i risultati dei calcoli relativi all'ultimo blocco e la loro configurazione finale di walker e poi terminano l'esecuzione (Fig. 3.7).
8. Dopo che il *master* ha ricevuto, in modo bloccante, da tutti gli *slaves*, procede nel riordinare le configurazioni e nel calcolare i risultati ultimi dell'equilibratura (Fig. 3.7).
9. A questo punto, il *master* genera i processi *slave* di media (*slavem* indicati con l'etichetta `ritz:wmc_slavem`, ecc.) e per questi ultimi si ripetono le operazioni descritte nei punti 3 e 4 per l'equilibratura (Fig. 3.7).
10. Anche ora, come nella fase di equilibratura, alla fine di ogni blocco ogni *slavem* spedisce al *master* i suoi risultati (linee sottili dagli *slavem* al *master* in Fig. 3.8), non per motivi di sincronizzazione, ma per evitare di perderli nel caso di un errore che ne interrompa l'esecuzione. Non si ha sincronizzazione poiché ogni *slavem*, alla fine di un blocco di M_B passi, aggiorna il valore di E_T ⁶ e indipendentemente dagli altri *slavem* procede con il blocco successivo. In questo modo, contrariamente alla parte di equilibratura, non si hanno attese dovute alla sincronizzazione e in genere tutti gli *slavem* eseguono un diverso numero di blocchi.
11. Il *master*, ogni volta che riceve i dati dagli *slavem* (in maniera non bloccante), conta il numero di passi fatti sino a quel momento da tutta la popolazione: quando raggiunge il valore desiderato \bar{M} calcola i risultati finali, termina l'esecuzione e così fanno anche gli *slavem*. Pur eseguendo in generale un diverso numero di blocchi tutti gli *slavem*

⁶Il motivo per cui E_T viene aggiornato dal *master* nella fase di equilibratura e dagli *slavem* stessi nella fase di media, è che mentre ogni *slaves* ha solo una parte della popolazione pari a N/P , ogni *slavem* ha il numero totale N di configurazioni (vedere Sez. 3.7).

terminano approssimativamente assieme: il carico di ogni processore in questo modo risulta ben bilanciato (Fig. 3.9).

Dai grafici si può osservare che uno degli *slave* (sia di equilibratura che di media) viene generato sullo stesso ospite del *master* dato che quest'ultimo ha bisogno di poco tempo di calcolo, e che questo è quello con cui il *master* comunica più velocemente, poichè non viene utilizzata la rete, ma il bus interno della macchina.

In conseguenza della sincronizzazione descritta nei punti 5 e 6, nella fase di equilibratura, come si osserva dalla Fig. 3.6, tutti gli *slaves* procedono nei calcoli alla velocità del più lento. Questo è il motivo per cui, in questa fase, non conviene aumentare troppo il numero di *slaves* ma vale la pena di scegliere gli ospiti più veloci. In ogni caso, l'utilizzo di un numero troppo elevato di task (sia di equilibratura che di media) può portare ad un peggioramento delle prestazioni della macchina virtuale dovuto sia all'aumento del numero delle operazioni di comunicazione, sia al possibile sovraccarico della rete a causa della comunicazione dell'utente e/o del software di gestione (PVM) che provvede alla supervisione dei vari task.

Come già osservato, la differenza di velocità dovuta a reti di tipo diverso si nota specialmente quando si spediscono grossi pacchetti di dati e quindi, nel nostro caso, quando il *master* spedisce i dati iniziali agli *slave* di equilibratura e di media, e quando gli *slave* di equilibratura spediscono al *master* la propria configurazione finale. In questi casi riportiamo i grafici XPVM (Fig. 3.10 e 3.11) relativi al programma DMC eseguito su **ritz**, **savoy** e **georgesv** collegate tramite la rete FDDI, in modo da poter eseguire il confronto con le figure precedenti per la rete Ethernet (Fig. 3.5 e 3.7). In entrambi i casi, possiamo notare che la macchina più lenta è **ritz** in quanto è l'ultima a terminare il primo blocco di equilibratura e quella più veloce è **georgesv** poiché è la prima ad ultimarlo. Quando il *master* spedisce i dati iniziali agli *slave* possono presentarsi due casi: (1) la rete è "lenta", gli *slave* sono in grado di ricevere prima che i dati siano arrivati e perciò vi sono dei tempi morti (spazi bianchi nelle figure) in cui i processori non lavorano per i nostri processi *slave*. Questo è il caso che si ha con il collegamento Ethernet prima che gli *slave* (di equilibratura e di media)

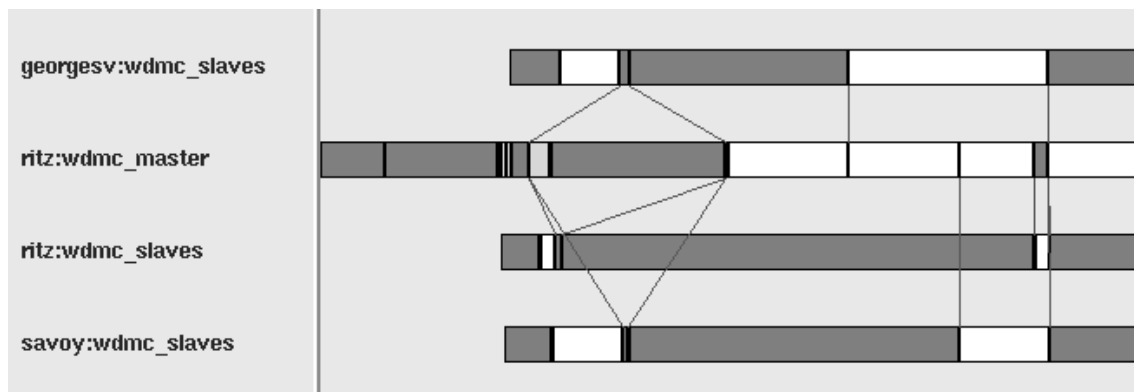


Figura 3.5: Evoluzione temporale dei task nella parte iniziale del programma DMC eseguito su tre ospiti (`ritz`, `georgesv` e `savoy`) collegati mediante la rete Ethernet (vedere testo). Il *master*, letti in maniera seriale i file di dati iniziali, genera i processi *slaves* e spedisce loro i dati necessari per la fase di equilibratura. Gli *slaves* ricevono in maniera bloccante, spediscono al *master* la conferma dell'avvenuta ricezione e procedono con il primo blocco di passi. Si può osservare che lo *slave* che riceve per primo i dati iniziali di equilibratura è quello che si trova sulla stessa macchina in cui sta il *master* e cioè `ritz`.

ricevano i dati iniziali dal *master* (vedere Fig. 3.5 e 3.7). (2) La rete è “veloce” e i dati arrivano prima di poter essere ricevuti. Perciò, appena è possibile, questi vengono letti e non si hanno attese. Questo è il caso che si ha con il collegamento FDDI prima che gli *slave* (di equilibratura e di media) ricevano i dati iniziali dal *master* (vedere Fig. 3.10 e 3.11). Infatti, dalle Fig. 3.10 e 3.11 si può notare che gli spazi bianchi precedenti alla ricezione dei dati iniziali da parte degli *slaves* e degli *slavem* sono molto piccoli e a volte assenti. Inoltre, contrariamente al caso con il collegamento Ethernet, con il collegamento FDDI si può notare che tutti gli *slave* ricevono i dati iniziali quasi contemporaneamente: questa è una conferma del fatto che i dati giungono prima di poter essere ricevuti (vedere punto 3). In realtà, nel nostro caso il guadagno che si ha utilizzando FDDI è trascurabile rispetto a Ethernet dato che lo è il tempo di comunicazione rispetto al totale, soprattutto quando il numero di passi è elevato.

Tenendo presenti le considerazioni fatte alla fine della Sez. 3.7.1, quanto detto per il DMC vale anche nel caso del VMC.

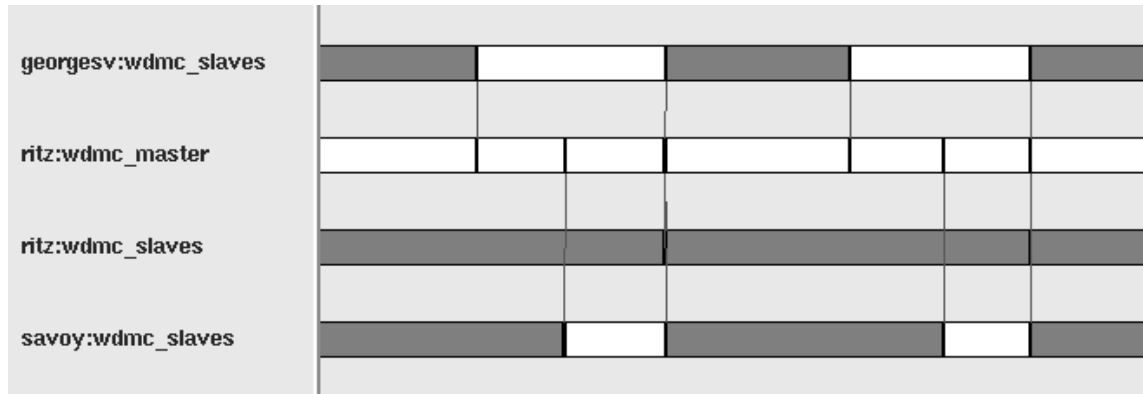


Figura 3.6: È il proseguimento della Fig. 3.5. Si nota che in questa fase, e cioè quella di equilibratura, che il processore di **ritz** non lavora quasi mai per il *master* e che i vari blocchi di passi di calcolo vengono iniziati contemporaneamente da tutti i processi *slaves* perchè ricevono contemporaneamente dal *master* il valore aggiornato di E_T (sincronizzazione).

