

Scuola di Scienze
Corso di Laurea Magistrale in Fisica

Approcci innovativi per future
simulazioni HPC verso l'exascale
computing con applicazioni alla fisica dei
materiali

Relatore:
Prof. Daniel Remondini

Presentata da:
Eric Pascolo

Correlatori:
Ph.D. Arrigo Calzolari
Ph.D. Carlo Cavazzoni
Prof. Renato Colle

Ad Adele
La mia piccola, grande donna che mi sopporta!

Abstract

In questo lavoro di tesi sono state evidenziate alcune problematiche relative alle macchine exascale (sistemi che sviluppano un exaflops di Potenza di calcolo) e all'evoluzione dei software che saranno eseguiti su questi sistemi, prendendo in esame principalmente la necessità del loro sviluppo, in quanto indispensabili per lo studio di problemi scientifici e tecnologici di più grandi dimensioni, con particolare attenzione alla Material Science, che è uno dei campi che ha avuto maggiori sviluppi grazie all'utilizzo di supercomputer, ed ad uno dei codici HPC più utilizzati in questo contesto: Quantum ESPRESSO. Dal punto di vista del software sono state presentate le prime misure di efficienza energetica su architettura ibrida grazie al prototipo di cluster EURORA sul software Quantum ESPRESSO. Queste misure sono le prime ad essere state pubblicate nel contesto software per la Material Science e serviranno come baseline per future ottimizzazioni basate sull'efficienza energetica. Nelle macchine exascale infatti uno dei requisiti per l'accesso sarà la capacità di essere energeticamente efficiente, così come oggi è un requisito la scalabilità del codice. Un altro aspetto molto importante, riguardante le macchine exascale, è la riduzione del numero di comunicazioni che riduce il costo energetico dell'algoritmo parallelo, poiché in questi nuovi sistemi costerà di più, da un punto di vista energetico, spostare i dati che calcolarli. Per tale motivo in questo lavoro sono state esposte una strategia, e la relativa implementazione, per aumentare la località dei dati in uno degli algoritmi più dispendiosi, dal punto di vista computazionale, in Quantum ESPRESSO: Fast Fourier Transform (FFT). Per portare i software attuali su una macchina exascale bisogna iniziare a testare la robustezza di tali software e i loro workflow su test case che stressino al massimo le macchine attualmente a disposizione. In questa tesi per testare il flusso di lavoro di Quantum ESPRESSO e WanT, un software per calcolo di trasporto, è stato caratterizzato un sistema scientificamente rilevante costituito da un cristallo di PDI - FCN2 che viene utilizzato per la costruzione di transistor organici OFET. Infine è stato simulato un dispositivo ideale costituito da due elettrodi in oro con al centro una singola molecola organica.

Indice

1	Verso l'exascale	9
1.1	Macchine Exascale	10
1.2	Le sfide dell'exascale	11
1.2.1	Energia	12
1.2.2	Concurrency	12
1.2.3	Memoria	12
1.2.4	Località dei dati	13
1.2.5	Resilience	14
1.3	Exascale e Material Science	14
2	Teoria e Codici utilizzati	17
2.1	Quantum ESPRESSO	17
2.1.1	Density Functional Theory	18
2.1.2	PWscf Algoritmica [16]	19
2.2	WanT	20
2.2.1	Teoria del Trasporto	21
2.3	Utilizzo di WanT e Quantum ESPRESSO	22
3	Misure di Efficienza Energetica su Architetture Ibride	23
3.1	Implementazione	23
3.2	Hardware	24
3.3	Set-up	25
3.4	Benchmark	25
3.4.1	Energy to solution	25
3.4.2	Analisi dei componenti	26
4	Studio e implementazione di strategie per l'aumento della località dei dati	31
4.1	Distribuzione G e FFT attuale	32
4.2	Nuova distribuzione dati gBlockDistro	33
4.3	Quantum Espresso Roadmap FFT	34
5	Applicazioni al calcolo di proprietà strutturali, elettroniche e di trasporto di materiali di interesse tecnologico	37
5.1	Stato dell'arte	38
5.2	Struttura e proprietà $PDI - FCN_2$ cristallino	39

5.3	Conto Bulk	40
5.4	Dispositivo a 2 terminali	44
6	Conclusioni	53
A	Attuale Distribuzione G in QE	57

Introduzione

Il mio progetto di tesi è stato molto lungo ed articolato, il tutto è iniziato nel maggio del 2013 presso il consorzio interuniversitario CINECA di Casalecchio di Reno (Bo). Qui mi sono avvicinato al mondo dell'HPC (High Performance Computing) ed alle sue problematiche, con particolare riferimento alla material science, ed ad uno dei codici HPC più utilizzati in questo contesto: Quantum ESPRESSO. Mi sono quindi concentrato sullo studio delle future macchine Exascale, che sono sistemi HPC con un ExaFlops di potenza di calcolo, e che si stima saranno disponibili per i ricercatori negli anni '20 di questo secolo. In questa tesi sono stati trattati due problemi riguardanti l'exascale, efficienza energetica e località dei dati, ed è stata realizzata una simulazione di material science, molto costosa dal punto di vista del calcolo per le attuali macchine, ma che si suppone sarà all'ordine del giorno per i futuri sistemi. La tesi si articola in 6 capitoli:

Verso l'exascale

In questo primo capitolo spiegherò cos'è un sistema exascale, quali sono i problemi di sistemi di questo tipo e quali sono le opportunità, dal punto di vista della material science, nell'utilizzo di queste macchine.

Teoria e codici utilizzati

In questo capitolo descriverò brevemente il codice e l'algoritmica di Quantum ESPRESSO, che è il software su cui ho lavorato per tutti i topic della tesi. Inoltre descriverò brevemente il software WanT, e la teoria che implementa, col quale ho realizzato la parte finale della simulazione del sistema fisico.

Misure di efficienza energetica su architetture ibride

In questa sezione riporto le misure di efficienza energetica che hanno portato alla pubblicazione dell'articolo [6] riguardante un benchmark di Quantum ESPRESSO eseguito su tre distinte architetture: nVIDIA K20, Intel Xeon Sandy Bridge e Intel Xeon Phi. Le misure di consumo energetico delle applicazioni in futuro saranno sempre più importanti poiché i centri di calcolo introdurranno un budget energetico affianco a quello già esistente di ore di calcolo.

Studio e implementazione di strategie per l'aumento della localizzazione dei dati

In questo capitolo ho esposto una strategia, e la relativa implementazione, per aumentare la località dei dati in Quantum ESPRESSO. Questa nuova strategia tende a ridurre il numero di comunicazioni tra i processori e quindi un minore

consumo energetico. Infatti prossime generazioni di macchine HPC avranno un costo energetico di comunicazione molto più elevato rispetto alle attuali.

Applicazioni al calcolo di proprietà strutturali, elettroniche e di trasporto di materiali di interesse tecnologico

Esporrò quindi i risultati della simulazione di Material Science, questa rappresenta ciò che sarà all'ordine del giorno per consumi e dimensioni computazionali nelle future macchine exascale, infatti è stata usata per testare la robustezza del flusso di lavoro che accoppia Quantum ESPRESSO a WanT. Il sistema preso in esame è di grande interesse sia scientifico che tecnologico, poiché è stato caratterizzato un cristallo di $PDI - FCN_2$ che viene utilizzato per la costruzione di transistor organici OFET. Infine si è riusciti a simulare anche un dispositivo ideale formato da due elettrodi in oro con al centro una singola molecola organica.

Capitolo 1

Verso l'exascale

Una delle sfide attuali nel campo dell'HPC¹ è riuscire a creare piattaforme innovative per le nuove generazioni di applicazioni che riescano a simulare, con performance sempre maggiori, sistemi a larga scala e a multi scala. Per far questo, oltre alla costruzione di nuove macchine ad elevato parallelismo, servirà trovare nuovi paradigmi di programmazione, nuovi algoritmi, nuovi tipi di device e nuove tecniche di gestione dell'infrastruttura e delle risorse di calcolo. Tutti questi aspetti del calcolo devono evolvere contemporaneamente, non basta agglomerare sempre più CPU all'interno di un singolo cluster. Si riporta un esempio per spiegarne il motivo.

Si supponga di avere una macchina "gigantesca" formata da alcune milioni di CPU tradizionali che sia in grado di produrre un enorme potenza di calcolo. Se questo dispositivo esistesse, scienziati e ingegneri potrebbero veder risolti molti dei loro problemi, ci sarebbero FLOPS per tutti, quindi ognuno opererebbe ad alta velocità. Emergerebbe subito tuttavia, un problema: l'alimentazione della macchina richiederebbe l'energia di una centrale nucleare, tenuto conto del consumo energetico di un processore moltiplicato per il numero totale. Migliorando perciò il consumo dei processori si abbasserebbero i costi del mantenimento della macchina a tutto vantaggio degli utenti del centro di calcolo. I problemi però non finirebbero qui. Gli utilizzatori della macchina si accorgerebbero ben presto che in realtà i loro programmi non andrebbero più veloci, non scalerebbero, aumentando il numero dei processori. Un secondo tentativo potrebbe essere quello di utilizzare sistemi più grandi, ma anche con questa tecnica, paradossalmente, arrivando a un certo numero di processori il sistema rallenterebbe. I gestori dei centri di calcolo si renderebbero conto ben presto che pur avendo la prima macchina per FLOPS al mondo, le statistiche di utilizzo e consumo non sarebbero favorevoli. In questo tipo di applicazioni parallele, infatti, molti processi rimangono in idle, aspettando che gli altri terminino. Ma aspettando consumano energia e elevano i costi senza tenere conto dell'elevata spesa che richiede il raffreddamento di quei pochi m³ d'aria della sala macchine. In pochi mesi ci si renderebbe quindi conto che il rapporto

¹High Performance Computer

costi benefici è totalmente spostato verso i costi e che il super computer non ha le performance ottenibili promesse. A questo punto le proposte dei vari addetti ai lavori per arrivare alla soluzione dei problemi emersi possono essere molteplici:

- *l'hardware dovrebbe essere rimodernato, i processori resi più efficienti e dovrebbero inoltre adattarsi alle applicazioni*
- *si potrebbero creare supercomputer eterogenei con diversi tipi di processori*
- *migliorare le tecnologie di raffreddamento o cambiarne proprio il principio*

Si ipotizzano inoltre soluzioni relative al problema della scalabilità del codice che riguarda oltre al paradigma di parallelismo anche l'algoritmo usato. Si potrebbero inserire anche nel codice delle direttive sulla selezione dell'hardware sul quale deve essere eseguito il programma, o spegnere l'hardware se il programma non lo sta usando o cambiarne la frequenza. Occorrerebbe anche risolvere il problema del costo energetico, monetario e di comunicazione della rete di interconnessione tra le unità di calcolo: non è possibile che tutti siano collegati con tutti e allo stesso modo anche le applicazioni dovrebbero tenere conto di questi aspetti per evitare di perdere tempo a spostare i dati all'interno della macchina. Le idee risolutive che riguardano l'ecosistema dei supercomputer sono tante, ed è chiaro come, per fare un salto di qualità nelle simulazioni la strada da fare sia ancora lunga e il lavoro da svolgere ancora molto.

Questo esempio dovrebbe aver chiarito quanto è complessa la tecnologia HPC e come, per ottenere un buon risultato, si devono far evolvere insieme le tecnologie dell'intero ecosistema.

1.1 Macchine Exascale

Prima di analizzare l'exascale bisogna definirlo. Per definire una macchina exascale è indispensabile definirne i suoi attributi fisici, le sue applicazioni e le performance delle macchine e delle applicazioni[5]. Gli attributi da prendere in considerazione sono:

- costo: deve tenere conto del costo iniziale della macchina e del suo mantenimento;
- dimensioni: area e volume occupati;
- potenza assorbita totale.

Le performance della macchina sono:

- flops: numero di operazioni in virgola mobile al secondo;
- capacità di memoria: quanta memoria è disponibile nelle diverse forme:
 - memoria principale: luogo in cui vengono immagazzinati i dati a runtime e dove l'applicazione può accedere direttamente, ad esempio la ram;

- scratch: area di memoria a bassa latenza dove possono essere memorizzati file temporanei di input output e checkpoint del software durante l'esecuzione;
- memoria persistente: memoria a bassa velocità dove vengono salvati i file di input ed output delle simulazioni;
- larghezza di banda: indica quanti dati al secondo si possono muovere all'interno del sistema. Si possono avere diverse misure a seconda della parte di sistema che si prende in considerazione.

Altri due parametri di performance della macchina sono la capability e la capacity. La prima identifica il picco di calcolo utilizzabile da una singola applicazione; l'altra identifica la capacità del sistema di riuscire a far girare un grande numero di jobs in contemporanea. Entrambi i parametri, data la giusta tipologia di applicazioni, più grandi sono più riducono il time to solution. Ci si riferisce al parametro di capacity quando vi è un'applicazione che deve esplorare uno spazio parametrico o differenti soluzioni, in questo caso la capacity esprime la possibilità di eseguire più istanze dell'applicazione con parametri diversi e quindi permette di esplorare tutto lo spazio nel minor tempo possibile. Per quanto riguarda il parametro capability, questo viene preso in considerazione quando l'applicazione deve risolvere un problema nel minor tempo possibile. In questo caso si ha a che fare con applicazioni dotate di un grande dataset di input oppure con applicazioni che hanno vincoli di elaborazione in tempo reale.

È ora possibile definire cos'è una macchina exascale. Si tratta di una macchina che possiede 10^{18} FLOPS, ovvero 1 EXAFLOPS, secondo lo standard IEEE754 double precision. Il benchmark con cui si ottiene questa prestazione è l'*High Performance Linpack* (HPL) [10], per eseguirlo in un tempo ragionevole un sistema exascale ha bisogno di circa 10^{16} - 10^{17} Byte di memoria principale, ovvero 1 PetaByte e una larghezza di banda di 10^{18} Byte al secondo. Per immagazzinare tutti i file e dati intermedi che possono essere generati da queste macchine si presuppone che debba esserci 1 exaByte di memoria persistente. Dal punto di vista delle applicazioni, su una macchina exascale, queste avranno una capability elevata poiché il set di dati sarà suddivisibile su molte unità di calcolo, ma all'interno dell'unità di calcolo la capacity potrebbe non essere sufficiente per contenere copie dello stesso problema. Per quanto concerne gli attributi fisici, è stato stimato che le macchine exascale, per essere economicamente vantaggiose, dovranno avere una dimensione di massimo 600 rack, una potenza assorbita di 20 MW e un costo complessivo minore di 200 milioni di dollari [25].

1.2 Le sfide dell'exascale

Per costruire una macchina con queste caratteristiche bisogna affrontare delle sfide già illustrate nell'esempio del paragrafo di introduzione di questo elaborato. [13]

1.2.1 Energia

Il limite di consumo totale per queste macchine dovrà essere dell'ordine di 20MW di TDP (Termal Design Point), non è possibile raggiungere questo obiettivo, se si considera un'evoluzione dell'hardware che ricalca quella avvenuta fino ad oggi. Di conseguenza le architetture subiranno dei drastici cambiamenti: per esempio si dovrà ridurre notevolmente il consumo di energia per unità di calcolo avendo un buon bilanciamento tra performance, capacità di memoria e costo energetico per flop. Dal punto di vista algoritmico, sarà sempre più importante misurare l'energy to solution per contenere i costi indotti del funzionamento del sistema (TCO), a tal punto che per contabilizzare l'uso del calcolatore affianco del budget ore calcolo vi sarà un budget energetico. Vista questa nuova tendenza si dovranno creare API per la gestione energetica dell'hardware via software [1] [34], in modo tale che il singolo utente possa gestirsi in autonomia il suo budget energetico; quindi, ad esempio, se un utente dovesse avere la necessità di minimizzare il time to solution potrà decidere di aumentare la frequenza dei processori essendo consapevole di massimizzare l'energy to solution e pagarne i costi conseguenti.

1.2.2 Concurrency

Per limitare i consumi, già adesso, i produttori di processori stanno creando nuovi prodotti con frequenza di clock più bassa rispetto a quella del passato e un maggiore numero di core; ciò implica che per diminuire il time to solution l'unica strada percorribile è la parallelizzazione del software. Fino a qualche anno fa vi erano due livelli di parallelizzazione[29]: uno intranodo e l'altro tra tutti i nodi che compongono il sistema HPC. Ciò non è sufficiente a garantire i livelli di prestazione necessari alle moderne architetture e quindi è stata reintrodotta una tecnologia di successo degli anni 80: la vettorizzazione [15]. All'epoca era a livello di sistema, oggi è stata sfruttata a livello di core andando a creare un ulteriore livello di parallelizzazione intracore. Nelle future macchine exascale sarà possibile avere molti più livelli di parallelismo e quindi gli algoritmi dovranno riuscire ad adattarsi a questi nuovi schemi[31]. Accanto all'introduzione del livello intercore, si sta sviluppando anche la tendenza di utilizzare acceleratori per aumentare le prestazioni del singolo nodo; questi spingono all'estremo il concetto di parallelizzazione intracore, per cui sono in grado di eseguire centinaia, o addirittura migliaia, di calcoli contemporaneamente. Per le macchine che ospitano questa tecnologia, si parla di architetture ibride.

1.2.3 Memoria

Se non avverrà un significativo cambiamento nelle architetture delle memorie, a causa dei limiti energetici precedentemente citati [25], avremo un aumento complessivo della memoria principale, ma non tale da garantire un eguale aumento della memoria per singolo core, quindi di fatto avverrà una diminuzione della

memoria di singolo core, poiché il numero di core per raggiungere la potenza computazionale di un exaflop dovrà crescere molto di più della memoria. Di conseguenza si dovranno adattare gli algoritmi a questi nuovi vincoli: ovvero si dovrà minimizzare l'uso della memoria.

1.2.4 Località dei dati

Per riuscire a comprendere meglio perché è importante mantenere i dati locali bisogna andare a indagare i costi energetici del calcolo, del mantenimento in memoria e dello spostamento di questi ultimi. Nei processori attuali una delle unità di elaborazione che consuma maggiore energia è la FMA (Fusion Multiple Add in doppia precisione): per ciascun calcolo consuma circa $100pJ$; mentre, per quanto riguarda la memoria, la sola operazione di lettura dei dati consuma almeno duecento volte tanto, per una lettura in DRAM si ha un costo di $2000pJ$ [1]. Per le prime macchine exascale è previsto che il costo di FMA diminuisca fino a $\sim 10pJ$ per operazione in doppia precisione, mentre per quanto riguarda la lettura di un dato in memoria ci sarà solamente un leggero miglioramento e si arriverà a $1000pJ$ 1.1. Se questi dati saranno veri si impiegheranno circa $100W$ per sviluppare $10TFLOPS$ e circa $2000W$ per le operazioni sulla memoria. Quindi il vincolo principale nel progettare i nuovi nodi per sistemi exascale non è la potenza di calcolo, ma il vincolo energetico sulla memoria.