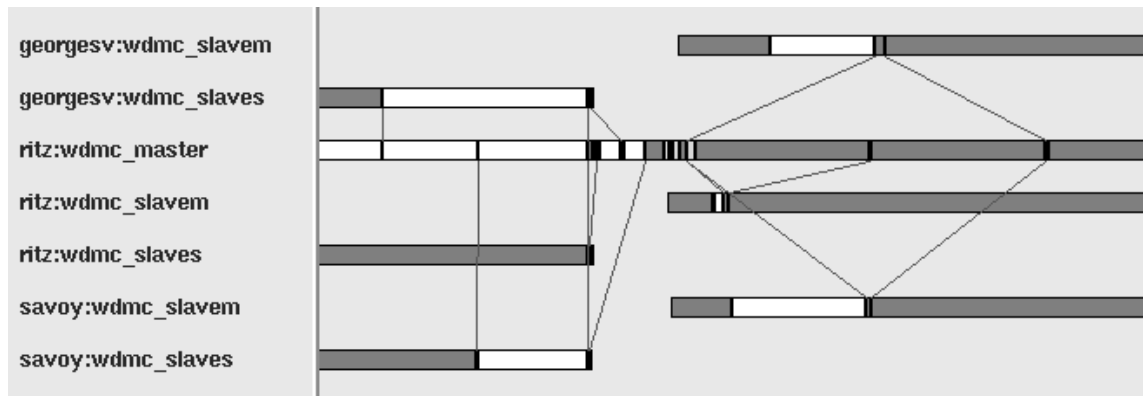


Figura 3.7: Parte finale di equilibratura e inizio fase di media: comunicazione al *master* (che riceve in modo bloccante) dei dati dell'ultimo blocco e della configurazione di walker, generazione degli *slavem*, distribuzione dei dati iniziali agli *slavem* e conferma dell'avvenuta ricezione degli *slavem* al *master*. Anche qui, come in Fig. 3.5, lo *slavem* che riceve per primo i dati iniziali di media è quello che si trova sulla stessa macchina in cui sta il *master*, **ritz**.

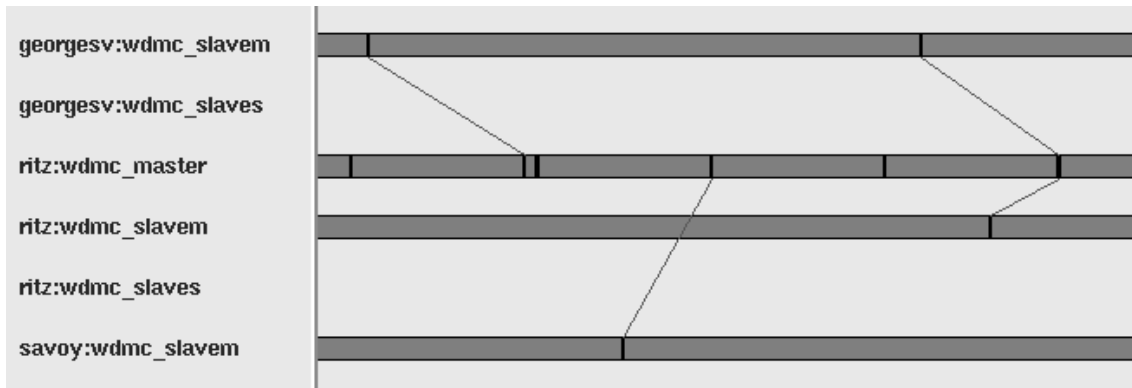


Figura 3.8: Questa figura mostra una parte della fase di media. Si può notare che, al contrario della Fig. 3.6, non vi è sincronizzazione e che il *master* esegue periodicamente una ricezione non bloccante (segmenti neri verticali) per ricevere le energie di blocco spedite dai vari *slavem*. Fra una ricezione non bloccante e l'altra il *master* sospende l'esecuzione (non visibile in figura).

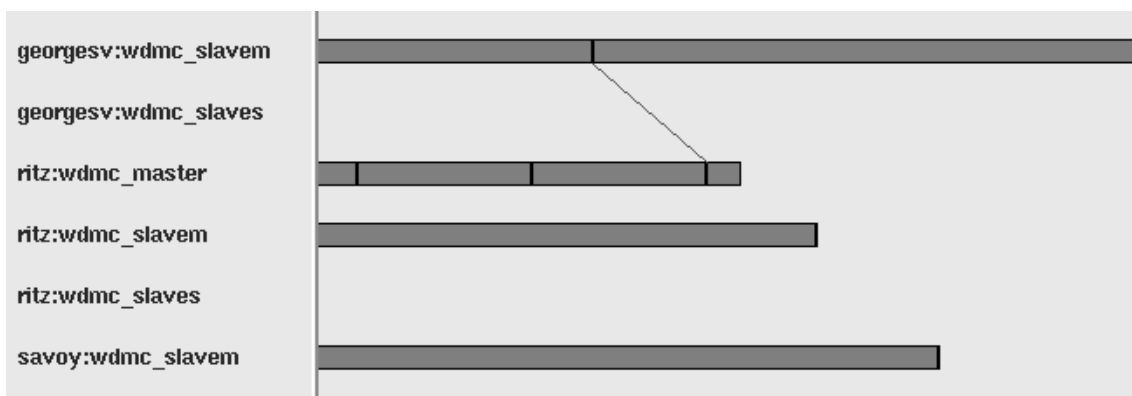


Figura 3.9: Parte finale del DMC. Quando il *master* si accorge che il numero totale di passi eseguiti è quello desiderato, interrompe la ricezione dei dati dagli *slavem*, esegue i calcoli finali e termina l'esecuzione. Una volta che i processi *slavem* si accorgono che il processo *master* è terminato cessano anch'essi l'esecuzione.

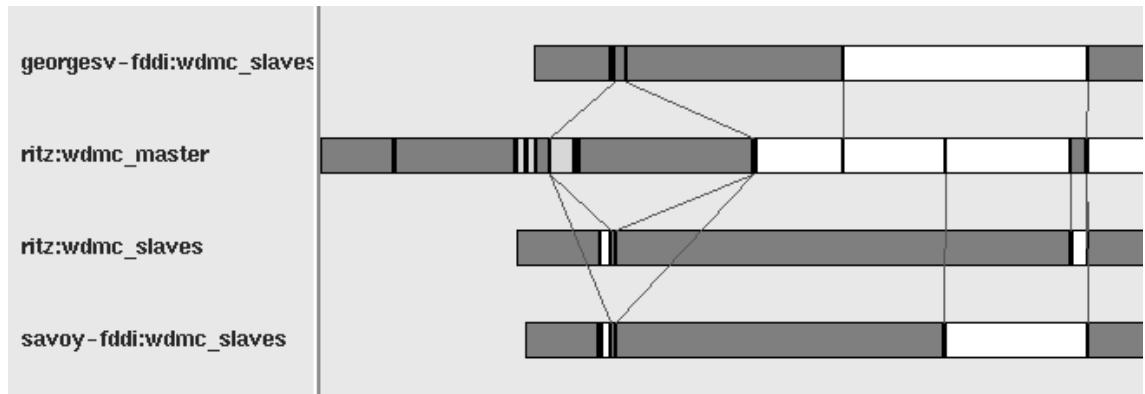


Figura 3.10: Evoluzione temporale dei task nella parte iniziale del programma DMC eseguito su `ritz`, `georgesv` e `savoy` collegati mediante la rete FDDI (vedere testo e la Fig. 3.5).

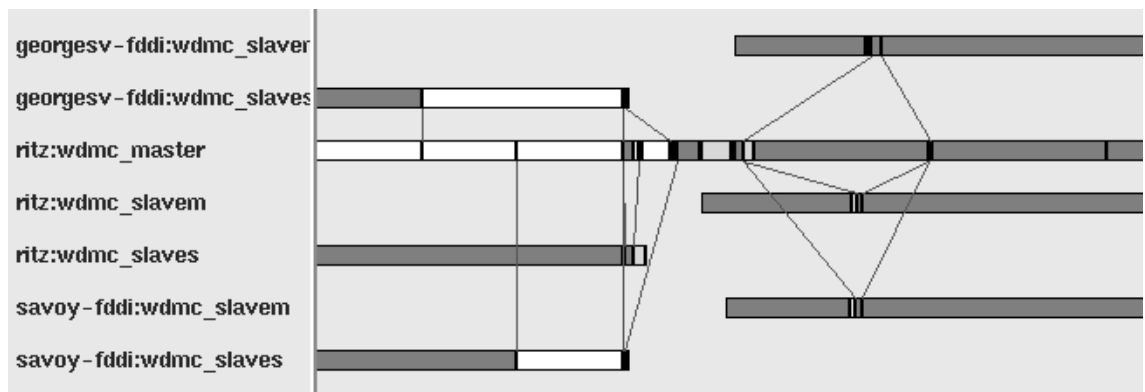


Figura 3.11: Parte finale di equilibratura e inizio fase di media: comunicazione al *master* in modo bloccante dei dati dell'ultimo blocco e della configurazione di walker, generazione degli *slavem*, distribuzione dei dati iniziali agli *slavem* e conferma dell'avvenuta ricezione degli *slavem* al *master* (collegamento con rete FDDI, vedere testo).

3.8.6 Confronto con APE100

Per concludere questo capitolo vogliamo fare un confronto fra le prestazioni dei programmi paralleli VMC e DMC da noi sviluppati per un'architettura distribuita con i corrispondenti programmi messi a punto da M. Rosati[14] per un calcolatore parallelo massivo: APE100. APE100[14] è un'architettura parallela massiva di tipo SIMD a indirizzamento globale della memoria⁷ che può avere da 8 a 2048 processori uguali ognuno dei quali può effettuare fino a 50 milioni di operazioni in virgola mobile al secondo (50 Mflop). Nelle Tab. 3.8 e 3.9 riportiamo i tempi di esecuzione in secondi dei programmi VMC e DMC per l'atomo Sr sul cluster descritto nel Sez. 3.8.1 e su APE100, insieme alla velocità massima di ogni processore in Mflop. Da queste possiamo osservare che in entrambe i casi il programma su APE100 con 8 processori dovrebbe essere più veloce di un fattore 5 (un fattore 2 nel numero di processori e un fattore 2.5 nella velocità) rispetto al nostro cluster di 4 processori, mentre in realtà i tempi di esecuzione sono più o meno equivalenti. È ragionevole pensare che il motivo di tale bassa efficienza nel caso di APE100 sia dovuto al maggior numero di operazioni che si rendono necessarie a causa dell'indirizzamento globale della memoria, una limitazione davvero importante. Inoltre, mentre nel caso del VMC il guadagno di tempo che si ottiene quadruplicando il numero dei processori è simile (un fattore di circa 3.8 sia per il cluster che per APE100), nel caso del DMC il guadagno per APE100 è decisamente inferiore: un fattore di circa 2.9 per APE100 contro 3.8 per il cluster.

In conclusione, possiamo dire che l'efficienza del cluster è decisamente superiore a quella di APE100 che, però, ha il vantaggio di poter utilizzare un numero molto più elevato di processori.

3.9 Conclusioni

Per concludere vogliamo tracciare un quadro riassuntivo dei risultati ottenuti e delineare quelli che potrebbero essere gli sviluppi futuri. Il presente lavoro è consistito nello sviluppo,

⁷Memoria a indirizzamento globale significa che gli indirizzi delle locazioni di memoria locale sono identici per tutti i processori.

architettura	numero di processori	velocità massima	tempo
CLUSTER (PVM 3.3)	4	20	1124
APE100	8	50	1396
APE100	32	50	364

Tavola 3.8: Tempi di esecuzione del programma VMC su un cluster e su APE100 (vedere testo).

architettura	numero di processori	velocità massima	tempo
CLUSTER (PVM 3.3)	4	20	2340
APE100	8	50	2114
APE100	32	50	739

Tavola 3.9: Tempi di esecuzione del programma DMC su un cluster e su APE100 (vedere testo).

verifica ed effettivo utilizzo di due diversi programmi orientati al calcolo parallelo, realizzazioni degli algoritmi del Monte Carlo Variazionale e del Monte Carlo di Diffusione (nella recente versione di Umrigar). In particolare, l'architettura parallela per la quale i programmi sono stati costruiti consiste di un insieme di stazioni di lavoro collegate fra loro tramite una rete (Ethernet o FDDI), che vengono viste dall'utente come un unico calcolatore parallelo (o macchina virtuale) grazie al software di gestione PVM. Questo tipo di architettura parallela distribuita, dai costi contenuti e di larga diffusione, si è rivelata adatta ed efficiente per la soluzione dei problemi da noi considerati. Inoltre, in un'ottica più generale, vale la pena osservare che le problematiche da noi affrontate sono le stesse anche nel caso dei calcolatori paralleli massivi MIMD. Ancora, la scelta di PVM come software di gestione, che si è rivelato "universale" (è disponibile per quasi tutte le diverse stazioni di lavoro) maneggevole e affidabile, ha permesso di utilizzare un linguaggio di programmazione molto diffuso nell'ambiente scientifico quale il FORTRAN, con il risultato che la trasformazione di un programma sequenziale esistente nel corrispondente programma parallelo risulta grandemente semplificata. Non ultima, la somiglianza sopra citata fra l'architettura distribuita utilizzata e i calcolatori MIMD massivi, si traduce nel fatto che i programmi da noi sviluppati me-

dante PVM sono in grado di funzionare senza alcuna modifica (sempre mediante PVM) anche su un certo numero di architetture MIMD massive come mostrato nella Tab. 3.2.

Per quanto riguarda l'obiettivo più specifico di sviluppare dei programmi paralleli in grado di massimizzare il guadagno in termini di tempo di attesa dei risultati (compatibilmente con le risorse a disposizione) in modo tale da rendere le tecniche VMC e DMC accessibili e competitive rispetto ad altre metodologie computazionali nello studio di sistemi complessi, possiamo dire che i risultati sono decisamente positivi. I calcoli da noi effettuati per determinare le proprietà dello stato fondamentale di un sistema di due elettroni interagenti in un potenziale esterno indicano che i tempi di attesa dei risultati hanno una dipendenza dal numero di processori utilizzati che si discosta poco dall'andamento teorico ottimale (almeno per un numero di processori inferiore a 10). Inoltre, il modo in cui è stato affrontato il problema della parallelizzazione, e cioè della suddivisione del lavoro fra i vari processori disponibili, in larga misura indipendente dal particolare sistema fisico da studiare, ci fa ritenere che le prestazioni ottenute rimangano sostanzialmente le stesse anche per sistemi fisici più complessi.

Dal punto di vista dei metodi VMC e DMC, quello che è risultato più facilmente parallelizzabile è stato il primo in quanto pone pochissimi problemi di comunicazione e lavora con una popolazione costante permettendo così un bilanciamento ottimale del carico dei vari task. Nel caso del DMC abbiamo lavorato con il più recente algoritmo di Umrigar, invece che con quello di Reynolds, più adatto alla parallelizzazione grazie alla maggior stabilità della popolazione. Lo studio esteso a vari sistemi a due elettroni ha dimostrato che, almeno in questo caso, l'algoritmo di Umrigar è perfettamente equivalente a quello di Reynolds anche se non porta a sostanziali miglioramenti come suggerito dagli autori. Il motivo di questo va probabilmente ricercato nella semplicità del sistema da noi studiato, dove molti degli accorgimenti ideati per casi più complessi non risultano sostanziali. Sia nel caso del VMC che del DMC la correlazione fra passi successivi è un aspetto importante perché può portare a una sottostima degli errori statistici. Per evitare questo inconveniente abbiamo studiato il problema nel dettaglio mediante il calcolo di due diverse grandezze, il tempo di correlazione

e la funzione di autocorrelazione lineare che ci hanno permesso di tenere in debito conto gli effetti della correlazione fra i passi.

Per quanto riguarda le prospettive future, i risultati più che soddisfacenti ottenuti nel caso dei sistemi a due elettroni e la relativa indipendenza dal particolare sistema fisico data al metodo utilizzato per la parallelizzazione degli algoritmi per il VMC e il DMC, dovrebbero permettere l'estensione del presente lavoro allo studio di sistemi complessi quali atomi pesanti, molecole e solidi finora difficilmente accessibili al calcolo sequenziale a causa dell'enorme onere computazionale.

3.10 Ringraziamenti

Un ringraziamento particolare va a G. Demurtas assieme alla quale questo lavoro è stato realizzato. Per la loro assistenza e disponibilità vorremmo inoltre ringraziare C. Nardone e D. Greco del Gruppo di Calcolo Parallelo del Centro Ricerche e Sviluppi Superiori in Sardegna diretto da P. Rossi, e S. Pulvirenti del Gruppo Calcolatori e Reti. Infine ringraziamo M. Rosati e G. B. Bachelet per averci messo a disposizione i risultati relativi a APE100 e per le utili discussioni intercorse.