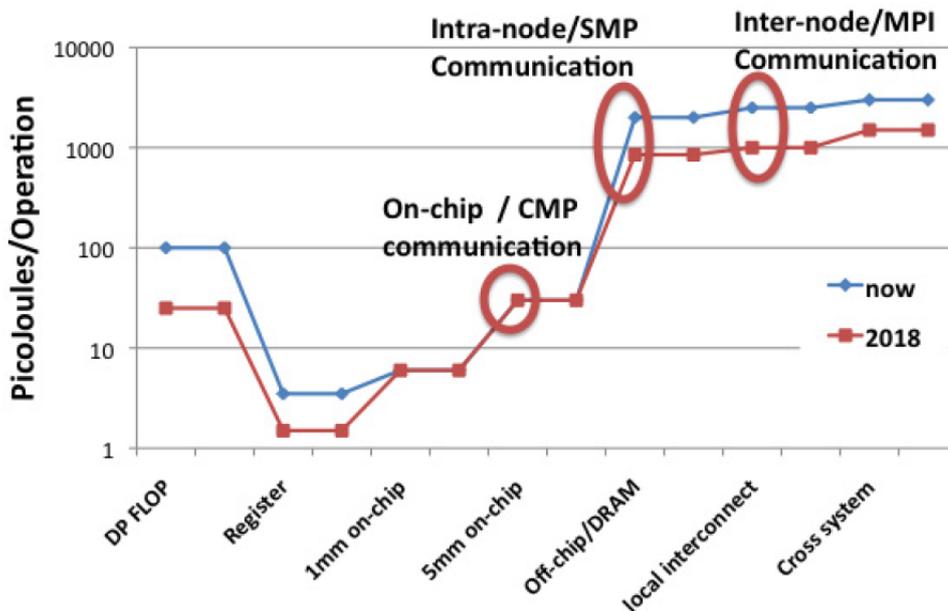


Figura 1.1: Grafico riportante il costo energetico dello spostamento dati tra differenti livelli di memoria [1]

Per quanto riguarda le comunicazioni nodo a nodo il problema è che le attuali linee di comunicazione all'interno di un sistema HPC sono tutte basate su segnali elettrici trasportati in fili di rame. Se si considera una linea elettrica convenzionale,

è possibile vederla come un circuito RC, quindi il filo sarà caricato e scaricato a un ritmo governato dalla costante di tempo RC e dalla lunghezza l del filo secondo la formula: $T = RC l^2$. Quindi quando la lunghezza del filo aumenta, aumenta anche T , ovvero il tempo di salita e quindi il bitrate cala [22]. Per diminuire il tempo di salita, si potrebbe diminuire la sezione del filo, così facendo diminuirebbe la resistenza, ma comunque la capacità rimarrebbe costante. Data l'equazione 1.1 dove B è la larghezza di banda, l la lunghezza del filo e A è la sezione del filo, si può affermare che più si aumenta il bit-rate, più cresceranno i consumi a parità di l ed A ; più si aumenta la distanza a cui si vuole portare il segnale, più la potenza crescerà quadraticamente e se si riesce a creare fili con sezioni più piccole questi non avranno miglior conducibilità, né migliore efficienza energetica. Un campo sul quale si sta investigando negli ultimi anni è quello di costruire la rete di interconnessione del supercalcolatore interamente a trasmissione ottica tra nodi, inclusi i vari livelli di switch e di circuiti stampato.

$$Power \simeq B \times l^2 / A \quad (1.1)$$

Per riuscire a far fronte a questo problema l'unica cosa che si può fare, senza sperare in un miglioramento energetico dell'hardware, è riuscire a modificare gli algoritmi minimizzando le comunicazioni e rendendo i dati il più locali possibile.

1.2.5 Resilience

Considerato l'aumento del numero di componenti in questi sistemi, ci si aspetta anche un aumento del numero di guasti. Per ovviare a questi inconvenienti occorre che le applicazioni adottino delle contromisure, una delle più note è il meccanismo di checkpoint e restart. L'attuale meccanismo di checkpoint, solitamente, è molto costoso sia in termini di tempo che in termini di energia e questo è dovuto alla sincronizzazione di tutte le unità di calcolo e dall'input output sul file system. Nelle nuove macchine, teoricamente, il tempo di checkpoint potrebbe essere più lungo del tempo che intercorre tra due rotture del sistema rendendo il restart inutilizzabile. Questo problema dovrà essere affrontato sia dal punto di vista architetturale che da quello software e algoritmico.

1.3 Exascale e Material Science

Uno dei campi che si pensa che avrà uno sviluppo rapido e notevole con l'uso di sistemi Exascale è la Material Science. Con questo termine si racchiudono tutte quelle tecniche scientifiche ed ingegneristiche che permettono la scoperta, la progettazione e l'ottimizzazione di nuovi materiali. Un materiale può essere schematizzato come in figura 1.2, quello che si vuole ottenere è la sua caratterizzazione, ossia riuscire a capire la sua struttura, le sue proprietà e se lo sviluppo e le proprietà sono adatte al processo di produzione e alle sue applicazioni. Il processo di creazione di un nuovo materiale per le aziende è uno dei più dispendiosi in

termini economici e temporali, poiché spesso si deve ricorrere al processo “prova ed errore” e per ognuno di questi tentativi può essere necessario qualche mese, questo può portare ad un intervallo di 15/20 anni dalla fase di progettazione in laboratorio a quella di commercializzazione del nuovo materiale [28]. Il vantaggio che si ha con l’uso di una macchina HPC è di non dover testare in laboratorio ogni singolo materiale; infatti, creando un database con i composti da testare si possono simulare se questi hanno le proprietà richieste ed evitare di creare il prototipo e testarle fisicamente. Avendo a disposizione un sistema Exascale con una grande capacity è possibile effettuare parallelamente le simulazioni sui composti nel database, in questo modo si dà uno speedup notevole allo sviluppo del nuovo materiale poiché si testano solo quei pochi materiali che sono risultati positivi ai test simulati [7]. Questo approccio appena citato è detto *High Throughput Material Science* e vi sono progetti che supportano questo tipo di approccio e rendono disponibili dei database [17]. Dato il framework di parallelizzazione delle simulazioni che sfrutta la capacity del sistema Exascale, si deve riuscire, all’interno della singola simulazione, a sfruttare la capability massima del sottosistema assegnato e soprattutto a impostare il problema di progettazione o testing del materiale dal punto di vista matematico. Il background matematico e lo sfruttamento della capability massima, per quanto riguarda la Material Science, sono una sfida per macchine exascale poiché queste promettono, con le loro prestazioni, di riuscire a studiare sistemi a multiscale e a multirisoluzione. Questo significa che per ogni scala spaziale e risoluzione in cui si vuole studiare il materiale, si deve avere un modello fisico matematico differente che permetta la soluzione delle equazioni e una teoria che riesca ad accoppiarli. Ad esempio, a scala nanoscopica si dovrà utilizzare la meccanica quantistica, ingrandendo la scala si passerà alla meccanica statistica e infine alla meccanica molecolare classica, fino ad arrivare alla meccanica del continuo. Allo stesso tempo, all’interno del singolo modello si dovrà riuscire ad avere un algoritmo ottimizzato che rispetti i vincoli citati nella sezione precedente, ovvero che massimizzi la capability del sottosistema, che per simulazioni su sistemi exascale dovrà essere di centinaia di migliaia di unità di calcolo.

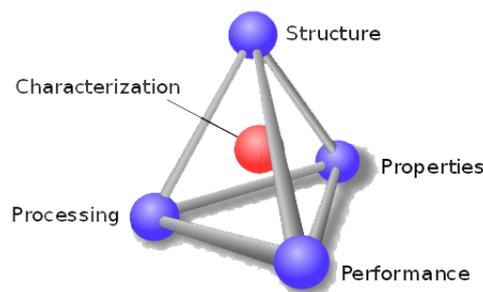


Figura 1.2: Schematizzazione di un materiale (Credits: wikipedia)

Capitolo 2

Teoria e Codici utilizzati

In questo capitolo si presenteranno i codici su cui si è lavorato durante il progetto di tesi, le teoria su cui si basano e il workflow utilizzato. Il software Quantum ESPRESSO è stato il utilizzato per effettuare le misure di efficienza energetica riportate nel capitolo 3, ed è il software su cui si è effettuato lo studio e la implementazione di cambiamento della distribuzione dati riportata nel capitolo 4, infine lo si è utilizzato per effettuare la preparazione del sistema, riportato nel capitolo 5, per il conto di conduttanza che successivamente è stato eseguito con il secondo software qua descritto WanT.

2.1 Quantum ESPRESSO

Quantum ESPRESSO (*opEn Source Package for Research in Electronic Structure, Simulation and Optimization*) [16] è una suite integrata di programmi per macchine HPC per il calcolo delle strutture elettroniche e la progettazione di materiali dell'ordine della nanoscala basato sulla Density Functional Theory. Il punto di forza di un software come Quantum ESPRESSO (QE) è che non si rivolge semplicemente a una piccolo gruppo di ricercatori nel campo della fisica della materia, ma ha l'interesse e il sostegno di ricercatori che provengono da tutti i campi riguardanti ricerca e ingegnerizzazione di nuovi materiali. QE non è un programma monolitico, ma è suddiviso in varie applicazioni, alcune di calcolo, altre di processing che servono per elaborare e visualizzare i risultati. I pacchetti più importanti, che sono il cuore del software, sono:

PWscf permette vari tipi di calcoli self-consistenti della struttura elettronica utilizzando la teoria DFT su un set di base di onde piane e degli pseudo potenziali per la descrizione dell'interazione elettronica.

CP questo modulo utilizza gli stessi strumenti su cui si basa PWScf e implementa le stesse funzionalità, ma aggiungendo la dinamica molecolare Car-Parrinello.

2.1.1 Density Functional Theory

Poiché il pacchetto Pwscf è basato sulla teoria DFT, si riporta qualche accenno teorico. La Density Functional Theory è una teoria di ground-state nella quale la quantità fisica principale è la densità di carica. Questa teoria ha grande successo nel descrivere la struttura e le proprietà elettroniche di vari materiali, dai semplici cristalli a sistemi più complessi ed estesi.

Teorema di HK e sue conseguenze

Teorema di Hohenberg-Kohn

Si consideri un sistema di N elettroni interagenti sottoposti ad un potenziale esterno $V(r)$. Per un guscio chiuso di elettroni non interagenti si può affermare che esisterà solo un potenziale $V(r)$ tale per cui si avrà una carica di ground state pari a $n(r)$.

Una conseguenza immediata del teorema è che l'energia di ground state è univocamente determinata dalla densità di carica del ground state stesso. Matematicamente si può dire che l'energia $E[n(r)]$ è un funzionale della densità di carica, ovvero:

$$E[n(r)] = F[n(r)] + \int n(r)V(r) dr \quad (2.1)$$

Quindi con la DFT è possibile ridurre un problema ad N corpi ad un problema di determinazione della funzione $n(r)$ in tre dimensioni, questa minimizzerà il funzionale energia.

Kohn-Sham equations

Kohn-Sham riformularono la DFT in modo da riuscire ad usarla per applicazioni pratiche. Un sistema di elettroni interagenti può essere descritto con l'ausilio di un sistema di elettroni non interagenti aventi la stessa densità di carica elettronica di ground state. Per il sistema di elettroni non interagenti la $n(r)$ è definibile come somma degli orbitali a un elettrone (detti *KS orbital*) $\Psi_i(r)$:

$$n(r) = 2 \sum_{i=1}^{\frac{N}{2}} |\Psi_i(r)|^2 \quad (2.2)$$

dove i va da 1 a $\frac{N}{2}$, poichè si è assunta una doppia occupazione di ogni stato. Gli orbitali KS sono soluzione dell'equazione di Schrodinger :

$$\left(-\frac{\hbar^2}{2m_e} \nabla^2 + V_{KS}(r) \right) \Psi_i(r) = \epsilon_i \Psi_i \quad \implies \quad \mathcal{H}_{KS} \Psi_i(r) = \epsilon_i \Psi_i \quad (2.3)$$

e imponendo il vincolo di ortonormalizzazione si ha che

$$\int \Psi_i^*(r) \Psi_j(r) dr = \delta_{ij} \quad (2.4)$$

L'unicità del potenziale V_{KS} che compare nella (2.3) per il teorema HK garantisce l'esistenza e l'unicità di $n(r)$, qualunque sia l'interazione elettrone-elettrone.

2.1.2 PWscf Algoritmica [16]

Il problema di trovare la soluzione self consistente delle equazioni KS può essere trasformato nella soluzione di un problema non lineare:

$$x^o = F[x^i]x = (x_1, \dots, x_N) \quad (2.5)$$

dove il vettore x^i contiene le N componenti, in spazio reale o in spazio di Fourier, della densità di carica ρ o del potenziale KS V e $F[x^i]$ è il loro funzionale, infine x^o è il vettore di output. La soluzione può essere ottenuta con un metodo iterativo, quello utilizzato da Pwscf è basato sul metodo Broyden [18] nel quale si ha che x^o contiene le componenti della densità di carica in spazio reciproco. Vengono utilizzati un mix di algoritmi per trovare un valore ottimale di x^i dalle combinazioni delle iterazioni precedenti che minimizzi la norma $\|x^o - x^i\|$, la quale si annulla a convergenza ed è detta *norma scf*. Questa fornisce una misura dell'errore self-consistente sull'energia totale del sistema in esame. Si prenda come esempio un isolante con uno pseudopotenziale norm-conserving con uno scambio correlazione LDA. Ad una data iterazione si avrà:

$$\left(-\frac{\hbar^2}{2m} + V_{ext}(r) + V^{in}(r) \right) \psi_i(r) = \epsilon_i \psi_i(r) \quad (2.6)$$

dove ϵ_i e ψ_i sono l'energia dell'orbitale e la funzione d'onda KS, mentre V_{ext} è il potenziale che rappresenta la somma degli pseudopotenziali di core, infine $V^{in}(r) = V_{Hxc}[\rho^{in}(r)]$ è il potenziale di Hartree e di scambio correlazione, che è funzione della densità di carica ρ . La densità di carica di output sarà data da

$$\rho^o(r) = \sum_i |\psi_i(r)|^2 \quad (2.7)$$

Se si compara l'energia calcolata in DFT nella modalità standard

$$E = \sum_i \int \psi_i^*(r) \left(\frac{\hbar^2}{2m} \nabla^2 + V_{ext}(r) + V^{in}(r) \right) \psi_i(r) + E_{Hxc}[\rho^{out}] \quad (2.8)$$

con la forma funzionale di Harris-Weinert-Foulkes, dove non compare il termine ρ^{out} ,

$$E' = \sum_i \int \psi_i^*(r) \left(\frac{\hbar^2}{2m} \nabla^2 + V_{ext}(r) + V^{in}(r) \right) \psi_i(r) - \int \rho^{in} V^{in}(r) + E_{Hxc}[\rho^{in}] \quad (2.9)$$

si ha che tutte e due sono forme variazionali, infatti al primo ordine ambedue si annullano in funzione della densità di carica ed entrambe convergono allo stesso risultato quando giungono a convergenza. Le differenze tra queste due formule possono essere scritte come nell'equazione seguente (dove solo il termine di Hartree è considerato):

$$E - E' \simeq \frac{1}{2} \int \frac{\Delta\rho(r)\Delta\rho(r')}{|r - r'|} = \frac{1}{2} \int \Delta\rho(r)\Delta V_H(r')dr \quad (2.10)$$

dove $\Delta\rho = \rho^{out} - \rho^{in}$ e ΔV_H è il potenziale di Hartree generato da $\Delta\rho$. Inoltre è possibile mostrare come, quando il contributo elettronico di scambio correlazione non è dominante rispetto al termine elettrostatico, questa quantità rappresenta il limite superiore all'errore di self consistenza in cui si incorre quando si utilizza la forma standard per il calcolo dell'energia DFT. Quindi in spazio reciproco si può calcolare facilmente questo termine, che non è altro che il quadrato della norma scf:

$$\| \rho^{out} - \rho^{in} \|^2 = \frac{4\pi e^2}{\Omega} \sum_G \frac{|\Delta\rho(G)|}{G^2} \quad (2.11)$$

dove G sono i vettori nello spazio reciproco e Ω è il volume della cella unitaria. Durante il procedimento iterativo si deve risolvere un problema agli autovalori per tutti gli N stati occupati $H\psi_i = \epsilon_i S\psi_i$ con $i = 1, \dots, N$ dove H è l'Hamiltoniana del sistema e S è la matrice di overlap e gli autovettori devono essere ortonormali. La difficoltà sta nel risolvere questo problema con un metodo iterativo e per far ciò Pwscf implementa l'algoritmo di *diagonalizzazione a blocchi di Davidson*.

Se si profilasse un'esecuzione di QE, si vedrebbe che gli algoritmi in cui viene speso maggior tempo sono due ed è su questi che si cerca di ottenere un miglioramento continuo delle performance:

FFT L'algoritmo Fast Fourier Transform viene utilizzato migliaia di volte in ogni singolo lancio di QE, poiché è fondamentale passare dallo spazio reale al reciproco o viceversa, ad esempio durante la diagonalizzazione.

GEMM l'algoritmo di moltiplicazione tra due matrici viene utilizzato ogni qual volta che si devono calcolare le quantità fisiche, come la densità di carica ed i potenziali.

2.2 WanT

WanT[27] (Wannier Transport) è un software open-source per lo studio del trasporto elettronico coerente in nanostrutture estese a bassa dimensionalità. Il software è basato sulla combinazione della teoria DFT con metodi a onde piane ed i metodi di calcolo della funzione di Green basati sul formalismo di Landauer; calcola la conducibilità quantistica.

2.2.1 Teoria del Trasporto

Si consideri un sistema composto da un conduttore C, connesso a due contatti semi-infiniti, R e L, si veda la figura 2.1. Un risultato molto importante è la seguente equazione che lega la conduttanza G , alla trasmittanza T all'energia di Fermi E_f :

$$G = \frac{2e^2}{h} T(E_f) \quad (2.12)$$



Figura 2.1: Schema ideale del sistema per il calcolo di conduttanza

L'equazione 2.12 è nota come formula di Landauer per il trasporto coerente in nanostrutture[11] ed esprime la probabilità che un elettrone inserito da uno dei due lati del conduttore arrivi dalla parte opposta. Se si considera un sistema fisico rappresentato dall'Hamiltoniana H è possibile descrivere il trasporto tramite la matrice di scattering S , che fornisce la probabilità di interazione dell'elettrone con il conduttore. Il calcolo con la matrice di scattering però può essere semplificato esprimendo quest'ultima in termini della funzione di Green G , quindi si può scrivere che:

$$(\omega \pm i\eta - H)G(\omega) = I \quad (2.13)$$

dove I è l'operatore identità e $i\eta$ è un termine immaginario infinitesimo che è aggiunto all'energia per incorporare le condizioni al contorno. La trasmittanza può essere espressa in termini della funzione di Green del conduttore e del termine di accoppiamento conduttore-contatto, questa relazione la si può scrivere facilmente attraverso la formula di Fisher e Lee:

$$T(\omega) = Tr[\Gamma_L G_C^r \Gamma_R G_C^a] \quad (2.14)$$

dove $G_C^{r,a}$ sono le funzioni di Green del conduttore, G^r sarà la soluzione negativa, mentre G^a sarà quella positiva. $\Gamma_{L,R}$ sono le funzioni che descrivono l'accoppiamento conduttore-contatto. Gli effetti dovuti ai contatti semi-infiniti sul conduttore possono essere descritti da operatori finiti detti di auto-energia $\Sigma_{L,R}$. Il termine di auto-energia può essere visto come un effettivo Hamiltoniano che tiene conto dell'interazione conduttore-filo. La funzione di Green G_C e $\Gamma_{L,R}$ possono essere ottenuti esplicitamente dall'auto-energia tramite:

$$G_C = (\epsilon - H_C - \Sigma_L - \Sigma_R)^{-1}, \quad (2.15)$$

$$\Gamma_{\{L,R\}} = i[\Sigma_{\{L,R\}}^r - \Sigma_{\{L,R\}}^a]. \quad (2.16)$$

dove H_C è l'Hamiltoniana del conduttore, calcolata rispetto al set di basi localizzate in spazio reale. L'espressione per l'auto-energia può essere scritta come:

$$\begin{aligned} \Sigma_L &= H_{LC}^\dagger (\epsilon - H_{00}^L - (H_{01}^L)^\dagger \bar{T}_L)^{-1} H_{LC}, \\ \Sigma_R &= H_{CR} (\epsilon - H_{00}^R - H_{01}^R T_R)^{-1} H_{CR}^\dagger, \end{aligned} \quad (2.17)$$

dove $T_{L,R}$ e $\bar{T}_{L,R}$ sono le matrici di trasferimento che possono essere calcolate tramite un procedimento iterativo. L'accuratezza del risultato dipende principalmente da due fattori:

- l'accuratezza della descrizione della struttura elettronica del sistema;
- la base con cui è descritta l'Hamiltoniana deve essere ben localizzata.

Per quanto riguarda il secondo punto, lo schema che si utilizza per descrivere la base è una combinazione lineare di orbitali atomici gli *pseudo atomic orbitals* detti PAOs $\phi(r)$ [2].