Appendici

Appendice A

Integrazione delle equazioni di Kohn-Sham

Descriviamo brevemente come vengono risolte numericamente le equazioni di Kohn-Sham (1.41). La necessità di richiamare un certo numero di cose note è collegata ad alcune scelte chiave nella costruzione della funzione di costo (vedere Sez. 1.3.2). Le equazioni di Kohn-Sham contengono gli autovalori ϵ_ℓ e gli autovettori $\chi_\ell(r)$ (sconosciuti) e la densità

$$n(r) = \frac{1}{4\pi r^2} \sum_{\ell} f_{\ell} |\chi_{\ell}(r)|^2, \quad (\text{A.1})$$

dove f_{ℓ} sono i numeri di occupazione degli orbitali di Kohn-Sham $\chi_{\ell}(r)$. La densità determina il potenziale di schermo $v_{HXC}(r)$ tramite l'Eq. (1.35) e la soluzione auto-consistente si ottiene mediante tecniche iterative standard. Il modo in cui risolviamo le Eq. (1.41) per una data densità di prova, e cioè dato un certo potenziale $v_{HXC}(r)$, è il seguente. Una volta fissato il valore del momento angolare ℓ , per ogni valore dell'energia $\epsilon < 0$ le equazioni di Kohn-Sham ammettono una soluzione $\chi_{\ell}^{IN}(\epsilon, r)$ e una soluzione $\chi_{\ell}^{OUT}(\epsilon, r)$, tali che [28, 41]

$$\chi_{\ell}^{IN}(\epsilon, r) \stackrel{r \rightarrow 0}{\sim} c_1 r^{\ell+1} + O(r^{\ell+3}), \quad (\text{A.2})$$

$$\chi_{\ell}^{OUT}(\epsilon, r) \stackrel{r \rightarrow +\infty}{\sim} c_2 \exp(-\sqrt{-2\epsilon}r), \quad (\text{A.3})$$

e queste possono essere calcolate facilmente per ogni valore negativo dell'energia ϵ su una griglia numerica finita (come descritto, per esempio, da Hamann[33]). Se, dopo aver riscalato

una delle due per continuità con la seconda al raggio R^1

$$\chi_\ell^{IN}(\epsilon, R) = \chi_\ell^{OUT}(\epsilon, R), \quad (\text{A.4})$$

troviamo che anche le derivate prime sono uguali in R

$$\chi_\ell'^{IN}(\epsilon, R) = \chi_\ell'^{OUT}(\epsilon, R), \quad (\text{A.5})$$

allora abbiamo trovato l'autovalore $\epsilon_\ell = \epsilon$ e la corrispondente autofunzione. La differenza $\chi_\ell'^{IN}(\epsilon, R) - \chi_\ell'^{OUT}(\epsilon, R)$ è dunque una misura della distanza di ϵ dall'autovalore ϵ_ℓ^2 e, al primo ordine in energia, l'autovalore può essere stimato da

$$\epsilon_\ell \simeq \epsilon + \frac{1}{2}[\chi_\ell'^{IN}(\epsilon, R) - \chi_\ell'^{OUT}(\epsilon, R)]\chi_\ell^{OUT}(\epsilon, R). \quad (\text{A.6})$$

Come noto, con la formula precedente è possibile ricavare l'autovalore ϵ_ℓ entro una data precisione in poche iterazioni. La ragione per cui ricordiamo qui questa procedura è per sottolineare che nel nostro procedimento non abbiamo bisogno nemmeno di queste iterazioni. Durante l'annealing simulato prendiamo $R = r_c$ e, per ogni nuovo pseudohamiltoniano di prova, abbiamo bisogno di integrare $\chi_\ell^{IN}(\epsilon, r)$ solo una volta all'energia dell'autovalore full-core. Il nostro pseudohamiltoniano finale deve soddisfare le Eq. (1.56) e (1.57): la funzione d'onda χ_ℓ e la sua derivata radiale χ_ℓ' devono semplicemente essere identiche alle corrispondenti quantità full-core oltre R , e così l'Eq. (A.6) ci permette di stimare la distanza fra l'autovalore full-core ϵ_ℓ^{FC} e l'autovalore dello pseudohamiltoniano di prova in considerazione. Questo chiarisce l'equivalenza di $d[x_\ell(\epsilon_\ell^{FC}, r_c), x_\ell^{FC}(\epsilon_\ell^{FC}, r_c)]$ e $d(\epsilon_\ell, \epsilon_\ell^{FC})$, discussa nella Sez. 1.3.2, e la convenienza di usare $d[x_\ell(\epsilon_\ell^{FC}, r_c), x_\ell^{FC}(\epsilon_\ell^{FC}, r_c)]$, che richiede una sola integrazione, invece di $d(\epsilon_\ell, \epsilon_\ell^{FC})$, che richiede più integrazioni iterative.

¹Un valore conveniente per R è il cosiddetto punto di inversione classico, definito da $v(r) \geq \epsilon_\ell \forall r \geq R$. Con questa scelta, R è il raggio dove la funzione d'onda radiale comincia a decadere verso zero dopo aver raggiunto il suo massimo più esterno, dato che la regione esterna a R corrisponde alla regione classicamente proibita. Così se R viene scelto, come facciamo noi, uguale al punto di inversione classico, abbiamo sempre $R > r_c$: il nostro raggio di core r_c è sempre compreso fra il nodo e il massimo più esterni delle funzioni d'onda di valenza rilevanti, mentre R è sempre più interno del massimo più esterno.

²Per distinguere fra stati dello stesso momento angolare, si conta il numero dei nodi (radiali) di $\chi_\ell^{IN}(\epsilon, r)$. Per lo stato di energia più bassa questo numero deve essere 0. Se non lo è, l'energia ϵ è troppo alta e deve essere diminuita.

Appendice B

Pseudohamiltoniani e Monte Carlo della Funzione di Green

Consideriamo brevemente cosa cambia nel metodo GFMC in approssimazione a nodi fissi quando utilizziamo lo pseudohamiltoniano. Seguendo Reynolds *et al.*[4] scriviamo l'equazione di Schrödinger nel “tempo immaginario” per un sistema atomico

$$\begin{aligned} -\frac{\partial \Phi(\mathbf{R}, t)}{\partial t} &= \hat{H}^{PS} \Phi(\mathbf{R}, t), \\ -\frac{\partial \Phi(\mathbf{R}, t)}{\partial t} &= \left[-\frac{1}{2} \nabla^2 + \hat{V}^{PS}(\mathbf{R}) + U(\mathbf{R}) \right] \Phi(\mathbf{R}, t), \end{aligned} \quad (\text{B.1})$$

dove abbiamo introdotto

$$\hat{V}^{PS}(\mathbf{R}) = \sum_i \left[-\frac{1}{2} \nabla_i A(r_i) \nabla_i + \frac{B(r_i)}{2r_i^2} \hat{L}_i^2 \right], \quad (\text{B.2})$$

$$U(\mathbf{R}) = \sum_i v_{ion}(r_i) + \frac{1}{2} \sum_{\substack{i,j \\ i \neq j}} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|}. \quad (\text{B.3})$$

Consideriamo poi una funzione di prova $\Psi_T(\mathbf{R})$ e ricaviamo l'equazione per la funzione $f(\mathbf{R}, t) = \Psi_T(\mathbf{R}) \Phi(\mathbf{R}, t)$. Moltiplichiamo l'Eq. (B.1) per $\Psi_T(\mathbf{R})$ usando l'espressione per $\hat{V}^{PS}(\mathbf{R})$ in cui tutte le derivate sono sulla sinistra

$$\hat{V}^{PS}(\mathbf{R}) = -\frac{1}{2} \sum_i \sum_{\alpha, \beta} \frac{\partial}{\partial r_{i, \alpha}} \frac{\partial}{\partial r_{i, \beta}} D_{\alpha \beta}(\mathbf{r}_i) - \sum_i \sum_{\alpha} \frac{\partial}{\partial r_{i, \alpha}} K_{\alpha}(\mathbf{r}_i), \quad (\text{B.4})$$

dove gli indici α, β si riferiscono alle tre componenti cartesiane x, y, z , e la matrice 3×3 $D_{\alpha \beta}(\mathbf{r}_i)$ e il vettore $K_{\alpha}(\mathbf{r}_i)$ sono definiti come

$$D_{\alpha \beta}(\mathbf{r}_i) = [A(r_i) + B(r_i)] \delta_{\alpha \beta} - r_{i, \alpha} r_{i, \beta} \frac{B(r_i)}{r_i^2}, \quad (\text{B.5})$$

$$K_{\alpha}(\mathbf{r}_i) = \left[-\frac{1}{2} \frac{\partial A(r_i)}{\partial r_i} + \frac{B(r_i)}{r_i} \right] \frac{r_{i, \alpha}}{r_i}. \quad (\text{B.6})$$