2.3 Utilizzo di WanT e Quantum ESPRESSO

Per il calcolo della struttura elettronica, prima, e, successivamente, della curva di trasmittanza si possono usare i pacchetti di Quantum ESPRESSO e Want accoppiati. Una classica esecuzione di questo tipo segue il seguente schema:

QE Pwscf RELAX: minimizza l'energia e le forze sul sistema;

QE Pwscf SCF: ottimizza la struttura elettronica;

QE Pwscf NSCF: esegue il conto su una griglia di punti K sulla zona completa di Brillouin, da questo conto, ad esempio, si possono ottenere le DOS (density of state) o la struttura a bande;

QE Projwfc ottiene la base in termini di PAOs;

Want Bands reinterpolata la struttura a bande, la quale deve essere consistente con quella ottenuta dal conto NSCF;

Want Conductor esegue il conto di conducibilità.

Capitolo 3

Misure di Efficienza Energetica su Architetture Ibride

Come già detto nella sezione 1.2.1, negli ultimi anni, in vista dei sistemi Exascale e in particolare delle architetture eterogenee, il problema dell'efficienza energetica [19] [12] è stato preso sempre più seriamente. Alcuni studi hanno già trattato il problema prendendo come test simulazioni che si basano su PDE [30] e CFD[23], ma nel campo della Material Science sono stati condotti pochi test. In questo capitolo si esporranno i risultati che hanno permesso la pubblicazione dell'articolo [6]. Grazie al prototipo di supercomputer EURORA è stato possibile testare differenti versioni di Quantum ESPRESSO (QE) per acceleratori e per processori sul medesimo test case scientifico. Grazie ad un innovativo sistema di misure si è riusciti ad ottenere tali risultati sia in termini di performance che energetici.

3.1 Implementazione

Il porting di Quantum ESPRESSO sugli acceleratori è cominciato da circa tre anni fa e inizialmente l'idea era quella di utilizzare librerie già esistenti per portare codice sulla GPU senza dover cambiare la maggior parte del codice. Sono state quindi sostituite le chiamate GEMM con un plugin phiGEMM¹ che spostasse una parte del carico di lavoro sulla GPU. phiGEMM tenta di ottimizzare il carico di lavoro tra GPU e CPU mascherando i tempi di latenza dovuti alla comunicazione sul PCI. Il porting su GPU attualmente è quindi limitato solo alla parte GEMM. Negli ultimi mesi è stata avviata un'attività di sviluppo per portare Quantum ESPRESSO sulla piattaforma Intel Xeon Phi(CINECA ha intrapreso questa iniziativa poiché ha preso parte al programma “Intel Parallel Computing Center”). Seguendo lo stesso approccio usato per le GPU, si è cercato di portare in offload le chiamate GEMM, in particolare le ZGEMM, sul coprocessore. A differenza delle GPU, l'intera computazione delle chiamate GEMM è sul coprocessore, l'unico compito lasciato alla CPU è il ridimensionamento delle matrici prima del

¹qe-forge.org/gf/project/phigemmm/

trasferimento. L'uso della tecnica di double buffering 3.1 permette di mascherare parzialmente la latenza dei tempi di trasferimento, poiché si riesce a fare il reshape e l'invio di una matrice durante la computazione di un'altra. La differenza tra i due approcci sta anche nel fatto che per la GPU un parametro molto importante che può massimizzare le performance è il rapporto tra il carico della CPU e della GPU. Per lo Xeon Phi, nel quale tutta la computazione è assegnata al coprocessore, si può invece regolare solamente la soglia delle matrici che vengono trasferite tra host e device.

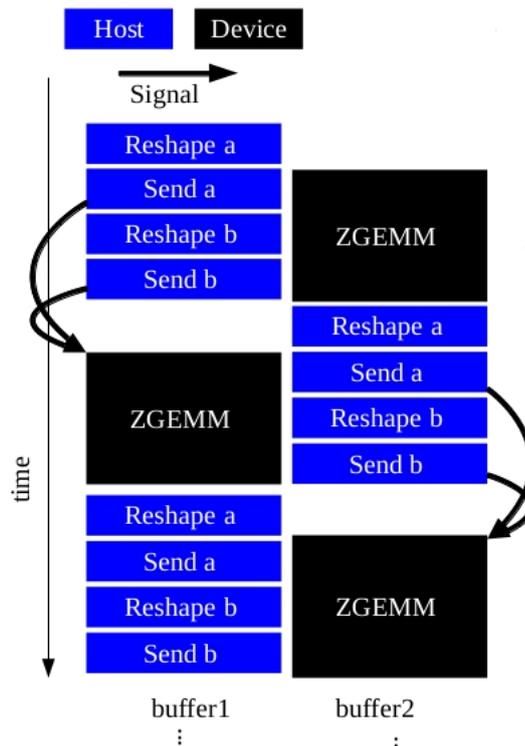


Figura 3.1: Schema del meccanismo di double buffering della versione QE per Xeon Phi

3.2 Hardware

Il Cluster EURORA è equipaggiato da due tipi di acceleratori: nVIDIA K20 GPUs e Intel Xeon Phi; questo rappresenta il prototipo perfetto di una macchina ibrida e quindi dove poter testare differenti versioni di QE. EURORA è un cluster in un unico rack, è stato costruito da Eurotech e sponsorizzato in parte dal progetto PRACE-2IP. Nel cluster sono presenti 64 nodi ognuno equipaggiato da un doppio socket Sandy Bridge e da due acceleratori. Grazie all'innovativo sistema di cooling EURORA nel giugno 2013 si è classificato primo nella classifica Green 500² con

²www.green500.org/lists/green201306

Device	Ets mean (Kj)	Time(sec)
<i>GPU + CPU - 3G</i>	145 ± 9	460 ± 4
<i>PHI + CPU - 2G</i>	105.86 ± 0.01	688 ± 1
<i>CPU - 2G</i>	112 ± 5	859 ± 3
<i>CPU - 3G</i>	136 ± 7	561 ± 2

Tabella 3.1: Time-to-solution e energy-to-solution ottenuti per il test case SiO₂. I risultati sono mostrati per CPU-2G, CPU-3G da soli e con Xeon-Phi o nVIDIA GPU

una performance di 3.150 MFlops/W. Su EURORA vi sono due tipi di nodo, quelli con acceleratori Xeon Phi a cui sono associati processori E5-2658 Xeon Sandy Bridge da 2.10 GHz (che da qui verranno chiamati CPU-2G) e quelli con acceleratori nVIDIA K20 a cui sono associati processori E5-2687W Xeon Sandy Bridge da 3.10 GHz (che da qui verranno chiamati CPU-3G). Per le politiche energetiche di CINECA, i coprocessori Xeon Phi vengono spenti e vengono accesi solo se sono richiesti dal job, mentre le GPU sono impostate in modalità power-save. Quindi quando l'utente lancia il job che richiede acceleratori il prologo dello scheduler setta le GPU in modalità active, mentre nel caso dei Phi esegue il boot della scheda. Le misure energetiche riportate successivamente sono state ottenute usando l' "EURORA Monitoring Framework" [4], questo Framework si occupa di monitorare l'infrastruttura, di raccogliere e di aggregare dati che arrivano dai sensori sparsi per la macchina. L'energy to solution dell'intero nodo invece è ottenuta grazie al software Powerdam che utilizza il framework sopracitato per eseguire il collect dei dati.

3.3 Set-up

I risultati mostrati in questo capitolo sono stati ottenuti usando il PWsfc Kernel che è contenuto nella versione 5.03 della suite di Quantum Espresso. Il software stato compilato utilizzando il compilatore Intel Composer XE-2013 e le librerie Intel MKL 11.01. Per il paradigma message passing è stato utilizzata la libreria IntelMPI 4.1.1. Per compilare la versione GPU è stato utilizzato il CUDA toolkit versione 5.5.22.

3.4 Benchmark

3.4.1 Energy to solution

Per effettuare i benchmark di performance e di efficienza energetica si è usato il test-case *SiO₂* composto da 109 atomi e 582 elettroni avente 800 bande e 1 punto

K. Come pseudo potenziale per la simulazione self-consistente è stato utilizzato un norm-conserving. Il sistema è stato testato su un intero nodo, ovvero due acceleratori e due socket utilizzando 2 task MPI e 8 thread OpenMP per task. I tempi di esecuzione e l'energia consumata per il *SiO2* sono riportati in tabella 3.1. Nella figura 3.2 è possibile osservare i risultati ottenuti dal nodo con i coprocessori Intel Xeon Phi, da questi si può notare che per quanto riguarda i consumi energetici la versione di QE che gira su Xeon Phi è molto più efficiente di quella che viene eseguita sulla sola CPU, indifferentemente dal fatto che essa sia CPU-2G o CPU-3G. Si comparano i tempi di esecuzione tra Xeon Phi e CPU-2G e CPU-3G si nota che la CPU-3G ha il minimo *time to solution*. Questo comportamento può essere spiegato poichè solo uno degli algoritmi più dispendiosi dal punto di vista energetico è stato portato su Intel Xeon Phi, quindi le performance guadagnate con l'uso dello Phi per la matrix multiplication sono compensate dalla riduzione di performance perse per gli altri due algoritmi. In figura 3.3 sono mostrati i risultati ottenuti per il nodo con le nVIDIA K20 GPU. Da questi dati è possibile vedere che la versione di QE per GPU è molto più veloce della versione per CPU ma all'aumentare della velocità si paga il prezzo del consumo energetico, che come si può vedere è molto alto. Confrontando le prestazioni ottenute dalla GPU con altri benchmark ³ si evince che vi è una perdita di performance, infatti lo speedup rispetto la CPU dovrebbe essere maggiore. Questa differenza può essere spiegata considerando la forte dipendenza dalla taglia del dataset dei 3 maggiori algoritmi usati nel kernel PWscf. L'impatto del dataset sull'accelerazione che può dare la GPU è amplificato dal fatto che non tutti i 3 algoritmi sono stati portati su GPU, quindi un dataset può stressare o meno ognuno dei tre algoritmi e mostrare prestazioni maggiori o minori a seconda di quale dispositivo faccia più lavorare. Il dataset scelto per questi benchmark, il *SiO2*, è un buon compromesso per il bilanciamento del tempo speso tra i due device, per mettere meglio in mostra il valore dell' *energy-to-solution* per questo device. Inoltre la taglia in memoria del dataset è consistente con la memoria disponibile nel nodo di EURORA, ossia un 1GB per core, ovvero una piccola richiesta di memoria in input. Mentre per quanto riguarda la richiesta di memoria sul device, questa è notevole, poichè il kernel PWscf sposta su di esso le operazioni di moltiplicazione di matrice su matrici molto grandi. Comparando i tempi di esecuzione tra le CPU-2G e CPU-3G questi sono in accordo con quello che ci si aspettava, poichè corrispondono alla differenza di frequenza di clock tra i due processori. Il piccolo speed-up che si può notare è in accordo alle dimensioni del test-case. La scelta delle dimensioni è stata dettata dal bisogno di avere una piccola richiesta di memoria che andasse bene per la RAM disponibile sui singoli nodi di EURORA (16 Gbytes).

3.4.2 Analisi dei componenti

Si è analizzato inoltre l'energy-to-solution e la potenza per singolo componente del nodo. Grazie all' Eurora Monitoring Framework si è riusciti a misurare la

³<http://on-demand.gputechconf.com/gtc/2012/presentations/S0220-Faster-Material-Science-Modeling-Accelerated-Quantum-Espresso.pdf>

potenza istantanea e si sono graficati i risultati ottenuti in 3.4, questi tre grafici sono relativi all'esecuzione del *SiO2* su specifici nodi: l'esecuzione sul nodo GPU + CPU-3G è stata svolta sul nodo 33, mentre quella solo CPU-3G sul nodo 45 e quella CPU-2G sul nodo 18. Sfortunatamente non si è riusciti ad avere informazioni sulla potenza dello Xeon Phi per colpa di un problema tecnico su EURORA, per riuscire ad ottenere queste informazioni bisognerebbe aggiornare il sistema operativo e cambiare la configurazione con cui si raccolgono i dati. I profili riproducono correttamente il comportamento riportato in figura 3.3 e permettono di estrarre maggiori informazioni sul comportamento di QE quando è in esecuzione. Comparando solo i profili CPU si può osservare che la CPU-3G usa molta più potenza della CPU-2G, questo è dovuto a due fattori, il primo è l'incremento della frequenza del processore, mentre il secondo è dovuto al consumo continuo della GPU anche quando è in modalità power-saving, che corrisponde a $13 \pm 1W$. L'esecuzione con la GPU mostra che durante la simulazione, una parte del calcolo è spostata sul device e quindi vi è un assorbimento minore da parte della CPU rispetto nel caso CPU-3G. Nel caso GPU+CPU-3G si può inoltre osservare che i due device lavorano in parallelo in accordo con la divisione del lavoro impostata dalla libreria phiGEMM. Se combiniamo il profilo di potenza con quello di carico della GPU, vediamo che il device non è utilizzato per la maggior parte del tempo di esecuzione, infatti notiamo pochi picchi. L'uso dei device nell'esecuzione di QE è fortemente dipendente dalla taglia del sistema, infatti come abbiamo già detto per questioni di memoria si è scelto un sistema piccolo, che non enfatizza le performance del device. In tabella 3.2 si sono riportati i valori di energy-to-solution per singolo componente ottenuti per integrazione dei profili di potenza. I valori di energy-to-solution calcolati possono essere differenti dai precedenti ottenuti con Powerdam, poichè il software integra il costo energetico dell'infrastruttura, ossia l'energia spesa per il cooling o per la rete ecc. Naturalmente i dati di energy-to-solution mostrano gli stessi risultati dei precedenti, ma da questi possiamo dedurre informazioni sui pesi dei diversi componenti nell'esecuzione del software. In tutti i casi si vede che il contributo minore è dato dalla memoria, infatti la RAM ha un energy-to-solution di un ordine di grandezza in meno rispetto agli altri device. Nel caso GPU+CPU-3G si può dire che la CPU spende maggior energia che la GPU ma questo è molto dipendente dalla taglia del sistema.

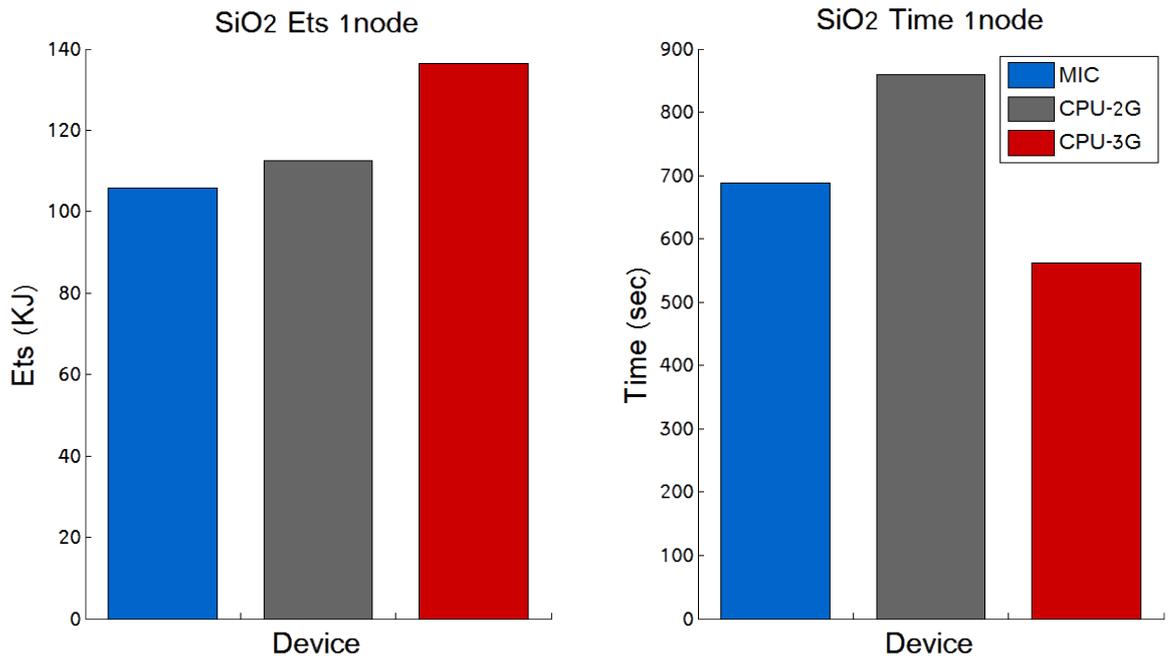


Figura 3.2: Time-to-solution (destra) and Energy-to-solution (sinistra) comparazione tra Xeon Phi e CPU ottenuti eseguendo QE su singolo nodo

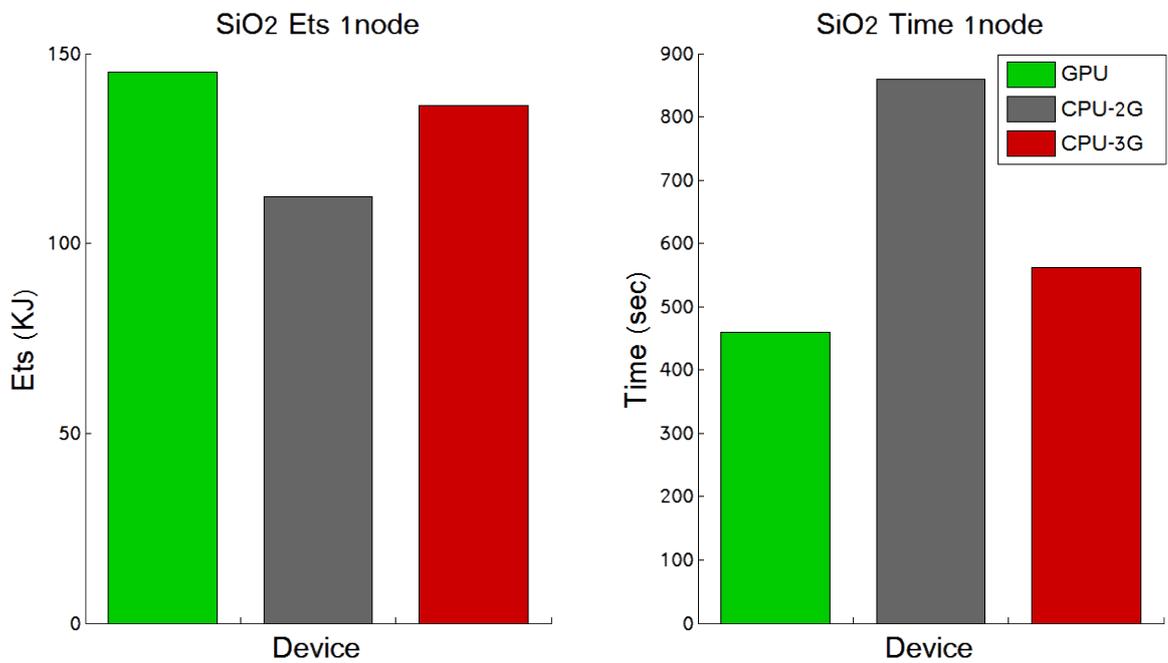


Figura 3.3: Time-to-solution (destra) e Energy-to-solution (sinistra) comparazione tra GPU e CPU ottenuti eseguendo QE su singolo nodo

	GPU + CPU-3G	CPU-3G	CPU-2G
Node	33	45	18
time (sec)	459 ± 1	540 ± 1	840 ± 1
E_{gpu1} (KJ)	23 ± 5	7.3 ± 0.1	-
E_{gpu2} (KJ)	23 ± 7	7.6 ± 0.1	-
E_{ram1} (KJ)	2.2 ± 0.1	1.5 ± 0.1	3.8 ± 0.2
E_{ram2} (KJ)	3.7 ± 0.3	2.6 ± 0.4	5.8 ± 0.7
E_{cpu1} (KJ)	38 ± 3	59 ± 14	41 ± 10
E_{cpu2} (KJ)	38 ± 7	56 ± 17	41 ± 11
E_{node} (KJ)	128 ± 23	134 ± 32	82 ± 22

Tabella 3.2: Consumi energetici per singolo componente ottenuti per 3 specifici job con il test case *SiO2*.I risultati sono riportati per CPU-2G, CPU-3G e GPU+CPU-3G.

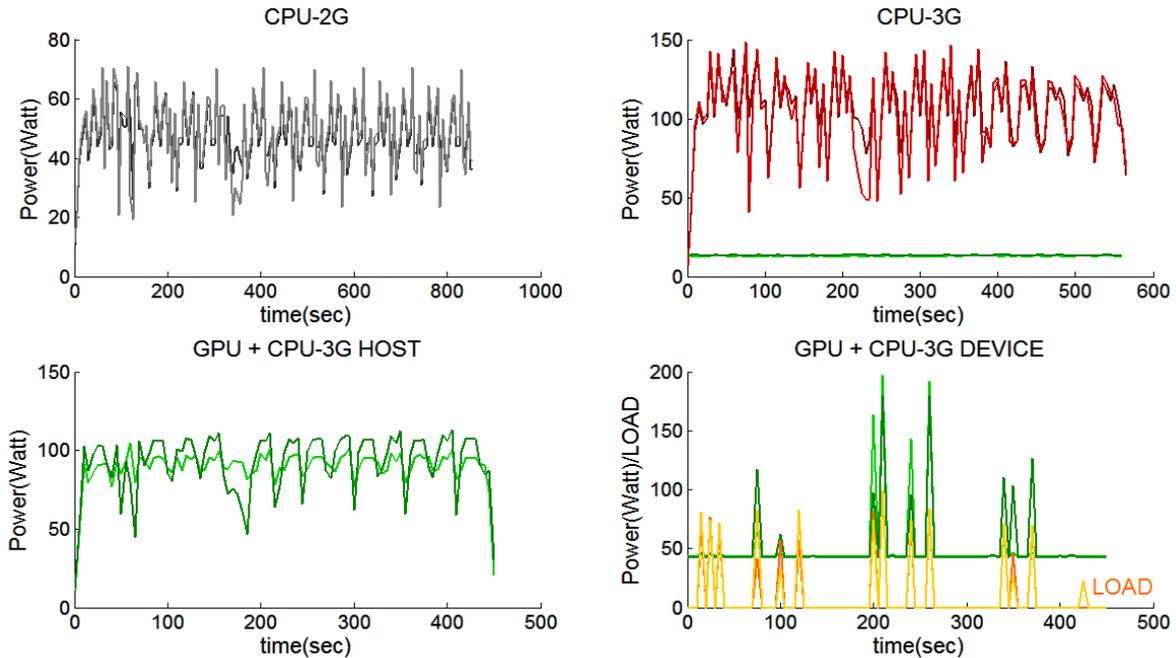


Figura 3.4: Comparazione dei power profile di ogni elemento sul nodo. In ogni grafico sono riportati i due device con linee di differente colore. Nel grafico in alto a destra, la linea sul fondo rappresenta la potenza consumata dalla GPU in modalit  power-saving. Nel grafico in basso a destra, la linea arancione rappresenta il carico di lavoro della GPU. In tutti i grafici la potenza consumata dalla ram   inclusa nel profilo della CPU.

Capitolo 4

Studio e implementazione di strategie per l'aumento della località dei dati

Come già detto nella sezione 1.2.4 una delle maggiori problematiche sui sistemi exascale sarà il costo energetico dello spostamento dei dati. Per tale motivo in questa sezione si presenterà una possibile soluzione per aumentare la località della distribuzione dei dati di Quantum ESPRESSO(QE). Le osservabili fisiche con cui QE lavora sono la densità di carica, le funzioni d'onda ed i potenziali (capitolo 2). Per semplificare i conti, alcune di queste vengono trattate in spazio reale ed altre in quello reciproco a seconda di dove queste siano diagonali. L'operazione con cui si passa da uno spazio all'altro è la trasformata discreta Fourier. Per la particolare implementazione di questa operazione in QE viene utilizzato un algoritmo 3D FFT ad hoc con lo scopo di minimizzare il numero di trasformate da effettuare. Questo implica, rispetto alle altre operazioni del codice stesso, uno scambio di dati con tutti i processi coinvolti nel calcolo e quindi, secondo quanto riportato sopra, dispendioso energeticamente se considerato nella prospettiva di sistemi exascale. Avendo come obiettivo quello di migliorare la resa energetica di calcoli ab-initio sulle macchine di prossima generazione, è stato studiato ed implementato un algoritmo alternativo per la distribuzione dei dati che aumenti la località di questi ultimi. Fino ad oggi (sistemi Petascale) questo problema non si poneva, mentre era essenziale bilanciare il carico di lavoro sui processori a discapito della minimizzazione delle comunicazioni. Nelle macchine Exascale vi sarà un numero tale di processori grazie al quale il costo principale non sarà più l'elaborazione dei dati, ma il loro spostamento, ovvero la comunicazione tra un processore e l'altro. L'attuale distribuzione dei dati di QE [9], essendo pensata per le attuali macchine, si basa sul bilanciamento del carico di lavoro, mentre la nuova distribuzione, che viene presentata in questo elaborato, si basa sulla minimizzazione della comunicazione e lo sbilanciamento del carico di lavoro.

4.1 Distribuzione G e FFT attuale

Una conseguenza del fatto che le funzioni d'onda (autostati del sistema) sono rappresentate usando come base le onde piane aventi un'energia cinetica minore di un cut-off pre assegnato, è quella di produrre una distribuzione di dati sferica nello spazio reciproco. Essendo i potenziali funzione della densità di carica, definita come una funzione quadratica delle funzioni d'onda, risultano rappresentati sulla base delle onde piane aventi un cut-off 4 volte superiore a quello delle funzioni d'onda. Nello spazio diretto le funzioni d'onda ed i potenziali sono rappresentati utilizzando come base i vettori reali quindi un reticolo 3D cartesiano. Riguardo alla distribuzione dati sui processori disponibili, si può affermare che:

- nello spazio reale, come si può osservare in 4.1 (1), la distribuzione dei dati sui processori è effettuata segmentando la matrice in fette perpendicolari all'asse Z.
- nello spazio reciproco, si veda figura 4.1 (2), sono rappresentati i dati come appaiono nel reticolo reciproco, ovvero come una semisfera.

Le linee tratteggiate indicano il limite dei vettori delle funzioni d'onda (E_c) e dei potenziali ($4E_c$). Come si può vedere dallo schema nel reticolo reciproco, la parallelizzazione sui processori non tiene conto di relazioni geometriche tra le diverse colonne dati inscritte nella sfera(stick), ma bilancia solamente il carico di lavoro. Di conseguenza per passare da uno spazio all'altro, la FFT dovrà eseguire un rimescolamento degli stick tra i processori, questo lo si può notare nella figura 4.2. La FFT che si deve eseguire è 3D, quindi poiché inizialmente vi sono gli stick già distribuiti lungo Z, si esegue una FFT 1D lungo tale direzione, successivamente si esegue una chiamata All2All per spostare gli stick da un processore all'altro e una trasformata di Fourier 2D sul piano XY.

I dettagli tecnici sull'implementazione della attuale distribuzione si possono trovare in appendice A.

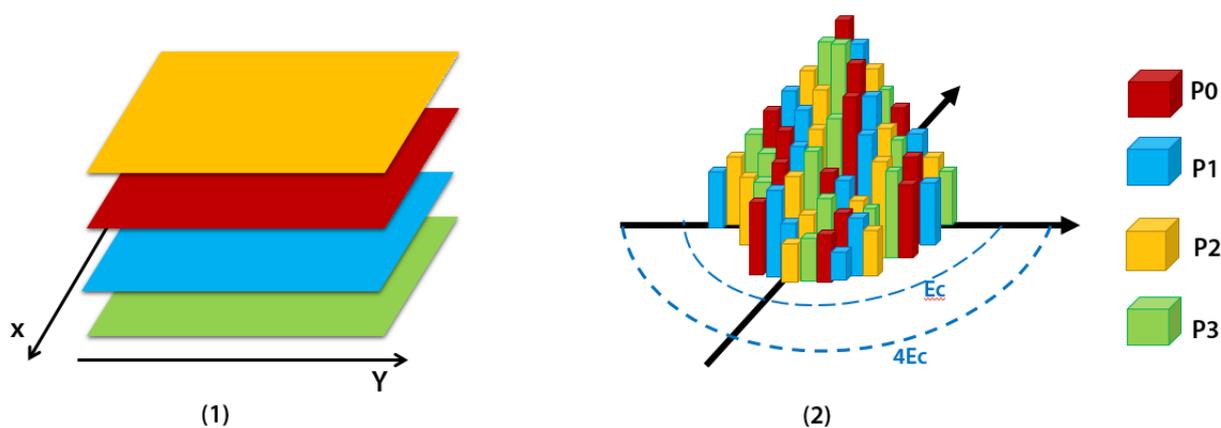


Figura 4.1: Schema della parallelizzazione dei dati nello spazio reale(1) e reciproco(2) per la distribuzione attuale implementata in QE (4 processori)

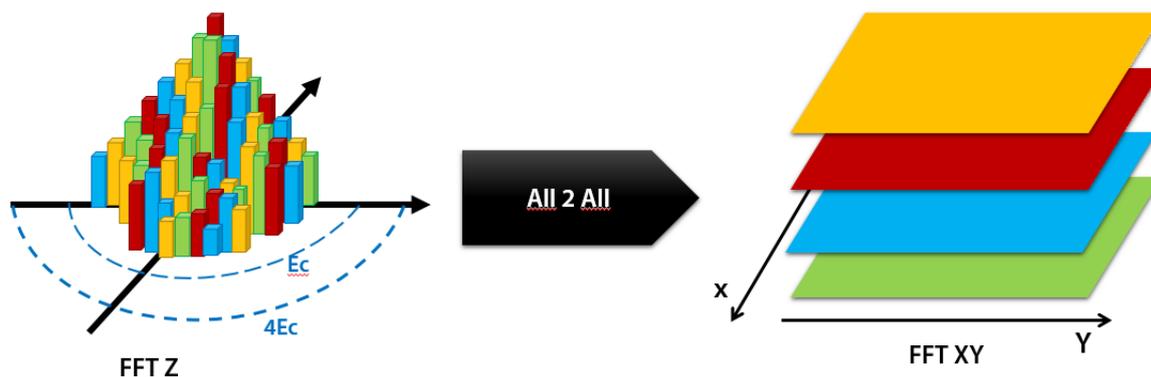


Figura 4.2: Schema della parallelizzazione sui dati della attuale implementazione dell'algoritmo FFT attuale (4 processori)

4.2 Nuova distribuzione dati gBlockDistro

Per migliorare gli aspetti della località dei dati si è studiata ed implementata una nuova distribuzione dati, gBlockDistro, che prende l'idea della distribuzione dati dalla libreria Scalapack che performa le operazioni di algebra lineare su architetture massivamente parallele. Questa distribuzione non bilancia più il carico di lavoro, ma minimizza la comunicazione per il calcolo della FFT mantenendo sulle righe e sulle colonne della griglia gli stessi processori. gBlockDistro è una distribuzione a blocchi: in funzione del numero di processori vengono definiti dei *coreblock*, questi hanno una dimensione che è definita in funzione al numero di processori. In ogni cella di questi blocchi viene memorizzato il numero di processore, partendo dal basso a sinistra, fino ad arrivare in alto a destra. In funzione del numero di processori viene definito come riempire la matrice che mappa il numero dei processori sugli indici di Miller, detta *procmap*. questa matrice è riempita in funzione del raggio massimo della griglia della funzione d'onda (R) che definisce di quanto deve essere esteso il coreblock, ogni cella del core block avrà una dimensione $(R/\text{numero core per riga}) \times (R/\text{numero core per colonna})$. Verranno inseriti tanti *coreblock* quanti saranno necessari a completare il riempimento della *procmap*. A differenza delle Scalapack, se avanzano degli elementi al bordo questi verranno assegnati allo stesso processore del blocco più vicino. In questo modo si distribuiscono più omogeneamente i vettori G sui processori poiché quelli a cui viene assegnato il nucleo centrale della matrice avranno in generale un carico maggiore avendo, in più rispetto a quelli esterni, l'assegnazione degli stick con numero di vettori G maggiore (colonne più alte). La distribuzione gBlockDistro viene inserita nel codice senza una modifica massiccia, cioè semplicemente vengono aggiunte due chiamate alla distribuzione dentro la subroutine *sticks_dist* (vedi appendice A) e viene inserita una condizione di selezione tra la vecchia e la nuova distribuzione. La chiamata a gBlockDistro ritorna una mappa dei processori indipendente dai vettori G . La condizione inserita evita che questa mappa sia sovrascritta da quella della distribuzione precedente. Il modulo Fortran gBlockDistro è costituito da sei subroutine in totale, per la creazione della mappa.

1. **find_max**: si occupa di individuare il raggio massimo della sfera delle funzioni d'onda, ossia identifica l'elemento non nullo con indice di Miller più alto.
2. **maps_sca** date in input tutte le informazioni sulle dimensioni delle mappe dei vettori G , gestisce il flusso che porta alla generazione della nuova distribuzione.
3. **set_block** dato il numero di processori totali, calcola la dimensione del *coreblock* definendo il numero di righe e colonne interessate. Il numero di righe è dato dal più grande modulo tra il numero di processori e un valore compreso tra 1 e la radice quadrata del numero di processori incrementata di uno, mentre il numero di colonne è dato dal rapporto tra il numero di core ed il numero di righe.
4. **get_info_block**: calcola il numero di elementi di *procmap* che corrispondono ad una singola cella di *coreblock*.
5. **set_matrix_processor**: riempie il *coreblock* con l'indice dei processori.
6. **distro_matrix_processor**: riempie *procmap* estendendo *coreblock* e ripetendolo.

4.3 Quantum Espresso Roadmap FFT

La nuova distribuzione dati sui processori, implementata con gBlockDistro, può essere schematizzata nello spazio reale e reciproco come in figura 4.3.

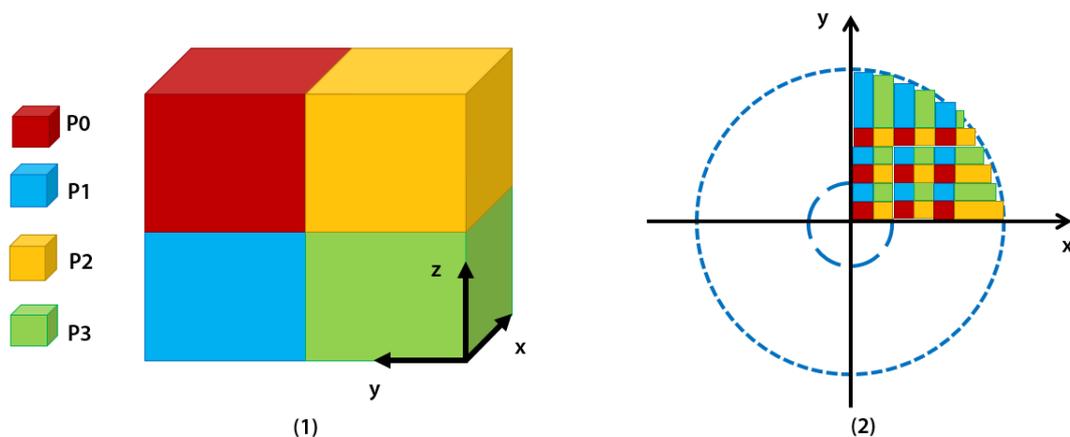


Figura 4.3: Schema della parallelizzazione dei dati nello spazio reale(1) e reciproco(2) per la nuova distribuzione che sarà implementata in QE (4 processori)

Per rendere effettiva la riduzione delle comunicazioni occorrono ulteriori sviluppi nella parte del codice che implementa la FFT. In particolare per ridurre la comunicazione sfruttando la geometria di questa nuova distribuzione a blocchi si potrebbe

implementare una decomposizione 2D del dominio 3D, ovvero quella che in letteratura è detta *3D FFT con decomposizione 2D* [32]. Applicando questa tecnica le FFT passerebbero da 2 a 3, poiché si applicherebbero 3 FFT 1D; mentre dal punto di vista della comunicazione, questa avverrebbe solamente tra sottogruppi di processori che mappano o le righe o le colonne del set di dati totale. Quindi il numero di comunicazioni tra i processori diminuirebbe. Lo schema della FFT che si pensa di implementare è riportato in figura 4.4:

1. I dati sono già distribuiti lungo la direzione z , quindi si esegue la FFT1D lungo questa direzione.
2. Si scambiano i dati lungo x e z in modo tale che ogni processore abbia tutti i dati lungo y su cui eseguire la FFT. Per distribuire x si eseguirà una All2All tra i processori che sono sulla stessa riga, mentre per distribuire y si eseguirà una All2All tra i processori che sono sulla stessa colonna.
3. Si esegue la FFT lungo la direzione y .
4. L'ultima direzione su cui eseguire FFT1D è lungo x , quindi si scambiano i dati lungo y e x in modo tale che ogni processore abbia tutti i dati lungo x su cui eseguire la FFT. Per distribuire x si eseguirà una All2All tra i processori che sono sulla stessa riga, mentre per distribuire y si eseguirà una All2All tra i processori che sono sulla stessa colonna.
5. Si esegue la FFT1D lungo x .

Se si osserva lo schema, ci si può accorgere che i processori che “si parlano” sono sempre gli stessi e non tutti come nel caso precedente. Anche in altri casi questo tipo di approccio ha portato a un miglioramento delle performance [14].

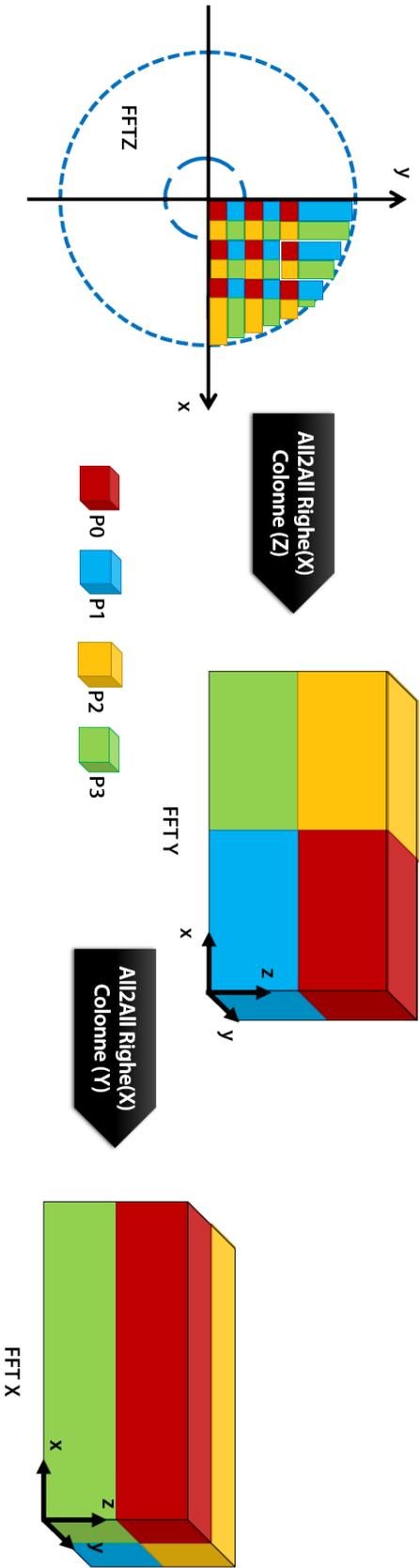


Figura 4.4: Schema della parallelizzazione sui dati della attuale implementazione dell'algoritmo FFT che sarà implementato (4 processori)

Capitolo 5

Applicazioni al calcolo di proprietà strutturali, elettroniche e di trasporto di materiali di interesse tecnologico

In questo capitolo presenteremo un esempio di applicazione degli strumenti di calcolo descritti nei capitoli precedenti ad un sistema scientificamente rilevante nell'ambito della scienza dei materiali. Una delle sfide in vista dell'exascale è quella di avere, quando sarà costruita la prima macchina, codici già pronti e funzionanti e soprattutto con l'idea di quali sono i parametri ottimali per la loro esecuzione; per fare ciò attualmente si cercano di eseguire simulazioni portando al limite le performance delle macchine attuali per testare la robustezza e la scalabilità del codice e per trovare i parametri migliori per il completamento della simulazione. Infatti la simulazione che si implementa dal punto di vista tecnologico è una delle prime di tali dimensioni e servirà per testare il flusso di lavoro riportato nel paragrafo 2.3 che accoppia la suite Quantum ESPRESSO e WanT. La simulazione finale di un dispositivo ideale comprendente gli elettrodi e il cristallo ha richiesto su Fermi(BGQ) più di 500 ore solari su 16384 processori ossia circa 8 milioni di ore calcolo. Per quanto riguarda il sistema fisico, ci occuperemo nello specifico del N,N'-1H,1H-perfluorobutyl dicyanoperylenecarboxydiimide detto $PDI - FCN_2$ [8], candidato ideale il calcolo su macchine exascale. Il materiale in esame è di grande interesse sia scientifico che tecnologico, poiché è stato proposto quale materiale ideale per la fabbricazione di transistor organici. Per riuscire a comprendere al meglio le proprietà del sistema, prima studieremo le proprietà di trasporto del cristallo 3D di $PDI - FCN_2$, poi quelle di un sistema a due terminali oro/molecola/oro quale prototipo di dispositivo di singola molecola. All'interno di questo ambito di ricerca, il contributo originale della mia tesi ha riguardato l'implementazione computazionale dei due sistemi con il $PDI - FCN_2$ (cristallo 3D e sistema a due terminali), il test del flusso di lavoro 2.3 su un sistema di così grandi dimensioni, e il calcolo delle proprietà di trasporto dei sistemi implementati.