Dopo alcuni passaggi otteniamo

$$\frac{\partial f(\mathbf{R}, t)}{\partial t} = \frac{1}{2} \sum_{s,t=1}^{3N} \nabla_s \nabla_t [d_{st}(\mathbf{R}) f(\mathbf{R}, t)] - \nabla \cdot [\mathbf{v}(\mathbf{R}) f(\mathbf{R}, t)] - E_L(\mathbf{R}) f(\mathbf{R}, t). \quad (\text{B.7})$$

$\underline{d}(\mathbf{R})$ è la matrice $3N \times 3N$ definita dalla somma diretta

$$\underline{d}(\mathbf{R}) = \sum_i^{\oplus} (\underline{D}(\mathbf{r}_i) + \underline{I}), \quad (\text{B.8})$$

dove \underline{I} è la matrice identità 3×3 , $\mathbf{v}(\mathbf{R})$ è il vettore dato da

$$\mathbf{v}(\mathbf{R}) = \frac{1}{2} [\underline{d}(\mathbf{R}) \mathbf{F}(\mathbf{R}) - 2\mathbf{K}(\mathbf{R})], \quad (\text{B.9})$$

$$\mathbf{F}(\mathbf{R}) = \nabla \ln \Psi_T(\mathbf{R})^2 = 2 \frac{\nabla \Psi_T(\mathbf{R})}{\Psi_T(\mathbf{R})}, \quad (\text{B.10})$$

$$\mathbf{K}(\mathbf{R}) = \sum_i^{\oplus} \mathbf{K}(\mathbf{r}_i), \quad (\text{B.11})$$

e $E_L(\mathbf{R})$ è l'energia locale

$$E_L(\mathbf{R}) = \frac{\hat{H}^{PS} \Psi_T(\mathbf{R})}{\Psi_T(\mathbf{R})}. \quad (\text{B.12})$$

Le differenze principali con l'equazione di diffusione standard per $f(\mathbf{R}, t)$ sono la matrice di diffusione non costante $\underline{d}(\mathbf{R})$ che modifica il termine di diffusione (derivate del secondo ordine), il termine di deriva (derivate del primo ordine), e la velocità di deriva $\mathbf{v}(\mathbf{R})$ che contiene il nuovo termine $-\mathbf{K}(\mathbf{R})$ oltre alla “forza quantica” modificata $\underline{d}(\mathbf{R}) \mathbf{F}(\mathbf{R})/2$. L'Eq. (B.7) è il punto di partenza per le tecniche GFMC. Nel Monte Carlo di Diffusione siamo interessati all'espressione a tempi corti per la funzione di Green associata all'Eq. (B.7). Per mezzo della formula di Baker-Hausdorff[74] l'approssimazione a tempi corti per la funzione di Green può essere scritta[67]

$$\begin{aligned} G(\mathbf{R}, \mathbf{R}', \tau) &= \frac{1}{(2\pi\tau)^{3N/2} \det[\underline{d}(\mathbf{R}' + \mathbf{v}(\mathbf{R}')\tau)]^{1/2}} \times \\ &\times \exp \left\{ - \frac{[\mathbf{R} - \mathbf{R}' - \mathbf{v}(\mathbf{R}')\tau] \underline{d}(\mathbf{R}' + \mathbf{v}(\mathbf{R}')\tau)^{-1} [\mathbf{R} - \mathbf{R}' - \mathbf{v}(\mathbf{R}')\tau]}{2\tau} \right\} \times \\ &\times \exp \left\{ - \frac{E_L(\mathbf{R}') + E_L(\mathbf{R})}{2} \tau \right\}, \end{aligned} \quad (\text{B.13})$$

$$\det[\underline{d}(\mathbf{R})] = \prod_i [A(r_i) + B(r_i) + 1]^2 [A(r_i) + 1]. \quad (\text{B.14})$$

Osserviamo che i termini di nascita-morte e deriva compaiono esattamente come nel metodo Monte Carlo di Diffusione standard[4], mentre il termine di diffusione non è più uniforme e costante ed è governato dalla matrice $\underline{d}(\mathbf{R})$: questa è solo una complicazione secondaria dato che $\underline{d}(\mathbf{R})$ è una matrice a blocchi e il lavoro aggiuntivo si riduce alla decomposizione di Cholesky[43] di una matrice 3×3 ad ogni passo[67].

Appendice C

Funzione di Green per gli pseudopotenziali non locali

Vogliamo calcolare la funzione di Green a un elettrone per l'operatore non locale

$$\hat{N} = \sum_{\ell=0}^{+\infty} \sum_{m=-\ell}^{\ell} v_{\ell}^{PS}(r) \hat{P}_{\ell m}, \quad (C.1)$$

dove $v_{\ell}^{PS}(r)$ è una funzione radiale (è necessario avere almeno due diverse funzioni per avere un operatore non locale) e $\hat{P}_{\ell m}$ è l'operatore di proiezione nel sottospazio a una particella di momento angolare ℓ con proiezione m lungo l'asse z

$$(\hat{P}_{\ell m} \psi)(\mathbf{r}) = Y_{\ell m}(\theta, \phi) \int Y_{\ell m}^*(\theta', \phi') \frac{\delta(r - r')}{rr'} \psi(\mathbf{r}') d^3 \mathbf{r}'. \quad (C.2)$$

L'operatore \hat{N} compare nella definizione degli pseudopotenziali non locali ed è hermitiano dato che può essere scritto come[28]

$$(\hat{N} \psi)(\mathbf{r}) = \int N(\mathbf{r}, \mathbf{r}') \psi(\mathbf{r}') d^3 \mathbf{r}', \quad (C.3)$$

$$\begin{aligned} N(\mathbf{r}, \mathbf{r}') &= \sum_{\ell=0}^{+\infty} v_{\ell}^{PS}(r) \sum_{m=-\ell}^{\ell} Y_{\ell m}(\theta, \phi) Y_{\ell m}^*(\theta', \phi') \frac{\delta(r - r')}{rr'} = \\ &= \sum_{\ell=0}^{+\infty} \frac{2\ell + 1}{4\pi} v_{\ell}^{PS}(r) P_{\ell}(\mathbf{n}_{\mathbf{r}} \mathbf{n}_{\mathbf{r}'}) \frac{\delta(r - r')}{rr'}, \end{aligned} \quad (C.4)$$

con $N(\mathbf{r}, \mathbf{r}') = N^*(\mathbf{r}', \mathbf{r})$; P_{ℓ} è il polinomio di Legendre di ordine ℓ e $\mathbf{n}_{\mathbf{r}}$, $\mathbf{n}_{\mathbf{r}'}$ sono i versori nelle direzioni di \mathbf{r} e \mathbf{r}' .

La funzione di Green è data da

$$G(\mathbf{r}, \mathbf{r}', \tau) = \langle \mathbf{r} | \exp(-\tau \hat{N}) | \mathbf{r}' \rangle, \quad (C.5)$$

e introducendo la base di onde piane $|\mathbf{k}\rangle$ tale che $\langle \mathbf{r}|\mathbf{k}\rangle = (2\pi)^{-3/2} \exp(-i\mathbf{k}\mathbf{r})$ possiamo scrivere

$$\begin{aligned} \langle \mathbf{r}|\exp(-\tau\hat{N})|\mathbf{r}'\rangle &= \int d^3\mathbf{k} \langle \mathbf{r}|\mathbf{k}\rangle \langle \mathbf{k}|\exp(-\tau\hat{N})|\mathbf{r}'\rangle = \\ &= \int d^3\mathbf{k} \langle \mathbf{r}|\mathbf{k}\rangle \overline{\langle \mathbf{r}'|\exp(-\tau\hat{N})|\mathbf{k}\rangle} = \\ &= \frac{1}{(2\pi)^3} \int d^3\mathbf{k} \exp(-i\mathbf{k}\mathbf{r}) \exp(-\tau\hat{N}) \exp(i\mathbf{k}\mathbf{r}'). \end{aligned} \quad (\text{C.6})$$

Consideriamo l'espansione[28]

$$\exp(-\tau\hat{N}) = \sum_{n=0}^{+\infty} \frac{(-\tau\hat{N})^n}{n!}, \quad (\text{C.7})$$

$$\exp(i\mathbf{k}\mathbf{r}) = \sum_{\ell=0}^{+\infty} i^\ell (2\ell+1) j_\ell(kr) P_\ell(\mathbf{n}_\mathbf{k}\mathbf{n}_\mathbf{r}), \quad (\text{C.8})$$

in cui j_ℓ è la funzione di Bessel sferica di ordine ℓ . Sostituendo nell'Eq. (C.6) otteniamo

$$G(\mathbf{r}, \mathbf{r}', \tau) = \frac{1}{(2\pi)^3} \int d^3\mathbf{k} \exp(-i\mathbf{k}\mathbf{r}) \sum_{n=0}^{+\infty} \sum_{\ell=0}^{+\infty} i^\ell (2\ell+1) j_\ell(kr') \frac{(-\tau\hat{N})^n}{n!} P_\ell(\mathbf{n}_\mathbf{k}\mathbf{n}_{\mathbf{r}'}). \quad (\text{C.9})$$