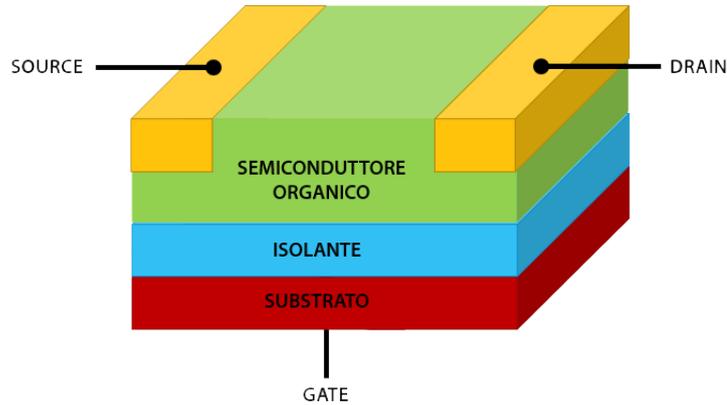


Figura 5.1: Schema ideale di un transistor OFET

5.1 Stato dell'arte

In vista delle potenziali applicazioni per la realizzazione di elettronica su larga scala, flessibile a basso costo, i semiconduttori organici hanno ricevuto una crescente attenzione negli ultimi decenni[26]. Il rapido sviluppo del campo dell'elettronica organica è dovuto all'ampia gamma di applicazioni nei quali questi dispositivi possono essere usati[20]: dalla e-paper, passando per biosensori e celle fotovoltaiche, fino ad arrivare a display e computer molecolari. I semiconduttori organici riescono, infatti, ad avere nel contempo le proprietà elettriche dei semiconduttori inorganici e quelle meccaniche dei materiali plastici come la flessibilità. Tuttavia rispetto ai semiconduttori inorganici le molecole che costituiscono materiali organici hanno la possibilità di essere ingegnerizzate al fine di ottimizzare le proprietà ottiche e di trasporto del materiale. I semiconduttori organici sono inoltre di facile scalabilità e purificazione. Questi vantaggi sono importanti non solo per le ricadute industriali ma anche per lo studio delle relazioni struttura-proprietà a livello microscopico. I composti organici possono ricreare tutti i corrispondenti dispositivi a base di silicio, dai diodi ai vari tipi di transistor. Di grande interesse sono i corrispettivi organici dei transistor ad effetto campo (FET), ovvero gli OFET (organic field effect transistor) [3]. La struttura di un OFET è basata sulla presenza di un canale semiconduttore organico che separa due elettrodi (detti source e drain) e un terzo elettrodo, separato da uno strato isolante, detto gate; vedi figura 5.1. Durante il funzionamento del transistor si ha l'applicazione del voltaggio V_{GS} tra gli elettrodi di gate e source che permette l'accumulo di cariche all'interfaccia tra il semiconduttore-dielettrico sfruttando la capacità della struttura composta dal gate, dal dielettrico e dal semiconduttore organico. In questo modo la corrente I_{DS} che fluisce tra l'elettrodo di drain e di source è funzione del rispettivo voltaggio che può essere modulato variando V_{GS} .

Un parametro molto importante per questo tipo di dispositivi è la mobilità ($\mu_{FET} [\frac{cm^2}{V \cdot s}]$) dei portatori di carica; vi possono essere due tipi di conduzione dette di tipo P, ossia conduzione per buche, e tipo N, ossia conduzione per elettroni. I composti che vengono utilizzati maggiormente per gli OFET sono a singolo cristal-

lo poiché hanno una migliore efficienza nel trasporto. Per quanto riguarda gli OFET di tipo P, questi sono basati su materiali quali tetracene ($\mu_{FET} = 2.4 \frac{cm^2}{V \cdot s}$) e rubrene ($\mu_{FET} = 20 \frac{cm^2}{V \cdot s}$) [24] che hanno mobilità fra le più alte, oppure su antracene, pentacene, TIPS-pentacene, DPh-BDSe, e BPT2 che hanno mobilità comprese tra $0.02 - 2.2 \frac{cm^2}{V \cdot s}$ [24]. Per quanto riguarda i transistor a canale N, questi sono più carenti in termini di performance e più rari in termini di numeri di composti. Il composto a canale N, già in produzione, che fornisce le maggiori performance è il Rame-FCuPc con $\mu_{FET} = 0.2 \frac{cm^2}{V \cdot s}$ e il TCNQ con $\mu_{FET} = 0.2 \frac{cm^2}{V \cdot s}$. A dispetto dei tanti miglioramenti raggiunti negli ultimi anni, lo sviluppo di semiconduttori organici di tipo N risente di forti ritardi rispetto ai semiconduttori di tipo P, sia in termini di mobilità che di stabilità ambientale. Lo sviluppo di materiali di tipo N è fondamentale per la realizzazione di transistor.

Un materiale di tipo N ideale dovrebbe avere un'alta mobilità, bassa soglia di voltaggio, buona stabilità in aria rispetto ai processi ossidativi e facile processabilità. Per ottenere questo tipo di materiali un prerequisito importante è un'alta affinità elettronica che permetta una facile iniezione di carica da parte degli elettrodi [21]. Un'alta affinità elettronica è associata alla posizione in energia del LUMO (lowest occupied molecular orbital) della molecola costituente, che andrà poi a definire il fondo di banda di conduzione del corrispondente semiconduttore organico. Una classe promettente da questo punto di vista è quella dei derivati del perilene [33], che mostrano un'eccellente performance nel trasporto elettrico e sono molto stabili nell'ambiente. Particolarmente rilevanti sono il PID18-CN2 ($\mu_{FET} = 0.1 - 0.6 \frac{cm^2}{V \cdot s}$) e il *PDI* – *FCN*₂ [24]. A fianco della realizzazione di dispositivi standard (tipo OFET), basati su cristalli molecolari e/o film sottili di materiale molecolare organico, i derivati del perilene sono stati recentemente utilizzati anche nell'ambito dell'elettronica molecolare, per la realizzazione di break junction. A partire dai lavori pionieristici di Aviram e Ratner degli anni 70, l'elettronica molecolare rappresenta l'estremo limite molecolare per la miniaturizzazione dei dispositivi elettronici. La realizzazione di dispositivi di singola molecola ha in prospettiva importanti vantaggi in termini di costi dei materiali, efficienza e potenza dissipate. Inoltre è possibile sfruttare proprietà di riconoscimento o auto-aggregazione per la realizzazione di tecnologie completamente bottom-up. Dal punto di vista puramente teorico, la descrizione di tali dispositivi molecolari richiede una profonda conoscenza dei meccanismi di trasporto di carica e scattering dissipativo e interazione con elettrodi, rispetto agli usuali dispositivi descritti in precedenza.

5.2 Struttura e proprietà *PDI* – *FCN*₂ cristallino

La molecola di *PDI* – *FCN*₂ ha una struttura planare composta da anelli aromatici di carbonio, tipica dei perileni, funzionalizzata agli estremi da due catene di perfluorobutile e da due gruppi ciano in posizione trans al centro vedi figura 5.2.

Attualmente i derivati del Perylene Carboxylic Diimide sono una delle più promettenti classi di semiconduttori organici per la costruzione di OFET. La caratteristica principale di questi composti è il basso valore dell'energia del LUMO che

Parameter	Value
a	5.135 Å
b	7.385 Å
c	19.598 Å
α	92.35°
β	82.22°
γ	93.25°
V	734.75 Å ³
ρ	1.817 gr/cm ³

Tabella 5.1: Parametri del Cristallo molecolare $PDI - FCN_2$

corrisponde ad un'alta affinità elettronica. Tra tutti i composti il N,N'-1H,1H-perfluorobutyl dicyanoperylenecarboxydiimide detto $PDI - FCN_2$ è particolarmente interessante poiché è facile da depositare e soprattutto ha un livello di affinità elettronica comparabile con l'energia di Fermi dell'oro e questo permette una conduzione per elettroni e non per buca come la maggior parte dei composti organici. Il cristallo molecolare ha una cella triclina con parametri di cella riportati in tabella 5.1; la struttura cristallina è caratterizzata da una "sovrapposizione traslata faccia a faccia" lungo la direzione a con uno spazio tra i piani di 3.2Å, la lunghezza della molecola è 23.5Å e l'angolo diedro tra la metà superiore e inferiore della molecola è di 3°. Dall'impaccamento di queste molecole si nota una forte anisotropia nella struttura cristallina che suggerisce un ottimo accoppiamento elettronico tra layer adiacenti lungo l'asse cristallografico a, dove l'overlap degli orbitali π è condizione necessaria per un buon trasporto in un OFET di tipo N. Lo studio delle proprietà strutturali ed elettroniche del cristallo $PDI - FCN_2$ sono state precedentemente studiate con metodi ad-initio simili a quelli descritti nel capitolo 2. Poiché è necessario caratterizzare lo stato fondamentale per poi procedere con il successivo calcolo del trasporto, qui abbiamo riottenuto le proprietà elettroniche fondamentali per lo stesso sistema. In figura 5.4 sono riportate le bande e la zona di Brillouin per il cristallo molecolare estratte dall'articolo [8].

5.3 Conto Bulk

Per prima cosa si è deciso di andare a studiare il volume del solido $PDI - FCN_2$. La cella simulata è una triclina con i parametri di cella riportati in tabella 5.1 contenente 66 atomi e 288 elettroni. Per la simulazione sono stati usati potenziali ultra soffici e PBE come funzionale di scambio correlazione, le funzioni d'onda di singola particella e la densità di carica sono state sviluppate in onde piane fino a un cutoff di 28 Ry e 280 Ry, l'integrazione sulla zona di Brillouin è stata fatta usando una griglia di $4 \times 4 \times 2$ sulla zona completa, che corrisponde sfruttando le simmetrie del cristallo a 16 punti K nella zona ridotta. Per quanto riguarda le posizioni atomiche iniziali sono state utilizzate quelle estratte da un precedente lavoro [8] in cui sono state ottenute da calcoli KS-DFT di ottimizzazione strut-

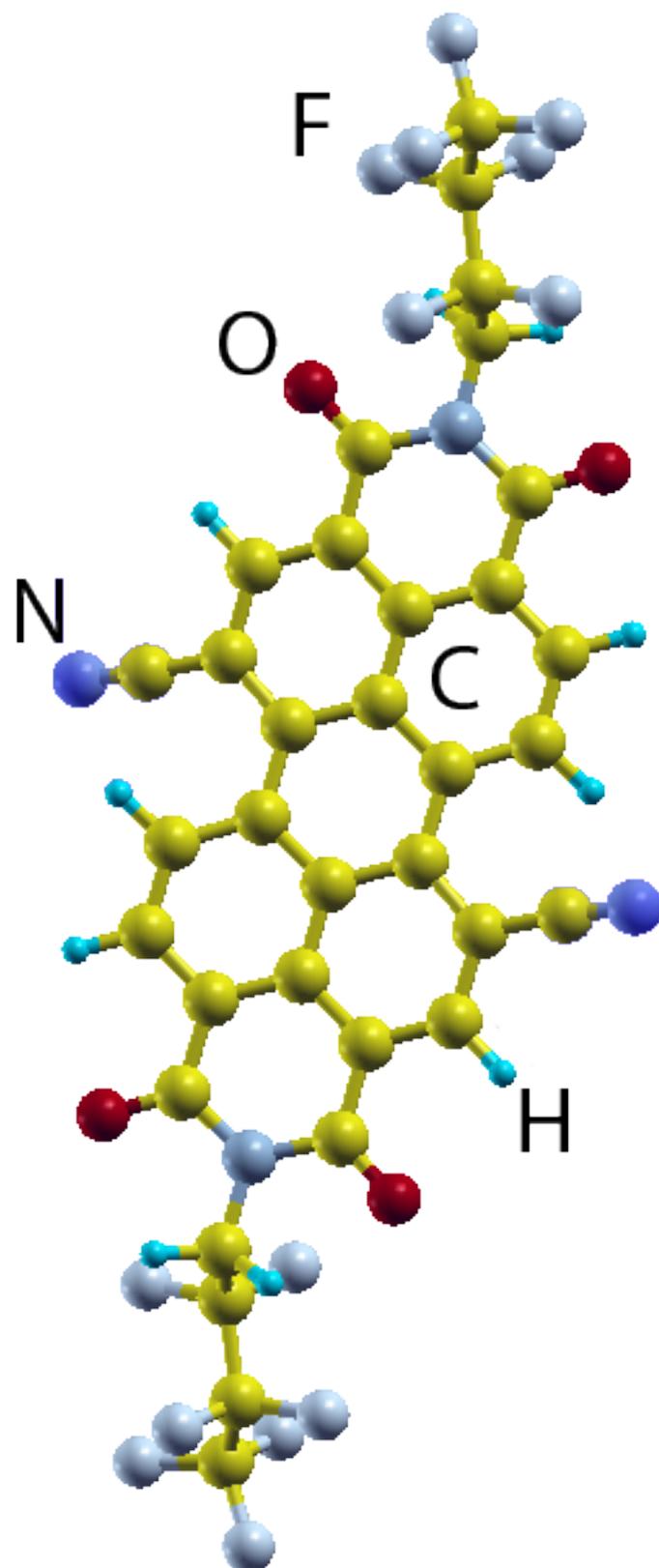


Figura 5.2: *StrutturamolecolaredelPDI – FCN₂*

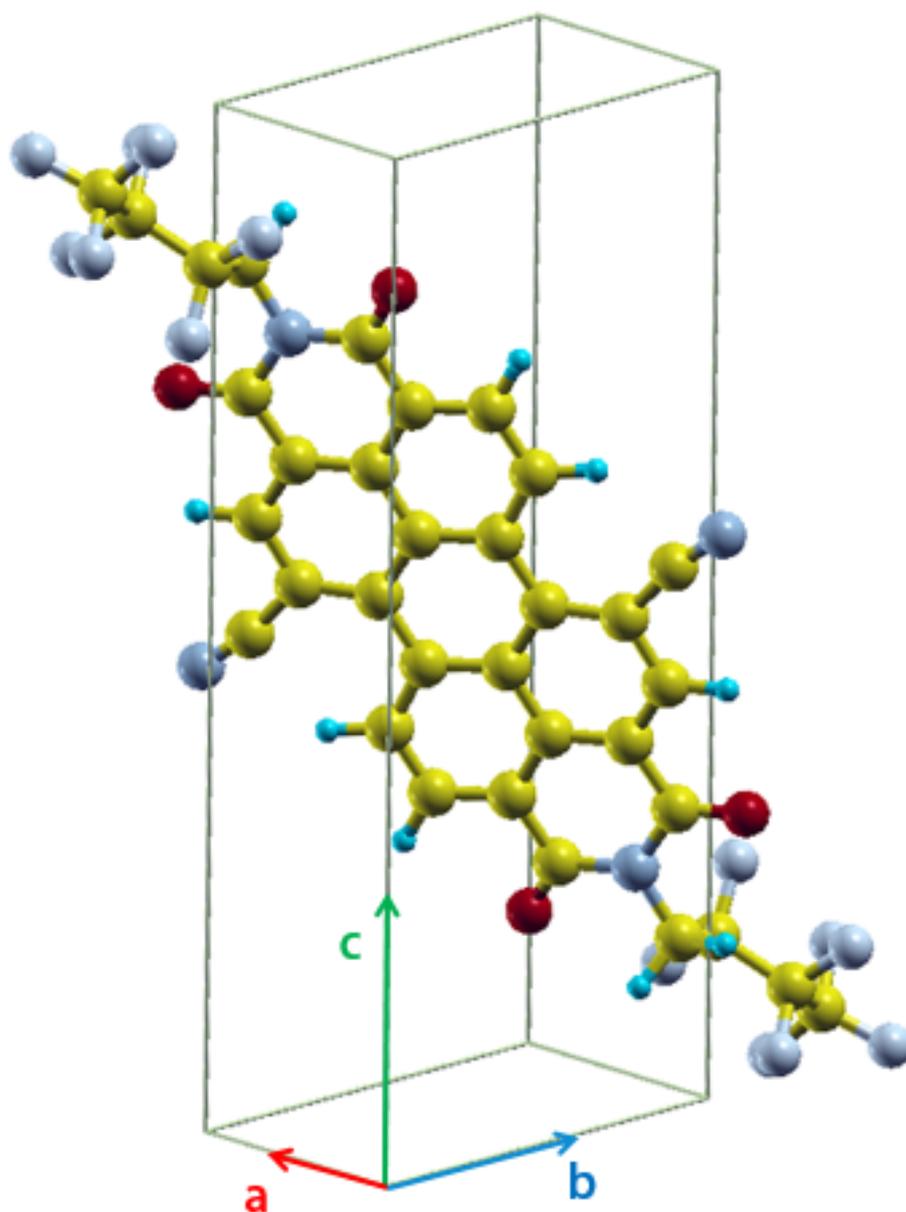


Figura 5.3: Cella unitaria del cristallo $PDI - FCN_2$

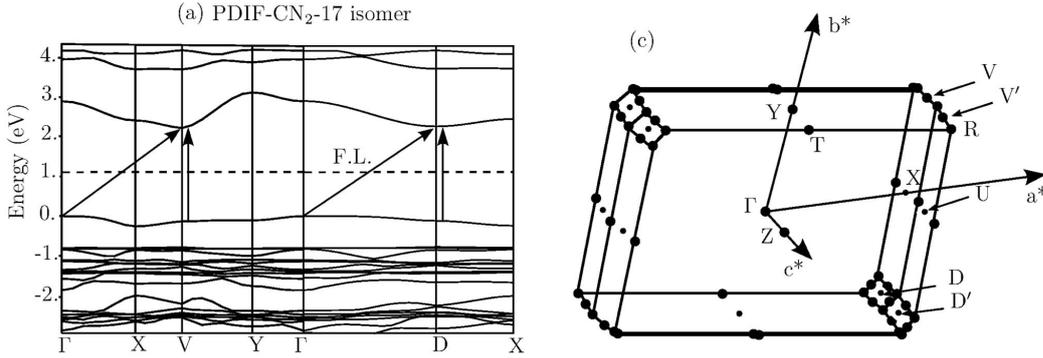


Figura 5.4: Struttura a bande $PDI - FCN_2$ estratta da [8]

turale. Dal calcolo nscf è stato possibile ottenere le bande del cristallo per la direzione $\Gamma \rightarrow X$ che sono riportate in figura 5.5 e si può vedere come queste sono in accordo con le bande ottenute da calcoli KS-DFT riportati nell'articolo [8] e in figura 5.4. In figura 5.6 vengono graficate le curve di trasmittanza (a meno della ben nota sottostima del valore del gap) lungo i tre assi principali del cristallo: lo zero dell'energia corrisponde al valore dell'energia di Fermi; nell'inserto del grafico si è solamente ingrandito il grafico intorno all'energia di Fermi. Confrontando le 3 direzioni si può notare che queste non sono equivalenti dal punto di vista del trasporto, avendo posto l'energia di Fermi sullo 0, è possibile notare che lungo l'asse C la funzione di trasmissione è sostanzialmente nulla per energie vicine a quelle del livello di Fermi. Questo è indice di una bassissima sovrapposizione spaziale tra gli orbitali elettronici, che mantengono un carattere puramente molecolare e localizzato sul singolo $PDI - FCN_2$, inibendone il trasporto di banda. Al contrario, lungo le altre due direzioni (a,b), l'accoppiamento molecola-molecola attraverso gli orbitali π dell'unità perilenica contribuisce a valori non nulli della trasmittanza e quindi a un migliore trasporto elettronico.