Quindi consideriamo l'azione di \hat{N} su P_ℓ

$$\hat{N}P_\ell(\mathbf{n}_\mathbf{k}\mathbf{n}_{\mathbf{r}'}) = \int d^3\mathbf{r}'' \frac{\delta(r' - r'')}{r' r''} \sum_{\ell'=0}^{+\infty} \frac{2\ell' + 1}{4\pi} v_{\ell'}^{PS}(r') P_{\ell'}(\mathbf{n}_{\mathbf{r}'}\mathbf{n}_{\mathbf{r}''}) P_\ell(\mathbf{n}_\mathbf{k}\mathbf{n}_{\mathbf{r}''}), \quad (\text{C.10})$$

e usiamo l'identità[28]

$$\int P_\ell(\mathbf{n}_1\mathbf{n}_2) P_{\ell'}(\mathbf{n}_1\mathbf{n}_3) d\Omega_{\mathbf{n}_1} = \delta_{\ell\ell'} \frac{4\pi}{2\ell+1} P_\ell(\mathbf{n}_2\mathbf{n}_3), \quad (\text{C.11})$$

dove $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$ sono versori e l'integrazione è sulle direzioni di \mathbf{n}_1 ($d\Omega_{\mathbf{n}_1} = \sin\theta_1 d\theta_1 d\phi_1$),

per ottenere

$$\begin{aligned} \hat{N}P_\ell(\mathbf{n}_\mathbf{k}\mathbf{n}_{\mathbf{r}'}) &= \int dr'' \delta(r' - r'') \sum_{\ell'=0}^{+\infty} \frac{2\ell' + 1}{4\pi} v_{\ell'}^{PS}(r') \times \\ &\times \int d\Omega_{\mathbf{r}''} P_{\ell'}(\mathbf{n}_{\mathbf{r}'}\mathbf{n}_{\mathbf{r}''}) P_\ell(\mathbf{n}_\mathbf{k}\mathbf{n}_{\mathbf{r}''}) = \\ &= v_\ell^{PS}(r') P_\ell(\mathbf{n}_\mathbf{k}\mathbf{n}_{\mathbf{r}'}). \end{aligned} \quad (\text{C.12})$$

Da qui segue

$$\hat{N}^n P_\ell(\mathbf{n}_\mathbf{k}\mathbf{n}_{\mathbf{r}'}) = [v_\ell^{PS}(r')]^n P_\ell(\mathbf{n}_\mathbf{k}\mathbf{n}_{\mathbf{r}'}), \quad (\text{C.13})$$

$$\begin{aligned}
G(\mathbf{r}, \mathbf{r}', \tau) &= \frac{1}{(2\pi)^3} \int d^3\mathbf{k} \exp(-i\mathbf{k}\mathbf{r}) \sum_{\ell=0}^{+\infty} i^\ell (2\ell+1) \exp[-\tau v_\ell^{PS}(r')] j_\ell(kr') P_\ell(\mathbf{n}_\mathbf{k} \mathbf{n}_{\mathbf{r}'}) = \\
&= \frac{1}{(2\pi)^3} \sum_{\ell, \ell'=0}^{+\infty} i^{\ell-\ell'} (2\ell+1)(2\ell'+1) \exp[-\tau v_\ell^{PS}(r')] \times \\
&\times \int k^2 j_{\ell'}(kr) j_\ell(kr') dk \int d\Omega_\mathbf{k} P_{\ell'}(\mathbf{n}_\mathbf{k} \mathbf{n}_\mathbf{r}) P_\ell(\mathbf{n}_\mathbf{k} \mathbf{n}_{\mathbf{r}'}) = \\
&= \frac{1}{(2\pi)^3} \sum_{\ell, \ell'=0}^{+\infty} i^{\ell-\ell'} (2\ell+1)(2\ell'+1) \exp[-\tau v_\ell^{PS}(r')] \times \\
&\times \int k^2 j_{\ell'}(kr) j_\ell(kr') dk \delta_{\ell\ell'} \frac{4\pi}{2\ell+1} P_\ell(\mathbf{n}_\mathbf{r} \mathbf{n}_{\mathbf{r}'}). \tag{C.14}
\end{aligned}$$

In conclusione, usando l'identità[28]

$$\int k^2 j_\ell(kr) j_\ell(kr') dk = \frac{\pi}{2} \frac{\delta(r-r')}{rr'}, \tag{C.15}$$

si ha

$$G(\mathbf{r}, \mathbf{r}', \tau) = \sum_{\ell=0}^{+\infty} \frac{2\ell+1}{4\pi} \exp[-\tau v_\ell^{PS}(r')] P_\ell(\mathbf{n}_\mathbf{r} \mathbf{n}_{\mathbf{r}'}) \frac{\delta(r-r')}{rr'}. \tag{C.16}$$

Questa espressione è da confrontare con la funzione di Green per un potenziale locale $v_{loc}(\mathbf{r})$,

$$G(\mathbf{r}, \mathbf{r}', \tau) = \exp[-\tau v_{loc}(\mathbf{r}')] \delta(\mathbf{r} - \mathbf{r}'), \tag{C.17}$$

da cui vediamo che la non località angolare di \hat{N} genera una funzione di Green che collega tutti i punti sulla superficie di una sfera di raggio r . L'altro fatto importante è la presenza di P_ℓ che rende il segno di G positivo solo se vengono imposte particolari disuguaglianze fra le funzioni v_ℓ^{PS} . Per esempio, consideriamo in dettaglio il semplice caso in cui abbiamo solo non località di tipo s , e cioè

$$v_\ell^{PS}(r) = \begin{cases} v_0(r) & \ell = 0, \\ v_1(r) & \ell \geq 1. \end{cases} \tag{C.18}$$

Dopo alcuni passaggi, la funzione di Green, Eq. (C.16), può essere scritta

$$\begin{aligned}
G(\mathbf{r}, \mathbf{r}', \tau) &= \exp[-\tau v_1(r')] \delta(\mathbf{r} - \mathbf{r}') + \\
&+ \frac{1}{4\pi} \frac{\delta(r-r')}{rr'} \{ \exp[-\tau v_0(r')] - \exp[-\tau v_1(r')] \}, \tag{C.19}
\end{aligned}$$

e cioè è la sovrapposizione della funzione di Green per il potenziale locale $v_1(r)$, definita positiva, più un termine che è positivo solo se risulta:

$$v_0(r) \leq v_1(r) \quad \forall r. \quad (\text{C.20})$$

Se dovessimo usare gli pseudopotenziali non locali standard[40] incontreremmo subito dei problemi perché, per la maggior parte della Tavola Periodica, $v_0^{PS}(r) \geq v_1^{PS}(r)$ almeno nella regione di core vicino all'origine. Questo è dovuto all'origine fisica della non località: nella regione di core la componente s dello pseudopotenziale è di solito molto più repulsiva di quella p per tener conto del fatto che nell'atomo vero full-core gli orbitali di valenza ns sono radialmente ortogonali agli $n - 1$ orbitali s più interni, mentre gli orbitali di valenza np sono radialmente ortogonali solo agli $n - 2$ orbitali p più interni. Questo effetto è più marcato per gli atomi della prima riga ($n = 2$) ma esiste per tutti gli atomi.