Nelle figure 5.7, 5.8, 5.9 sono riportate le curve $I - V$ lungo i tre assi del cristallo. In ogni figura vi è una colonna per la trasmittanza e una per la corrente; in ognuna di queste sono riportati tre grafici a diversi shift del livello di Fermi, che in regime lineare corrisponde all'applicazione di un potenziale esterno di gate. Essendo in regime lineare, abbiamo considerato qui potenziale solo fino a un massimo di $1.0eV$ ad energie superiori si potrebbero avere alterazioni della struttura elettronica. Quindi per ottenere questi grafici, si è applicato al grafico uno shift di $0.5eV$ poiché è l'ordine di grandezza del gap. Nel caso dello shift positivo si ha che il livello di Fermi cade all'interno nella banda di conduzione ma questo è dovuto solamente all'utilizzo dello schema scambio correlazione adottato (PBE) che porta ad una sottostima del gap. Tolle possibili ulteriori correzioni rigide dello shift, si può dire che la curva $I - V$ ha un andamento per le direzioni a,b:

- $V_g = +0.5$ il solido è in conduzione, l'andamento è lineare quindi siamo in regime ohmico.
- $V_g = +0.0$: il solido non conduce.
- $V_g = -0.5$: All'interno del regime lineare abbiamo conduzione.

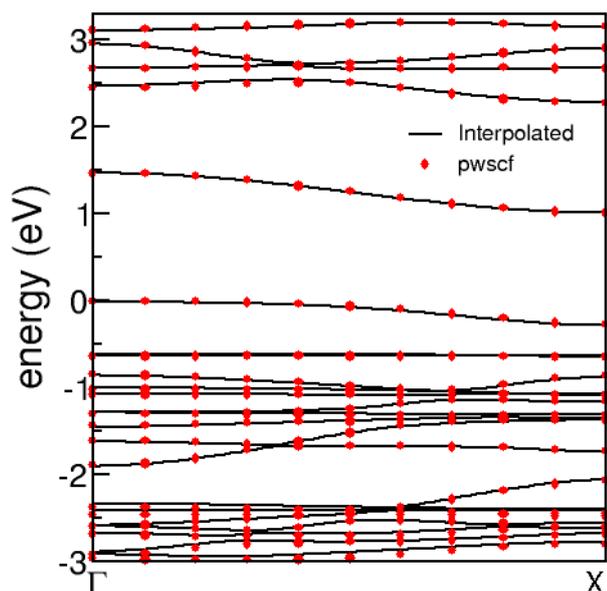


Figura 5.5: Bande ottenute dal conto PWscf per $PDI - FCN_2$

Nella direzione a, si nota che la conduzione è maggiore e questo è dovuto alla maggior presenza di stati disponibili di trasmissione; mentre per quanto riguarda la direzione c non c'è trasporto di carica.

5.4 Dispositivo a 2 terminali

Successivamente al conto di bulk per investigare le proprietà di conducibilità di singola molecola, si è simulato un dispositivo ideale con 2 elettrodi in oro e con una molecola al centro.

La cella simulata è una esagonale con i parametri di cella $a = 17.59\text{\AA}$ e $c = 31.38\text{\AA}$ contenente un elettrodo d'oro formato da 9 piani atomici di dimensioni $14.7\text{\AA} \times 14.7\text{\AA}$, ognuno composto da 6×6 atomi e una molecola di $PDI - FCN_2$ affacciata sulla superficie (111) dell'elettrodo, in totale 390 atomi e 3852 elettroni, vedi figura 5.10.

Per la simulazione sono stati usati potenziali ultra soffici con PBE come funzionale di scambio correlazione, le funzioni d'onda di singola particella e la densità di carica sono state sviluppate in onde piane fino a un cutoff di 25 Ry e 280 Ry, l'integrazione sulla zona di Brillouin è stata fatta usando una griglia di $2 \times 2 \times 4$ sulla zona completa, che corrisponde sfruttando le simmetrie del cristallo a 16 punti K nella zona ridotta. La configurazione iniziale della molecola rispetto alle superfici d'oro è stata scelta sulla base di risultati sperimentali preliminari che suggeriscono la formazione del legame N-Au quale favorito per l'adsorbimento della molecola. L'intera struttura è stata rilassata con il pacchetto Pw di QE, e questo ha richiesto più di 7 milioni di ore di calcolo su Fermi su 16384 core, questo può dare un'idea delle dimensioni computazionali del problema e che senza tecniche

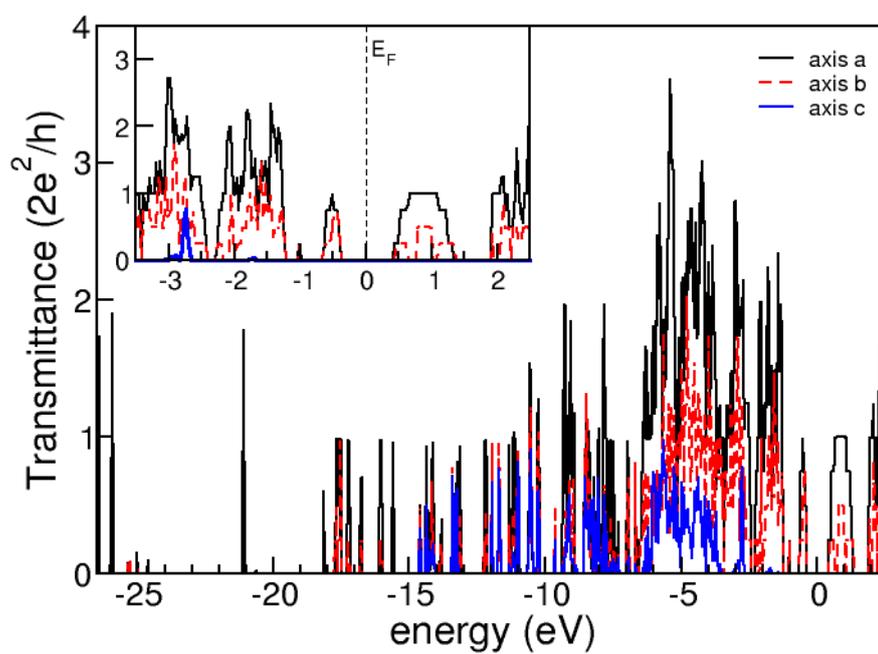


Figura 5.6: Curve di trasmittanza per il cristallo $PDI - FCN_2$, la linea nera si riferisce alla direzione a, la linea rossa si riferisce alla direzione b, la linea nera si riferisce alla direzione c

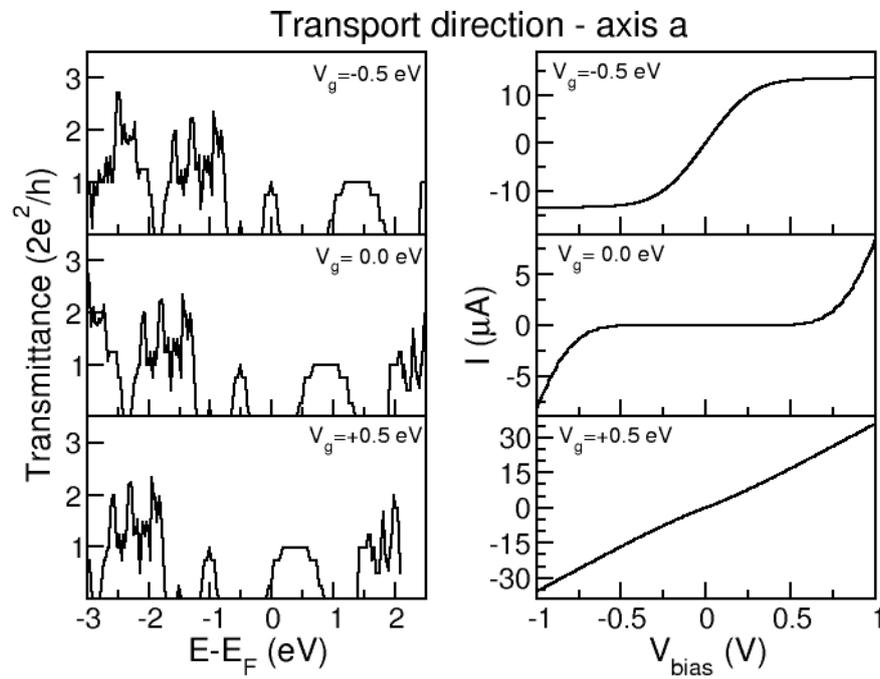


Figura 5.7: Curve di trasmittanza e I-V per direzione a

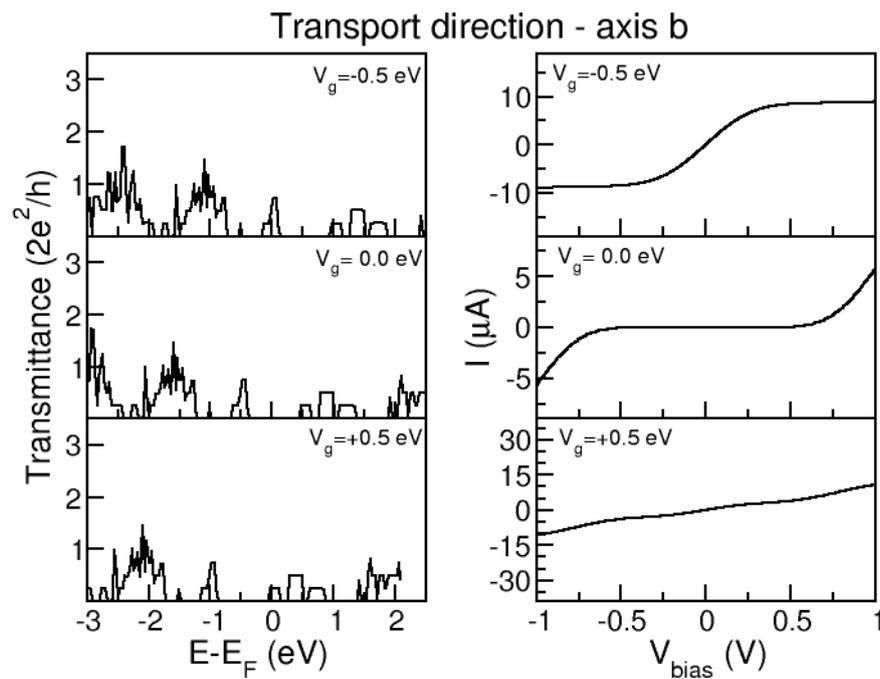


Figura 5.8: Curve di trasmittanza e I-V per direzione b

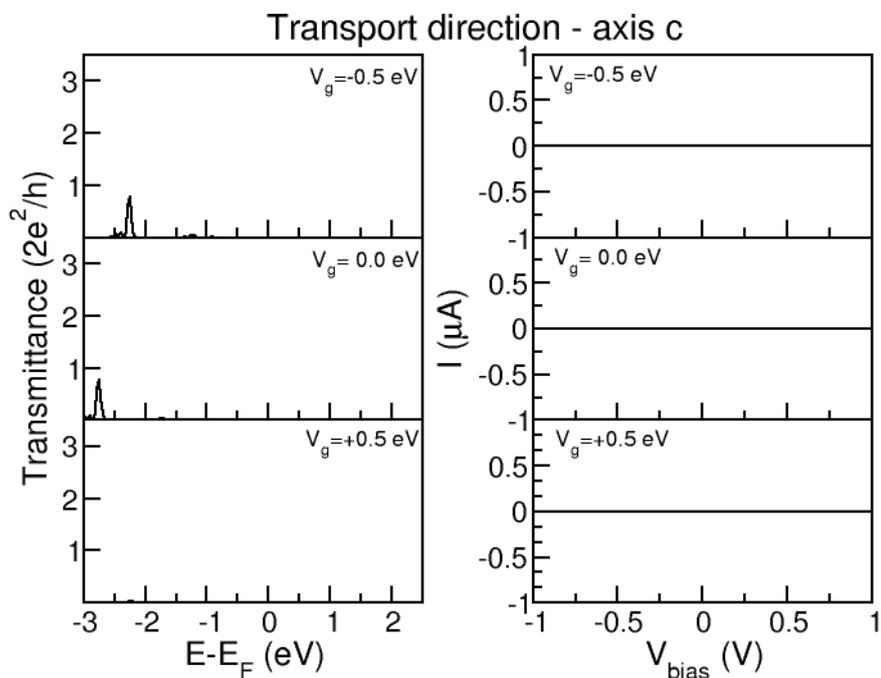


Figura 5.9: Curve di trasmittanza e I-V per direzione c

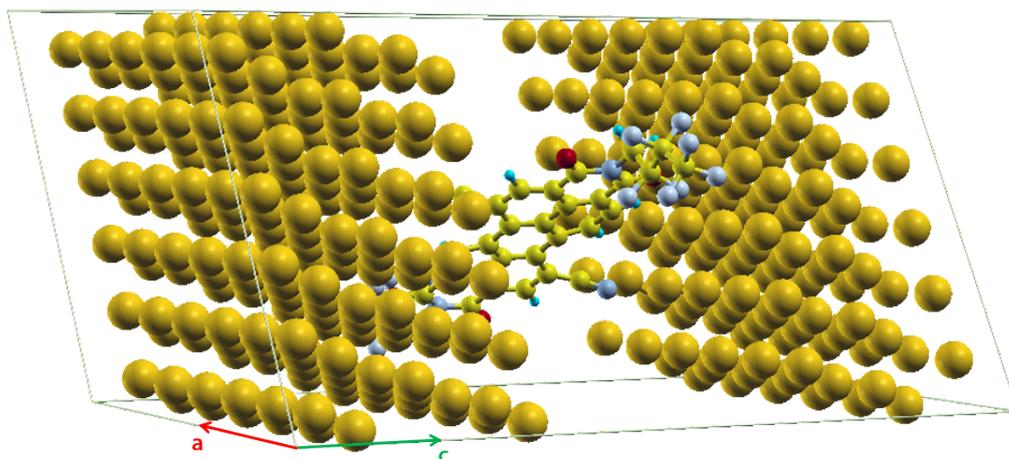


Figura 5.10: Cella del sistema a due terminali con elettrodi e una singola molecola al centro

HPC avanzate e macchine di grande dimensioni sarebbe inaffrontabile. Dopo la fase di rilassamento la molecola mantiene l'orientazione iniziale con il piano degli anelli aromatici quasi ortogonale alla superficie, a meno di una distorsione fuori dal piano dei legami CN (5.11 a) che tendono ad ottimizzare l'interazione con gli ori primi vicini.

Finita la fase di ottimizzazione strutturale abbiamo effettuato il conto auto consistente per indagare le proprietà di ground state del sistema e ottenere la struttura a bande. In figura 5.12 è riportata la pDOS (densità degli stati proiettata) ottenuta per il sistema, in cui sono state graficate separatamente i contributi dell'oro e della molecola. Il grafico è stato centrato sul valore dell'energia di Fermi del sistema. A questa energia ci sono stati sia per l'oro sia per la molecola e quindi ci può essere conduzione. Si nota in particolare che, in seguito alla formazione dell'interfaccia, il LUMO della molecola risulti sostanzialmente allineato con il livello di Fermi dell'Oro. Questa è esattamente il prerequisito descritto in precedenza per l'ottenimento di semiconduttori organici di tipo N, per provare questo in figura 5.12 abbiamo riportato la DOS della molecola isolata. L'allineamento delle curve è stato fatto allineando i livelli di semicore ad alta binding energy. La posizione degli stati elettronici molecolari sono facilmente rintracciabili anche nel sistema di interfaccia. Gli stati di energia di Fermi corrispondono appunto, ai primi stati vuoti della molecola.

Una volta ottenuto il ground state, si è proceduto con il conto *want* e si è ottenuta la curva di trasmittanza riportata in figura 5.13, la trasmittanza è stata calcolata solamente lungo la direzione *c*, che è l'unica dove si poteva avere conduzione attraverso la molecola. Dalla curva di trasmittanza si può notare che vi è un picco all'energia di Fermi e ciò vuol dire che gli elettroni hanno una buona probabilità di attraversare il sistema. Passando dalla curva di trasmittanza alla curva IV (figura 5.14) possiamo notare che il sistema sia omhnico.

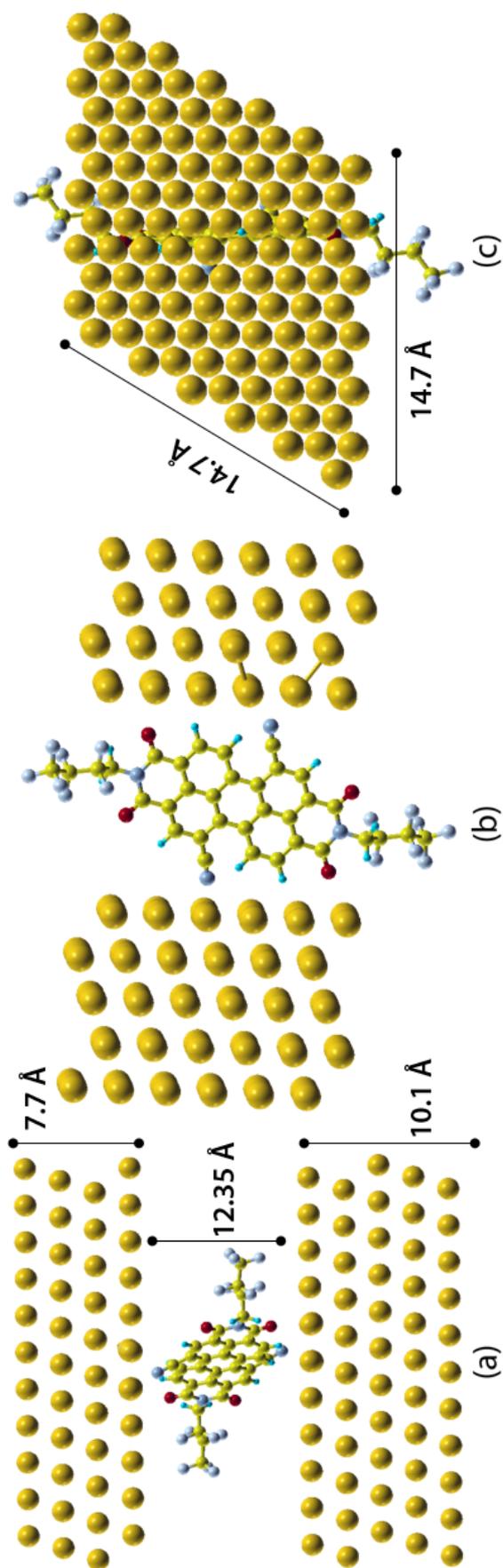


Figura 5.11: Schema del sistema a due terminali riportante le dimensioni fisiche in Å

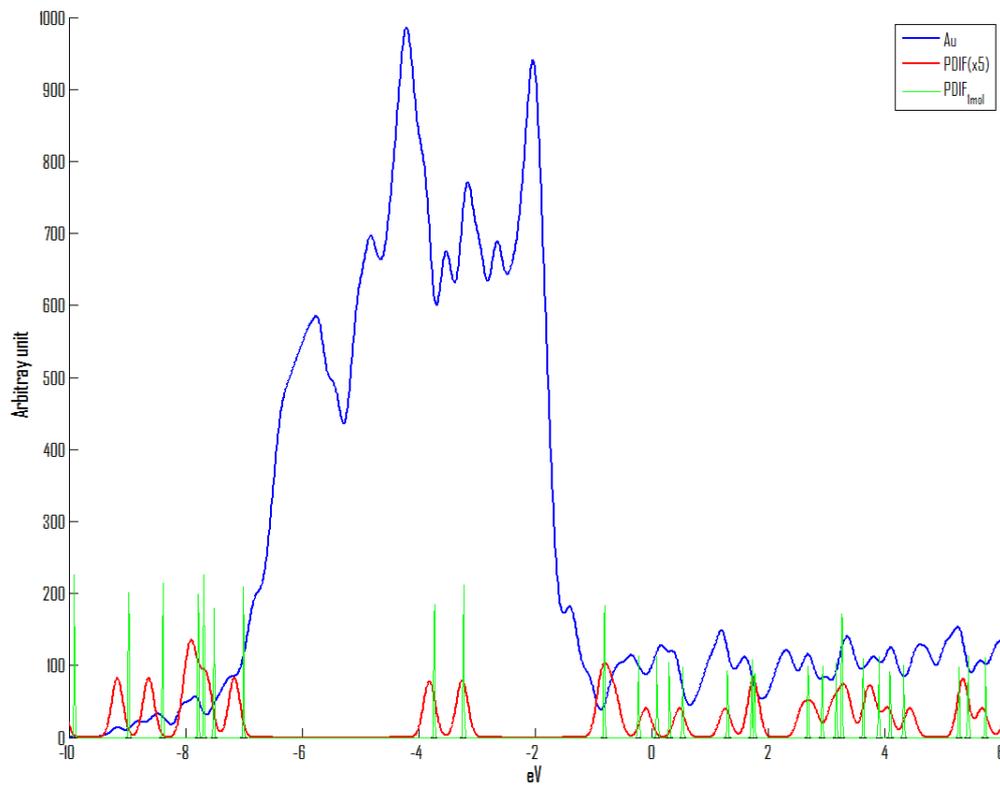


Figura 5.12: Densità degli stati del sistema a due terminali. In blu è rappresentata la DOS dell'Au e in rosso la DOS del $PDI - FCN_2$ nel sistema a due terminali. In verde è rappresentata la DOS di molecola isolata del $PDI - FCN_2$.