Appendice D

Elementi di matrice delle onde piane per lo pseudohamiltoniano

Il cuore del metodo delle onde piane all'interno della DFT-LDA è l'equazione secolare per gli stati a un elettrone

$$\sum_{\mathbf{G}'} H^{PS}(\mathbf{k}, \mathbf{G}, \mathbf{G}') c_{\mathbf{k}}(\mathbf{G}') = E(\mathbf{k}) c_{\mathbf{k}}(\mathbf{G}) \quad (\text{D.1})$$

con \mathbf{G}, \mathbf{G}' vettori del reticolo reciproco, $E(\mathbf{k})$ autovalore per ogni vettore \mathbf{k} nella zona di Brillouin, $c_{\mathbf{k}}(\mathbf{G})$ coefficienti degli autostati di Bloch $\psi_{\mathbf{k}}(\mathbf{r})$ rappresentati in termini di un'espansione in onde piane

$$\psi_{\mathbf{k}}(\mathbf{r}) = \sum_{\mathbf{G}} c_{\mathbf{k}}(\mathbf{G}) e^{i(\mathbf{k}+\mathbf{G})\mathbf{r}}, \quad (\text{D.2})$$

e $H^{PS}(\mathbf{k}, \mathbf{G}, \mathbf{G}')$ elementi di matrice delle onde piane per l'hamiltoniano del cristallo \hat{H}^{PS} , Eq. (1.14), per un elettrone

$$H^{PS}(\mathbf{k}, \mathbf{G}, \mathbf{G}') = \langle \mathbf{k} + \mathbf{G} | \hat{H}^{PS} | \mathbf{k} + \mathbf{G}' \rangle. \quad (\text{D.3})$$

Seguendo Ihm, Zunger, e Cohen[68] possiamo scrivere

$$\begin{aligned} H^{PS}(\mathbf{k}, \mathbf{G}, \mathbf{G}') &= \frac{1}{2}(\mathbf{k} + \mathbf{G}')^2 \delta_{\mathbf{G}\mathbf{G}'} + v_{HXC}(\mathbf{G} - \mathbf{G}') + \\ &+ \frac{1}{\Omega_c} \sum_I S_I(\mathbf{G} - \mathbf{G}') v_I^{PS}(\mathbf{k}, \mathbf{G}, \mathbf{G}'), \end{aligned} \quad (\text{D.4})$$

dove l'indice I si riferisce ai differenti ioni nella cella unitaria, S_I è il fattore di struttura associato allo ione I -esimo, Ω_c è il volume della cella, $v_{HXC}(\mathbf{G} - \mathbf{G}')$ sono le componenti di

Fourier di $v_{HXC}(\mathbf{r})$ e $v_I^{PS}(\mathbf{k}, \mathbf{G}, \mathbf{G}')$ sono dati da

$$v_I^{PS}(\mathbf{k}, \mathbf{G}, \mathbf{G}') = \langle \mathbf{k} + \mathbf{G} | \hat{v}_I^{PS} | \mathbf{k} + \mathbf{G}' \rangle. \quad (\text{D.5})$$

Omettendo l'indice I possiamo scrivere esplicitamente

$$\begin{aligned} v^{PS}(\mathbf{k}, \mathbf{G}, \mathbf{G}') &= \langle \mathbf{k} + \mathbf{G} | -\frac{1}{2} \nabla A(r) \nabla | \mathbf{k} + \mathbf{G}' \rangle + \langle \mathbf{k} + \mathbf{G} | \frac{B(r)}{2r^2} \hat{L}^2 | \mathbf{k} + \mathbf{G}' \rangle + \\ &+ \langle \mathbf{k} + \mathbf{G} | v_{ion}(r) | \mathbf{k} + \mathbf{G}' \rangle, \end{aligned} \quad (\text{D.6})$$

e questi elementi di matrice sono stati calcolati da Foulkes e Schlüter[21]. Essi trovano

$$\langle \mathbf{k} + \mathbf{G} | -\frac{1}{2} \nabla A(r) \nabla | \mathbf{k} + \mathbf{G}' \rangle = \frac{1}{2} (\mathbf{k} + \mathbf{G})^T A(|\mathbf{G} - \mathbf{G}'|) \mathbf{I} (\mathbf{k} + \mathbf{G}'), \quad (\text{D.7})$$

$$A(|\mathbf{G} - \mathbf{G}'|) = \int_0^{+\infty} A(r) j_0(|\mathbf{G} - \mathbf{G}'|r) 4\pi r^2 dr, \quad (\text{D.8})$$

con \mathbf{I} matrice identità 3×3 e j_0 funzione di Bessel di ordine 0;

$$\langle \mathbf{k} + \mathbf{G} | \frac{B(r)}{2r^2} \hat{L}^2 | \mathbf{k} + \mathbf{G}' \rangle = \frac{1}{2} (\mathbf{k} + \mathbf{G})^T \mathbf{B}(|\mathbf{G} - \mathbf{G}'|) (\mathbf{k} + \mathbf{G}'), \quad (\text{D.9})$$

$$\mathbf{B}(|\mathbf{G} - \mathbf{G}'|) = B_{\perp}(|\mathbf{G} - \mathbf{G}'|) \mathbf{I} + \mathbf{g} [B_{\parallel}(|\mathbf{G} - \mathbf{G}'|) - B_{\perp}(|\mathbf{G} - \mathbf{G}'|)] \mathbf{g}^T, \quad (\text{D.10})$$

$$B_{\perp}(|\mathbf{G} - \mathbf{G}'|) = \int_0^{+\infty} B(r) \left[\frac{2}{3} j_0(|\mathbf{G} - \mathbf{G}'|r) - \frac{1}{3} j_2(|\mathbf{G} - \mathbf{G}'|r) \right] 4\pi r^2 dr, \quad (\text{D.11})$$

$$B_{\parallel}(|\mathbf{G} - \mathbf{G}'|) = \int_0^{+\infty} B(r) \left[\frac{2}{3} j_0(|\mathbf{G} - \mathbf{G}'|r) + \frac{2}{3} j_2(|\mathbf{G} - \mathbf{G}'|r) \right] 4\pi r^2 dr, \quad (\text{D.12})$$

con \mathbf{g} versore nella direzione di $\mathbf{G} - \mathbf{G}'$ e j_2 funzione di Bessel sferica di ordine 2;

$$\langle \mathbf{k} + \mathbf{G} | v_{ion}(r) | \mathbf{k} + \mathbf{G}' \rangle = \int_0^{+\infty} v_{ion}(r) j_0(|\mathbf{G} - \mathbf{G}'|r) 4\pi r^2 dr. \quad (\text{D.13})$$

Bibliografia

-
- [1] O. Gunnarsson and R. O. Jones, Phys. Rev. B **31**, 7588 (1994).
 - [2] D. M. Ceperley, J. Stat. Phys. **43**, 815 (1986).
 - [3] B. L. Hammond, P. J. Reynolds and W. A. Lester Jr., Phys. Rev. Lett. **61**, 2312 (1988)
 - [4] P. J. Reynolds, D. M. Ceperley, B. J. Alder and W. A. Lester, Jr., J. Chem. Phys. **77**, 5593 (1982).
 - [5] S. Fahy, X. W. Wang and S. G. Louie, Phys. Rev. B **42**, 3503 (1990).
 - [6] G. B. Bachelet, D. M. Ceperley and M. G. B. Chiochetti, Phys. Rev. Lett. **62**, 2088 (1989); M. Chiochetti, Tesi di laurea, Università di Trento, 1988 (non pubblicato).
 - [7] L. Mitáš, E. L. Shirley and D. M. Ceperley, J. Chem. Phys. **95**, 3467 (1991); L. Mitáš, Phys. Rev. A **49**, 4411 (1994); L. Mitáš and R. M. Martin, Phys. Rev. Lett. **72**, 2438 (1994). Gli ultimi due articoli trattano rispettivamente ferro e azoto.
 - [8] A. Bosin, V. Fiorentini, A. Lastrì and G. B. Bachelet, in *Materials Theory and Modelling*, edited by P. D. Bristowe, J. Broughton and J. M. Newsam, MRS Symposia Proceedings No. 291 (Materials Research Society, Pittsburgh, PA, 1993), p. 21.
 - [9] A. Bosin, V. Fiorentini, A. Lastrì and G. B. Bachelet, sottomesso per la pubblicazione a Phys. Rev. A (preprint, 1994).
 - [10] P. H. Acioli and D. M. Ceperley (non pubblicato).
 - [11] M. Menchi, A. Bosin, F. Meloni and G. B. Bachelet in *Materials Theory and Modelling*, edited by P. D. Bristowe, J. Broughton and J. M. Newsam, MRS Symposia Proceedings No. 291 (Materials Research Society, Pittsburgh, PA, 1993), p. 291.
 - [12] M. Menchi, A. Bosin, F. Meloni and G. B. Bachelet, in corso di sottomissione a Phys. Rev. B (preprint, 1994).
 - [13] G. Demurtas, Tesi di laurea, (Cagliari, 1994).