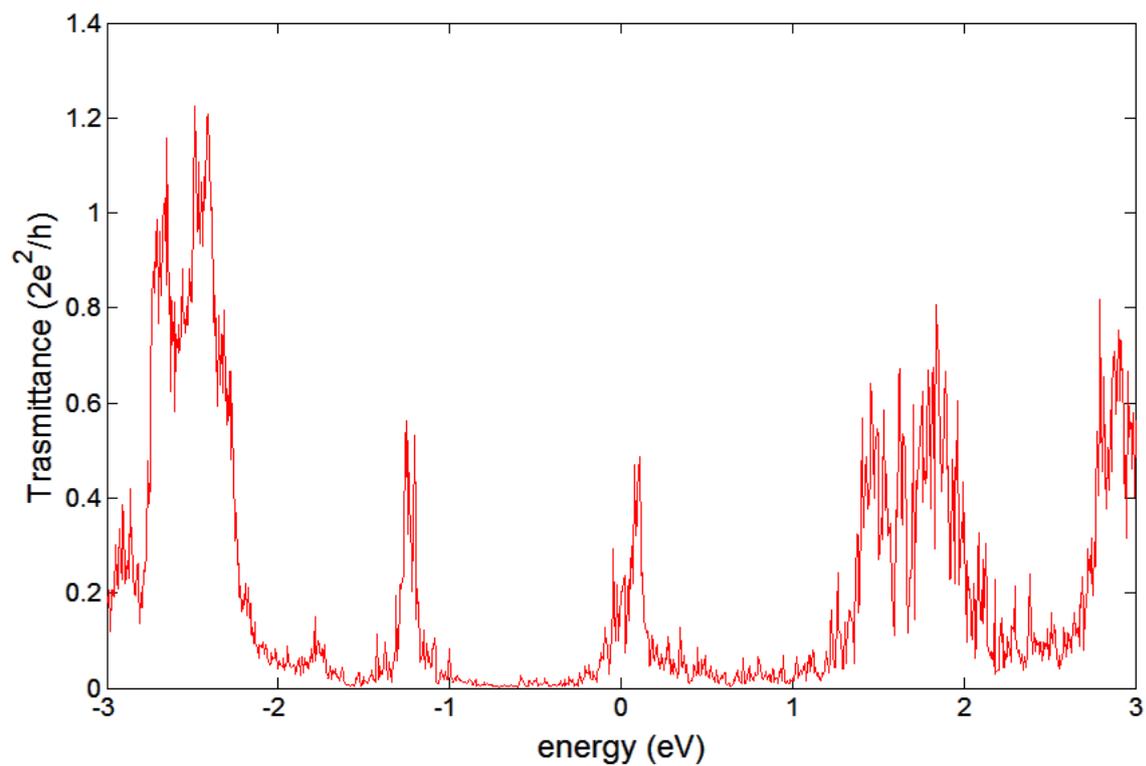


Figura 5.13: Curva di trasmittanza in direzione c per il sistema a due terminali

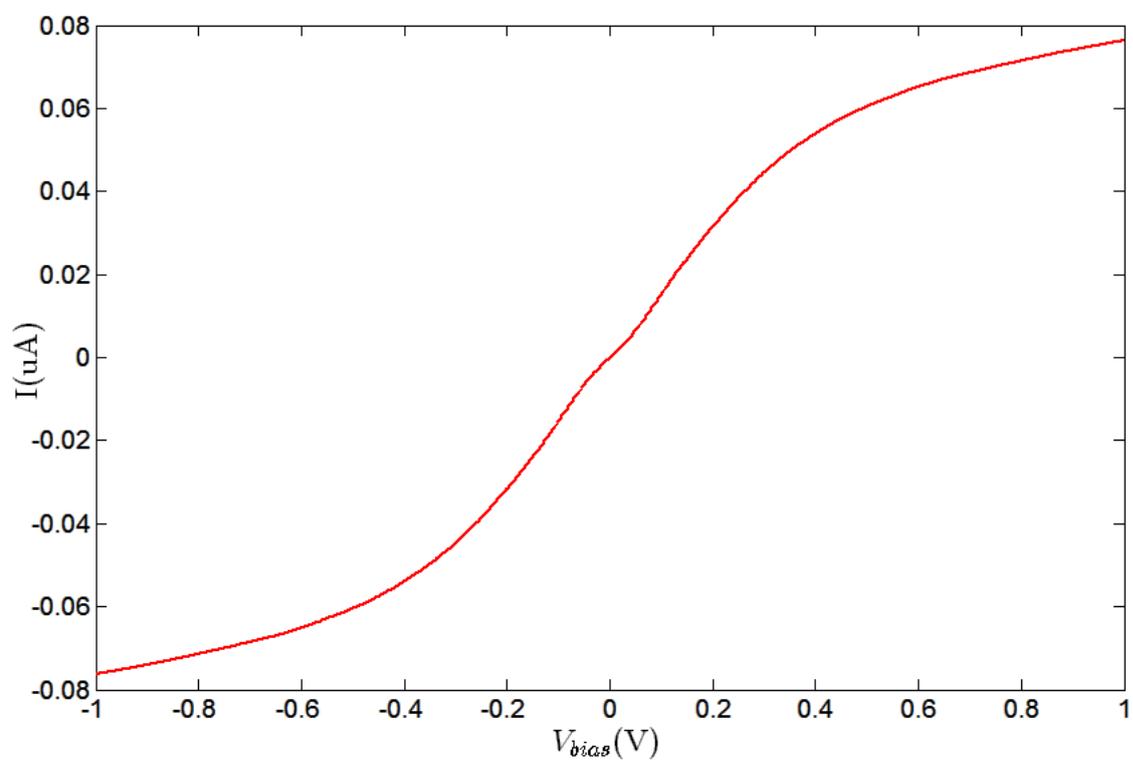


Figura 5.14: Curva I-V in direzione c per il sistema a due terminali

Capitolo 6

Conclusioni

In questo lavoro di tesi sono state evidenziate alcune problematiche relative alle macchine exascale e all'evoluzione dei software che saranno eseguiti su questi sistemi, ma si prende in esame principalmente la necessità del loro sviluppo, in quanto indispensabili per lo studio di problemi scientifici e tecnologici di più grandi dimensioni e a diverse risoluzioni. Basti pensare che il problema esaminato nel capitolo 5 ha richiesto 8 milioni di ore di calcolo e l'utilizzo di 16384 core. Questo poiché ciò che sulle macchine attualmente esistenti risulta particolarmente complesso, sui sistemi exascale sarà risolvibile con maggiore efficacia e minor tempo. Di seguito vengono riportate le conclusioni relative alle varie tematiche trattate precedentemente.

Misure di efficienza energetica

In questo capitolo sono state analizzate le performance di un benchmark, SiO_2 , che si avvicina ad una reale applicazione che l'utente lancia realmente su un sistema HPC, inoltre per la prima volta si è potuto avere il confronto delle prestazioni di QE su architetture differenti: nVIDIA K20, Intel Xeon Sandy Bridge (2.1 GHz e 3.1 GHz) e Intel Xeon Phi. I risultati, per quanto riguarda quelli su CPU, sono risultati come atteso dalle specifiche tecniche: il processore da 2.1GHz è risultato più lento, ma più efficiente dal punto di vista energetico, rispetto al 3.1GHz. Per quanto riguarda il run su GPU, non è stato notato, dal tempo di vista time to solution, un elevato speed up rispetto alla versione CPU, questo perché è stato utilizzato un sistema di piccole dimensioni. Per quanto riguarda la performance energetica la GPU è risultata quella meno efficiente. La versione per Xeon Phi ha mostrato una buona efficienza energetica rispetto alla versione CPU, ma dal punto di vista del time to solution risulta più rapido della CPU da 2.1GHz, ma più lento di quello da 3.1 GHz. Si precisa che la versione Xeon Phi è ancora a livello di prototipo e quindi si potrà fare un confronto diretto con la versione per GPU solamente dopo un ulteriore sviluppo ed ottimizzazione di questa.

Aumento della località dei dati

È stata esposta una strategia per l'aumento della località della distribuzione dati sui processori dei vettori G di QE. Si è implementata una distribuzione dati a blocchi in modo tale che con la futura implementazione della FFT, spiegata in 4.3, il numero di comunicazioni complessive diminuisca. Purtroppo non si sono riusciti ad eseguire benchmark e performance per due motivi: il primo è che la nuova FFT non è ancora stata implementata e il secondo è che non è ancora disponibile una rete tale per cui questa nuova distribuzione di dati porti ad un reale vantaggio.

Studio di un materiale di interesse tecnologico

In questo capitolo abbiamo analizzato le proprietà di un materiale organico il $PDI - FCN_2$, molto interessante dal punto di vista tecnologico poiché è stato proposto come candidato ideale per la costruzione di transistor organici di tipo OFET. Inoltre è stata presa questa simulazione, molto costosa dal punto di vista computazionale, per testare, in vista dell'exascale, la robustezza del codice e per la scelta di parametri da ottimizzare per riuscire a portare a termine la simulazione. Inizialmente partendo dai dati di cella estratti dall'articolo [8] abbiamo studiato il volume del cristallo 3D di $PDI - FCN_2$. Tramite il software Pw siamo riusciti a ottenere le bande del cristallo per la direzione $\Gamma \rightarrow X$ fig. 5.5 e si può vedere come queste sono in accordo con le bande ottenute dal conto GW riportate nell'articolo [8] figura 5.4. Successivamente con il software WanT si è proceduto al calcolo di conduttanza e si sono ottenute le curve I-V per le 3 direzioni cristallografiche, riportate nelle figure 5.7, 5.8, 5.9. Dalle immagini si può affermare che il cristallo 3d conduce nelle due direzioni cristallografiche a,b perpendicolari alla struttura planare aromatica. In seguito abbiamo messo a punto il sistema a due terminali composto da due elettrodi d'oro con una singola molecola di $PDI - FCN_2$ posizionata al centro (figura 5.11). Dopo la fase di rilassamento la molecola mantiene l'orientazione iniziale con il piano degli anelli aromatici quasi ortogonale alla superficie, a meno di una distorsione fuori dal piano dei legami CN (5.11 a) che tendono ad ottimizzare l'interazione con gli ori primi vicini. Dopo la fase di ottimizzazione e calcolo della struttura elettronica siamo andati a calcolare la pDOS del sistema (figura 5.12) dove si può vedere che in seguito alla formazione dell'interfaccia, il LUMO della molecola risulti allineato con il livello di Fermi dell'oro e questo è esattamente il prerequisito per l'ottenimento dei semiconduttori di tipo N. Eseguendo il conto di conduzione lungo la direzione c del sistema si è ottenuta la curva di trasmittanza (figura 5.13) e di tensione corrente (figura 5.14), dalla prima possiamo notare che vi è un picco all'energia di Fermi e ciò ci indica che gli elettroni hanno una buona probabilità di attraversare il sistema, mentre la seconda conferma il comportamento ohmico del sistema.

In conclusione, in questo capitolo abbiamo presentato come le tecnologie exascale descritte nei capitoli precedenti possano essere utilizzate per la risoluzione di problemi fisici di interesse nel campo della materia condensata per applicazioni

nel campo dell'elettronica molecolare. L'interesse per i risultati qui ottenuti è duplice:

- questo sistema modello ci ha permesso di testare ed ottimizzare i codici e le loro performance nel caso di applicazioni su larga scala che hanno richiesto codici altamente performanti e capacità di supercalcolo.
- I risultati ottenuti sono rilevanti dal punto di vista fisico, perché confermano la potenziale capacità dei derivati del perilene nel campo dell'elettronica molecolare sia per la realizzazione di sistemi 3D, quali OFET di tipo N, sia per la realizzazione di dispositivi a singola molecola, che rappresentano il limite sulla nanoscala dei dispositivi elettronici di prossima realizzazione.

La caratterizzazione delle proprietà di trasporto elettronico di tali materiali è stata possibile solo attraverso l'utilizzo di approcci da primi principi che danno accesso diretto alla struttura atomica ed elettronica dei sistemi in esame, senza il bisogno di introdurre parametri semi-empirici e/o sperimentali. Il buon accordo fra i teorici risultati ottenuti e quelli sperimentali (ancora preliminari, esperimenti in parte in corso) dimostra inoltre il forte potere predittivo della teoria che permette di guidare la scelta di materiali, architetture e sistemi attraverso *esperimenti virtuali* al calcolatore.

Appendice A

Attuale Distribuzione G in QE

Considerando l'organizzazione del codice QE, la distribuzione dati come descritto nel capitolo 4 viene generato dalla funzione *pstickset* che è contenuta nel Module Fortran *stick_set* ed è localizzato nella cartella Module del pacchetto QE; a sua volta *pstickset* si serve delle subroutine contenute nel Module Fortran *stick_base*. Il flusso dell'algoritmo è rappresentato nell'infografica in fig A.1 i numeri delle chiamate corrispondono a quelli dell'elenco sottostante dove si riporta la descrizioni delle varie subroutine

1. **pstickset:** questa subroutine prende in input tutte le osservabili fisiche, per creare la griglia, e le variabili e i descrittori MPI per creare la distribuzione dei dati sui processori. La subroutine si occupa di allocare le strutture dati, queste verranno riempite dalle subroutine chiamate da *pstickset*. In output vengono ritornati, sia per la griglia dei potenziali (ST) che per quella delle funzioni d'onda (STW), un descrittore che contiene tutte le informazioni per gestire l'algoritmo FFT e due vettori contenenti il numero di vettori G associati ad ogni processore, uno per le funzioni d'onda e l'altro per i potenziali. Se esiste la griglia dei vettori di Smooth (STS), vengono ritornati anche per questa un descrittore ed un vettore analoghi a quelli appena descritti.
2. **sticks_map:** viene generata una mappa 2D della sfera tridimensionale dei vettori G in spazio reciproco per ogni tipo di vettore: funzione d'onda, potenziale e smooth. La mappa 2D è contenuta nella griglia denominata ST, questa ha come coordinate gli indici di Miller del sistema in esame e come valore di ogni cella il numero di vettore G con le coordinate prese in considerazione. Come si può vedere nell'infografica, la matrice viene riempita sommando il contributo di ogni singolo processore calcola parallelamente gli stick lungo z; alla fine, per ogni processore, viene fatta un'operazione di somma attraverso una MPI reduce sugli stick lungo z. Per le griglie STW e STS, il procedimento è analogo a quello appena descritto per la ST.
3. **sticks_count:** data ogni griglia, conta quanti vettori G di ogni tipo ci sono per coppia di indici di Miller. Vengono ritornati cinque array: due conten-

gono gli indici di Miller, gli altri tre i conteggi corrispondenti ai tre tipi di vettori G.

4. **sticks_sort**: costruisce un array di indici ordinati in modo crescente dei vettori che contengono i conteggi dei potenziali, smooth e funzioni d'onda. Viene dato un peso diverso ai tre tipi di vettore, in modo tale da avere il vettore suddiviso in tre aree: una prima area in cui ci sono gli indici che contengono tutti e tre i tipi, una parte centrale che contiene gli indici che contengono sia potenziali che smooth, ed una parte finale che contiene gli indici con i restanti potenziali.
5. **sticks_dist**: ritorna tre mappe corrispondenti alla distribuzione sui processori dei vettori G dei tre tipi. Seguendo l'ordine ottenuto da *sticks_sort* distribuisce sui processori prima gli indici che contengono vettori dei tre tipi (funzioni d'onda, smooth e potenziali), poi quelli che contengono sia potenziali che smooth ed infine quelli rimasti contenenti solo i potenziali. L'algoritmo è fatto in modo da ottimizzare il bilanciamento del carico di lavoro, quindi assegna i vettori G sempre al processore che ne ha meno.



IN

OUT

gcut Cut off potential
gkcut Cut off (G + K)
gcuts Cut off smooth mesh
gamma_only true for gamma calc
bg reciprocal space base vector
dfftp dense grid type
dffts smooth grid type
mype index in band group
root root index in band group
nproc number proc in band group
comm intraband MPI_COMM
nogrp_ number of band group

ngw number of G wave for proc
ngm number of G potential for proc
ngs number of G smooth for proc
dfftp dense grid type, this datatype contains information of G potential and wave matrix and relative Proc distribution.
dffts smooth grid type, this datatype contains information of G smooth and wave matrix and relative Proc distribution.

Description
 Pstickset subroutine allocate the data on processor and initialize FFT descriptor.

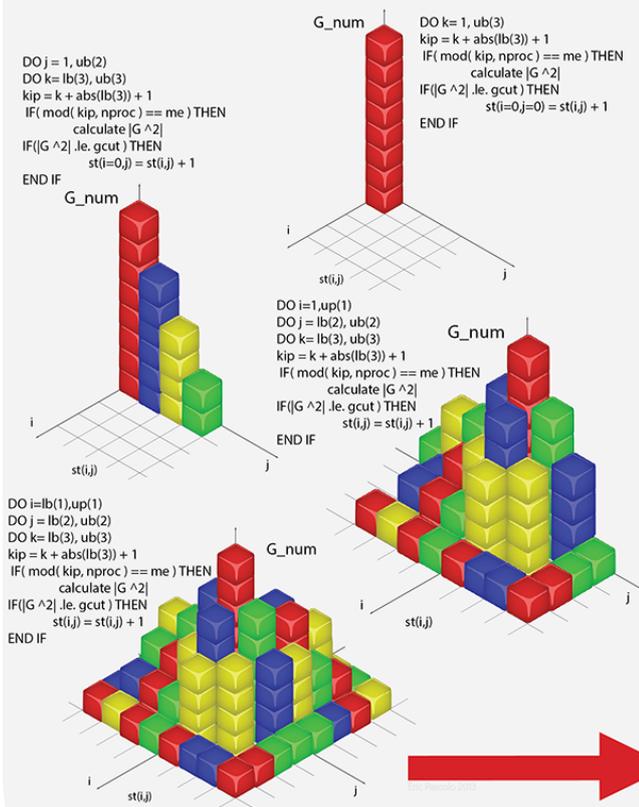
1:pstickset

2:sticks_map

Description

Sticks_map subroutine generate 2D maps of 3D G sphere. Mapping is split in four step. For each step calculate a number of Gvectors at fixed (x,y) miller index and z coordinates collapse. The calculus is parallelized along z direction. MPI_Allreduce(SUM) combine sticks of all processor. In figure there are potential maps example, but the same work is done for wave and smooth maps.