- [14] M. Rosati, Tesi di laurea (Roma, 1994).
- [15] A. Bosin, G. Demurtas and M. Rosati, *Variational and Diffusion Monte Carlo on parallel architectures* presentato alla “Euroconference on Computer Simulation in Condensed Matter Physics and Chemistry”, Marciana Marina, Isola d’Elba, 16-23 settembre 1994.
- [16] *Monte Carlo Method in Statistical Physics*, K. Binder editor (Springer-Verlag, Berlin, 1979); *Applications of the Monte Carlo Method in Statistical Physics*, second edition, K. Binder editor (Springer-Verlag, Berlin, 1987).
- [17] M. L. Cohen and V. Heine, in *Solid State Physics*, vol. 24, edited by H. Ehrenreich, F. Seitz and D. Turnbull (Academic, New York, 1970), pp. 37-248.
- [18] W. E. Pickett, *Comp. Phys. Rep.* **9**, 115 (1989).
- [19] J. F. Annett, *Phys. Rev. Lett.* **69**, 2244 (1992).
- [20] D. R. Hamann, M. Schlüter and C. Chiang, *Phys. Rev. Lett.* **43**, 1494 (1979).
- [21] W. M. C. Foulkes and M. Schlüter, *Phys. Rev. B* **42**, 11505 (1990).
- [22] X. P. Li, D. M. Ceperley and R. M. Martin, *Phys. Rev. B* **44**, 10929 (1991).
- [23] R. M. Dreizler and E. K. U. Gross, *Density Functional Theory* (Springer-Verlag, Berlin, 1990); P. Hohenberg and W. Kohn, *Phys. Rev.* **136**, B864 (1964); W. Kohn and L. J. Sham, *Phys. Rev.* **140**, A1133 (1965).
- [24] P. Focher, A. Lastri, M. Covi and G. B. Bachelet, *Phys. Rev. B* **44**, 8486 (1991).
- [25] M. Teter, *Phys. Rev. B* **48**, 5031 (1993).
- [26] L. Serra, G. B. Bachelet, N. Van Giai and E. Lipparini, *Phys. Rev. B* **48**, 14708 (1993).
- [27] G. B. Bachelet, D. M. Ceperley, M. G. B. Chiochetti and L. Mitáš, in *Progress in Electron Properties of Solids*, edited by E. Doni, R. Girlanda, G. Pastori Parravicini, A. Quattropani (Kluwer, Dordrecht, 1989), p. 11.

-
- [28] Vedere, per esempio, L. D. Landau and E. M. Lifschitz, *Quantum Mechanics: non-relativistic theory* (Pergamon, London-Paris, 1958).
- [29] M. Reed and B. Simon, *Methods of Modern Mathematical Physics - IV: Analysis of Operators* (Academic, New York, 1978), Vol. 4, p. 212-213.
- [30] N. F. Mott and H. S. W. Massey, *The Theory of Atomic Collisions* (Oxford, New York, 1965), p. 181.
- [31] V. I. Smirnov, *A Course of Higher Mathematics - V: Integration and Functiona Analysis*, (Pergamon, Oxford, 1964), Vol. 5, pp. 383-384.
- [32] Vedere, per esempio, L. D. Landau and E. M. Lifschitz, *The classical theory of fields* (Pergamon, London, 1962).
- [33] D. R. Hamann, Phys. Rev. B **40**, 2980 (1989).
- [34] D. M. Ceperley and B. J. Alder, Phys. Rev. Lett. **45**, 566 (1980).
- [35] J. Perdew and A. Zunger, Phys. Rev. B **23**, 5048 (1981).
- [36] G. P. Kerker, J. Phys. C **13**, L189 (1980).
- [37] E. L. Shirley, D. C. Allan, R. M. Martin and J. D. Joannopoulos, Phys. Rev. B **40**, 3652 (1989).
- [38] A. Lastri, Tesi di laurea, Università di Trento, 1991, (non pubblicato).
- [39] A. Filippetti, A. Bosin, and G. B. Bachelet (non pubblicato).
- [40] G. B. Bachelet, D. R. Hamann and M. Schlüter, Phys. Rev. B **26**, 4199 (1982).
- [41] A. Bosin, Tesi di laurea, Università di Trento, 1990 (non pubblicato).
- [42] V. I. Smirnov, *A Course of Higher Mathematics*, (Pergamon, Oxford, 1964), Vol. 3 Part 2, p. 367.

-
- [43] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, *Numerical Recipes*, (Cambridge University Press, N.Y., 1986).
- [44] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller, J. Chem. Phys. **21**, 1087 (1953).
- [45] S. Kirkpatrick, C. D. Gelatt and M. P. Vecchi, Science **220**, 671 (1983).
- [46] P. J. M. van Laarhoven and E. H. L. Aarts, *Simulated Annealing: Theory and Applications*, (Reidel, Dordrecht, 1987).
- [47] L. Mitáš and G. B. Bachelet, 1989 (non pubblicato).
- [48] D. Vanderbilt and S. G. Louie, J. Comp. Phys. **56**, 259 (1984).
- [49] A. Bosin, Rapporto Tecnico CNR “Progetto Finalizzato Sistemi Informatici e Calcolo Parallelo” No. 1/157, 1993 (non pubblicato).
- [50] L. Mitáš and H. Mitášová, Comput. Math. Applic. **16**, 983 (1988).
- [51] A. Talmi and G. Gilat, J. Comp. Phys. **23**, 93 (1977).
- [52] S. G. Louie, S. Froyen and M. L. Cohen, Phys. Rev. B **26**, 1738 (1982).
- [53] U. von Barth and C. D. Gelatt, Phys. Rev. B **21**, 2222 (1980).
- [54] G. Galli, D. M. Ceperley and G. B. Bachelet, 1989 (non pubblicato).
- [55] M. Methfessel, Phys. Rev. B **38**, 1537 (1988).
- [56] V. Fiorentini, M. Methfessel and M. Scheffler, Phys. Rev. B **47**, 13353 (1993).
- [57] O. K. Andersen, O. Jepsen, and D. Glötzel, in *Highlights of Condensed Matter Theory*, F. Bassani, F. Fumi, and M. P. Tosi eds. (North-Holland, Amsterdam 1985).
- [58] H. J. Monkhorst and J. D. Pack, Phys. Rev. B **13**, 5188 (1976).
- [59] M. J. Gillan, J. Phys. Cond. Matter **1**, 689 (1989).

-
- [60] A. De Vita, comunicazione privata.
- [61] M. Methfessel and A. T. Paxton, Phys. Rev. B **40**, 3616 (1989).
- [62] R. J. Needs, R. M. Martin, and O. H. Nielsen, Phys. Rev. B **33** 3778 (1986).
- [63] C.-L. Fu and K.-M. Ho, Phys. Rev. B **28** 5480 (1983).
- [64] C. Filippi and C. J. Umrigar, 1994 (non pubblicato).
- [65] A. Continenza, comunicazione privata.
- [66] Landolt-Börnstein, Vol. 17a and 22a, edited by K. H. Hellwege and O. Madelung (Springer, Berlin, 1987).
- [67] A. Bosin (non pubblicato).
- [68] J. Ihm, A. Zunger and M. L. Cohen, J. Phys. C **12**, 4409 (1979); vedere anche corrigendum: J. Ihm, A. Zunger and M. L. Cohen, *ibid.* **13**, 3095 (1980)
- [69] C. J. Umrigar, K. G. Wilson and J. W. Wilkins, Phys. Rev. Lett. **60**, 1719 (1988).
- [70] C. J. Umrigar, M. P. Nightingale and K. J. Runge, J. Chem. Phys. **99**, 2865 (1993).
- [71] J. Casulleras and J. Boronat, 1994 (non pubblicato).
- [72] S. Zhang and M. H. Kalos, Phys. Rev. Lett. **67**, 3074 (1991).
- [73] Siu A. Chin, Phys. Rev. A **42**, 6991 (1990).
- [74] H. Kleinert, *Path Integrals* (World Scientific, 1990), p. 128.
- [75] T. Kato, Comm. Pure Appl. Math. **10**, 151 (1957).
- [76] J. H. Hetherington, Phys. Rev. A **30**, 2713 (1984).
- [77] C. J. Umrigar, Phys. Rev. Lett. **71**, 408 (1993).

-
- [78] *Handbook of Chemistry and Physics*, 69th ed., edited by R. C. Weast (CRC, Florida, 1988).
- [79] *Handbook of Physicochemical Properties of the Elements*, edited by G. V. Samsonov (Oldbourne, London, 1968).
- [80] Johnny Petersen, *Computer Physics Communications* **73**, 72 (1992).
- [81] Dennis W. Duke, *Computers in physics* **7**, 176 (1993).
- [82] Bernardo A. Huberman, *Computer Physics Communications* **56**, 25 (1989).
- [83] V. S. Sunderam, G. A. Geist, J. Dongarra, and R. Manchek (preprint).
- [84] Manuali del sistema SPARCstation 10 (Sun Microsystems, USA, 1990).
- [85] Louis H. Turcotte, *A Survey of Software Environments for Exploiting Networked Computing Resources*, Engineering Research Center for Computational Field Simulation (Mississippi, USA, 1993).
- [86] Jack Dongarra, G. A. Geist, Robert Manchek, and V. S. Sunderam, *Computers in Physics* **7**, 166 (1993). È disponibile tramite “anonymous ftp” sul nodo netlib.att.com.
- [87] A. Geist, A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, V. Sunderam, *PVM 3 user's guide and reference manual*, Oak Ridge National Laboratory (Tennessee, USA, 1993).
- [88] M. Menchi, *Tesi di laurea*, (Cagliari, 1992).
- [89] XPVM è stato sviluppato da James Arthur Kohl presso Oak Ridge National Laboratory (Tennessee, USA); esso è disponibile tramite “anonymous ftp” sul nodo netlib.att.com.