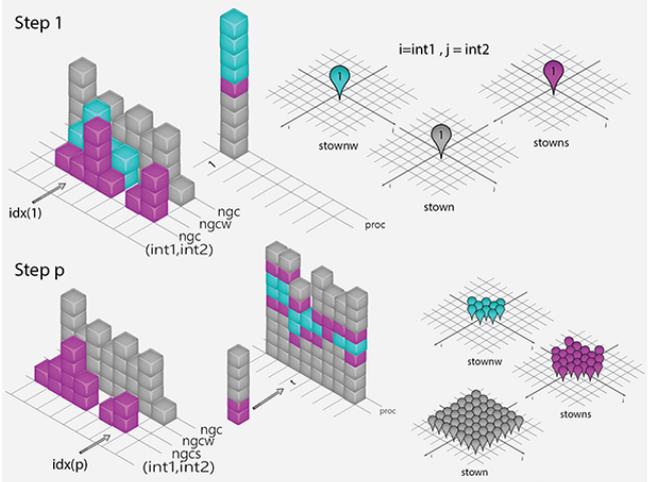
tk not gamma_only
ub upper miller index
lb lower miller index
bg(:,1) => b1 base vector
bg(:,2) => b2 base vector
bg(:,3) => b3 base vector
gcut Cut off potential
gkcut => gcutw Cut off (G + K)
gcuts Cut off smooth mesh
mype => me index proc in band group
nproc number of proc in band group
comm intraband MPI_Comm
st sticks map for potential (OUT)
stw sticks map for wave function (OUT)
sts sticks map for smooth mesh (OUT)



5:sticks_dist

nst => nct number of st elements
nstp => ncp (OUT) counter sticks Gp
nstpw => ncpw (OUT) counter sticks Gw
nstps => ncps (OUT) counter sticks Gs
sstp => ngp (OUT) number per Gp/proc
sstpw => ngpw (OUT) number Gw/proc
sstps => ngps (OUT) number Gs/proc
st => stown (OUT) dist Potential G of proc
stw => stownw (OUT) dist Wave G of proc
sts => stowns (OUT) dist Smooth G of proc

tk not gamma_only
ub upper miller index
lb lower miller index
idx ascending ordered map
ist(:,1) => in1 x coordinates Miller index
ist(:,2) => in2 y coordinates Miller index
ist(:,4) => ngc potential G array
ist(:,3) => ngcw wave G array
ist(:,5) => ngcs smooth G array
nproc number of proc in band group

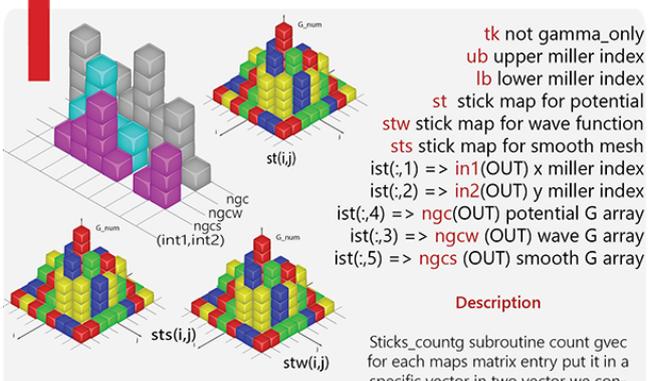


Description
 Subroutine distribute G calculated from sticks_map among processor, balancing workload. For assignment we follow order of idx array. The matrix stown, stowns and stownw contains the G vectors job for each processor.

4:sticks_sort

nst => nct total number potential stick
idx index ascending maps
nproc number of proc in band group
ist(:,4) => ngc potential G array
ist(:,3) => ngcw wave G array
ist(:,5) => ngcs smooth G array

Description
 This subroutine build a sort map (idx) of vector ngc,ngcs,ngcw.



tk not gamma_only
ub upper miller index
lb lower miller index
st stick map for potential
stw stick map for wave function
sts stick map for smooth mesh
ist(:,1) => in1 (OUT) x miller index
ist(:,2) => in2 (OUT) y miller index
ist(:,4) => ngc (OUT) potential G array
ist(:,3) => ngcw (OUT) wave G array
ist(:,5) => ngcs (OUT) smooth G array

Description
 Sticks_countg subroutine count gvec for each maps matrix entry put it in a specific vector in two vector we conserve associated coordinates.

3:sticks_count

Figura A.1: Infografica rappresentante lo schema di chiamate che porta alla generazione della attuale distribuzione dati di Quantum ESPRESSO

Bibliografia

- [1] Architectures and technology for extreme scale computing.
- [2] Luis A. Agapito, Andrea Ferretti, Arrigo Calzolari, Stefano Curtarolo, and Marco Buongiorno Nardelli. Effective and accurate representation of extended bloch states on finite hilbert spaces. Phys. Rev. B, 88:165127, Oct 2013.
- [3] Tsumura Koezuka Ando. Macromolecular electronic device: Field effect transistor with a polythiophene thin film. Applied Physics Letters, 49(18), 1986.
- [4] Bartolini Cacciari Cavazzoni Tecchiolli Benini. Unveiling eurora - thermal and power characterization of the most energy-efficient supercomputer in the world.
- [5] Kogge Bergman Borkar Campbell and et all. Exascale computing study: Technology challenges in achieving exascale system.
- [6] Pascolo Affinito Cavazzoni. Performance and energy efficiency in material science simulation on heterogeneous architectures. High Performance Computing and Simulation Conference, 2014.
- [7] Gerbrand Ceder. Opportunities and challenges for first-principles materials design and applications to li battery materials. MRS Bulletin, 35:693–701, 9 2010.
- [8] Colle Grosso Cassinese Centore. Structural, electronic and vibrational proprieties of n,n'-1h,1h-perfluorobutyl dicyanoperylene-carboxydiimide (pdi-fcn2) crystal. The Journal of Chemical Physics, 2013.
- [9] Cavazzoni Chiarotti. A parallel and modular deformable cell car-parrinello code. Computer Physics Communications, 1999.
- [10] Petitet Whaley Dongarra Cleary. Hpl - a portable implementation of the high-performance linpack benchmark for distributed-memory computers.
- [11] Datta. Electronic Transport in Mesoscopic System.
- [12] Padoin de Oliveira Velho Navaux. Time-to-solution and energy-to-solution: A comparison between arm and xeon. In Applications for Multi-Core Architectures.

- [13] Dongarra and et all Exascale Mathematics Working Group. Applied mathematics research for exascale computing.
- [14] Guarrasi Frigio Emerson Erbacci. Scalability improvements for dft codes due to the implementation of the 2d domain decomposition algorithm. PRACE White Paper, 2013.
- [15] Pawel Gepner, Victor Gamayunov, and David L. Fraser. Early performance evaluation of avx for hpc. Procedia Computer Science, 4(0):452 – 460, 2011. Proceedings of the International Conference on Computational Science, {ICCS} 2011.
- [16] P. Giannozzi and et al. Quantum espresso: a modular and open-source software project for quantum simulations of materials. Journal of Physics: Condensed Matter, 2009.
- [17] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, and Kristin A. Persson. Commentary: The materials project: A materials genome approach to accelerating materials innovation. APL Materials, 1(1):-, 2013.
- [18] D. D. Johnson. Modified broyden’s method for accelerating convergence in self-consistent calculations. Phys. Rev. B, 38:12807–12813, Dec 1988.
- [19] Auweter Bode Brehm Huber Kranzlmüller. Principles of energy efficiency in high performance computing. Lecture Notes in Computer Science, 2011.
- [20] Baoand Locklin. Organic Field Effect Transistors.
- [21] Jones Facchetti Wasielewski Marks. Tuning orbital energetics in arylene diimide semiconductors. materials design for ambient stability of n-type charge transport. Journal of the american chemical society, 2007.
- [22] Miller. Rationale and challenges for optical interconnects to electronic chips. Journal of Parallel and Distributed Computing, 180(41), 1997.
- [23] Göddekea Strzodkab Mohd-Yusofc and et al. Exploring weak scalability for fem calculations on a gpu-enhanced cluster. Parallel Computing, 2007.
- [24] Molinari Alves Chen Facchetti Morpungo. High electron mobility in vacuum and ambient for pdif-cn2, single cristal transistor.
- [25] Shalf Dosanjh Morrison. Exascale computing technology challenges. VECPAR 2010 LNCS, 2011.
- [26] Mijalkovic Green Nejim Whiting Rankov Smith Hallsand Murphy. Modelling oforganic field-effect transistors for technology and circuit design. In Proceedings of INTERNATIONAL CONFERENCE ON MICROELECTRONICS, Grenoble, France, May 11–14 2008.

-
- [27] Calzolari Marzari Souza Buongiorno Nardelli. Ab initio transport properties of nanostructures from maximally localized wannier functions. Phys. Rev. B, 69:035108, Jan 2004.
- [28] Ceder Persson. The stuff of dreams. Scientific American, 309(6), Dec 2013.
- [29] Jost Rabenseifner, Hager. Hybrid mpi/openmp parallel programming on clusters of multi-core smp nodes. In IEEE, editor, Parallel, Distributed and Network-based Processing, 2009 17th Euromicro International Conference on.
- [30] Goddeke Komatitsch Ramirez and et al. Energy efficiency vs. performance of the numerical solution of pdes: An application study on a low-power arm-based cluster. Journal of Computational Physics, 2013.
- [31] Vivek Sarkar, William Harrod, and Allan E Snavely. Software challenges in extreme scale systems. Journal of Physics: Conference Series, 180(1):012045, 2009.
- [32] Roland Schulz. 3d fft with 2d decomposition.
- [33] Liang Wang Ananth Dodabalapur Brooks A. Jones Antonio Facchetti Michael R. Wasielewski Tobin J. Marks Taeho Jung, Byungwook Yoo. Nanoscale n-channel and ambipolar organic field-effect transistors, 2006.
- [34] Andreas Weissel and Frank Bellosa. Process cruise control-event-driven clock scaling for dynamic power management. In Proceedings of the International Conference on Compilers, Architecture and Synthesis for Embedded Systems (CASES 2002), Grenoble, France, October 8–11 2002.

Grazie

Questa tesi non rappresenta solo la fine della mia laurea magistrale, ma rappresenta la fine di un periodo della mia vita iniziato ormai tre anni fa. Quello che è racchiuso in questa tesi non è solo il mio lavoro presso il CINECA, ma è il frutto di tutta l'esperienza che ho accumulato nel corso di questi tre anni ed è frutto soprattutto degli incontri che ho avuto e delle relazioni con le persone con cui ho lavorato. Infatti libri, dispense, ed esami insegnano tanta teoria, ma alla fine chi ti insegna veramente sono le persone e solo intrecciando rapporti, confrontandosi e collaborando si cresce e si impara qualcosa di nuovo. È per questo che alla fine di questa tesi esiste il capitolo dei ringraziamenti: un capitolo per ringraziare tutti quelli che mi hanno fatto crescere, insegnato, supportato in questi 3 anni della mia vita.

Visto che sono in tema "tesi", la prima persona che devo ringraziare è **Carlo**, il mio Boss in CINECA! Naturalmente lo ringrazio per avermi sostenuto in questo periodo di tesi e per avermela pazientemente corretta. In verità Carlo per me ha fatto molto di più: infatti quando sono arrivato in CINECA, dopo un anno di magistrale a Modena, a mio avviso non troppo motivante, ero un po' abbacchiato, disorientato e disinteressato a qualsiasi argomento tecnico-scientifico; Carlo è riuscito a farmi tornare l'interesse per argomenti più tecnici che scientifici, lavorando fianco a fianco con me, portandomi ai convegni, dandomi anche la possibilità di pubblicare un articolo e soprattutto insegnandomi il mestiere del programmatore HPC. Quindi... Grazie Carlo!

Rimanendo in tema CINECA c'è un'altra persona che vorrei ringraziare, anche se forse mi avrebbe "strozzato" volentieri più di una volta quando lo tormentavo chiedendogli continuamente spiegazioni...vero **Fabio**?! Ho conosciuto Fabio Affinito per la mia tesi triennale, era il mio correlatore e già allora dava segni di cedimento nel sopportarmi perché gli chiedevo troppe cose. Pensate poi quando ho scritto il mio primo articolo e lui era uno degli autori...! Quindi Grazie mille Fabio per tutte le volte che mi hai sopportato ed aiutato!

Già che sto parlando di CINECA finisco i ringraziamenti rivolti a chi ho conosciuto lì. Un grande grazie va a **Daniela** (la capa delle macchine) e ai suoi sistemisti **Sbrighi, Marcello, Giusi, Vanni, Isabella** e a tutti gli altri per avermi aiutato a capire quei complicati sistemi che sono i supercomputer; grazie a **Giovanni** (big boss dei progetti europei) per avermi ricordato per un anno e mezzo che dovevo sbrigarmi a laurearmi; grazie a **Cinzia** per avermi fatto capire come funzionano i

complicati meccanismi aziendali e a **Andy** (l'inglese del CINECA) perché senza di lui il mio articolo in inglese non avrebbe mai visto la luce; grazie a tutti i ragazzi dello User Support **Alessandro, Nicola, Mirko, Silvia** per avermi supportato e aver risposto sempre a tutte le mie domande! Insomma grazie a tutti quelli del dipartimento HPC, perché sicuramente senza di voi non sarei mai riuscito a realizzare questa tesi.

Ritornando alla tesi, un grande grande grazie ad **Arrigo** Calzolari, per avermi pazientemente spiegato e corretto nella tesi tutto quello che non so di fisica della materia, e credetemi è molto, e soprattutto per essersi preso a cuore il mio lavoro di tesi! Quindi grazie infinite Arrigo, magari ci fossero molti più scienziati come te!

Devo ringraziare il Prof. **Renato** Colle per la fiducia datami, affidandomi il sistema fisico di questa tesi da computare e soprattutto per non aver perso la pazienza quando ogni volta che mi chiamava io gli dicevo che eravamo fermi perché il conto era in coda e la macchina piena! Grazie anche al Prof. **Daniel** Remondini che ha accettato di farmi da relatore a scatola chiusa, senza sapere, o quasi, di cosa parlasse la mia tesi! E per ultimo, ma non ultimo, devo ringraziare il Prof. **Mauro** Ferrario che mi ha permesso di iniziare la tesi, e per tutto il tempo in cui ero a Modena, ha risposto ai miei dubbi computazionali.

Proprio perché sono convinto che molte delle persone che ho incontrato abbiano contribuito alla mia formazione e, quindi indirettamente a questo lavoro di tesi, devo ringraziare i ragazzi del gruppo di Luca Benini, soprattutto **Andrea** Bartolini per avermi spiegato un po' di cose sull'energy efficiency e **Francesco** Fraternali per aver lavorato un bel po' di tempo insieme a me su EURORA e per aver tradotto il mio inglese a quel simpatico tedesco che parlava giapponese! Sempre in tema di "lavori esterni con CINECA", devo ringraziare **Luca** Zanni e **Riccardo** Zanella per avermi dato la possibilità di lavorare con loro su SGP (un software che potrebbe essere usato in CSI!).

Il mio percorso formativo di questi tre anni non è assolutamente separabile dall'Associazione Culturale *inco.Scienza - incontriamo la Scienza* di cui io sono uno dei fondatori. Lavorare per questa Associazione, come speaker e organizzatore di conferenze e laboratori scientifici, ha dato una grandissima spinta alle mie abilità oratorie e gestionali. È per questo che devo ringraziare tutti i soci, soprattutto **Matteo** Giganti, il mio consulente pubblicitario e i miei "comunicazionisti" **Gianluca** Dotti e **Erika** Nerini che hanno passato molte ore con me a insegnarmi come si migliora la comunicazione e la divulgazione scientifica. Un grande grazie, inoltre, a tutte e le persone che hanno collaborato con l'Associazione e hanno reso possibile la sua fondazione. In primis **Pierluigi** Giacobazzi (detto Pier). Pier è stato il primo fan dell'Associazione, perché se non fosse per lui io non avrei mai organizzato il mio primo ciclo di conferenze "Young Doctor for Science" al Planetario di Modena e, molto probabilmente, senza di lui non sarebbe mai nata l'Associazione. Quindi grazie Pier per essere stato così pazzo da aver accolto quella pazza idea da un ragazzo di 22 anni. Secondo, ma non meno importante, **Enrico** Artioli, che ha sempre dato sostegno alla nostra idea di permettere ai

giovani di divulgare la scienza e che ci ha dato tutto l'appoggio di cui avevamo bisogno. Grazie Enrico per avermi sempre coinvolto nelle tue iniziative e per aver dato fiducia a un ragazzo giovane e inesperto!

Un grande e immenso grazie a tutti "quelli" di Fisica modenese perché se non fosse stato per per voi chissà dove sarei finito adesso, quindi anche se per me l'incontro non sempre è stato positivo, mi ha fatto crescere e anche da questa esperienza ho imparato molto. Grazie comunque ai Prof. che mi hanno dato una mano quando ne avevo bisogno. Se a Modena alla magistrale non mi sono trovato molto bene, questo non è vero per Bologna. Ci sono in particolare due persone che devo veramente ringraziare di cuore perché senza di loro, la tesi probabilmente ci sarebbe stata comunque, ma la laurea forse no! Grazie alle mie compagne di pranzo preferite Anna ed Elena, peccato non esserci laureati insieme... mi sarebbe piaciuto! Un Grazie ad **Anna** per le nostre chiacchierate, discussioni, risate durante Schedsim e per essere stata un po' troppo puntigliosa! Un grazie ad **Elena** per essere una fisica tutta d'un pezzo e per avermi dato un po' di ripetizioni di svariate materie! Un grazie a tutti gli altri compagni di corso che mi hanno tenuto compagnia fuori dall'aula nell'attesa delle correzioni di Campanini.

Fortunatamente nella vita non c'è solo il lavoro, ma anche le vacanze, lo svago e il divertimento.. Quindi per iniziare voglio ringraziare i miei amici **Elisa, Gianluca, Los, Giulia, Balta, Free, Pollo, Vero, Giulio, Vale, Robby** con cui trascorro delle piacevoli serate, ma a cui sfortunatamente ogni tanto tocca sentirmi cantare! Un grazie in particolare a Veronica per aver composto una canzone in mio onore mentre eravamo in quella bellissima e fredda terra che è la Norve.

Un grazie ai miei amici mezzi mantovani e mezzi americani **Barbara** Arvani e **Federico** Perini per le piacevoli serate mantovane che mi hanno aiutato a distrarmi e per le videochiamate oltre oceano.

Un grande immenso grazie ad un'amica che ha cercato in tutti i modi e per un sacco di anni di insegnarmi l'inglese, ma che ormai "ci ha dato a mucchio" anche lei... grazie **Claudia** Allegretti!

Avventure come quelle che ho vissuto con te, posso dire che non le ho vissute mai con nessun altro. Ci conosciamo da una vita e anche se ci vediamo poco... da piccoli solo al mare, poi ogni tanto a Modena da grandi, ma quando lo facciamo solitamente salta fuori qualcosa di indimenticabile, grazie **Giagio** (Gianluca Barbati) per tutte le avventure irraccontabili che abbiamo vissuto insieme!

Un grazie anche al mio specialista in visione artificiale, e compagno di serate al Goblin, **Stefano** Alletto e consorte **Angelica** Cruz a cui piace tanto vedermi punzecchiare la mia ragazza; senza il palio di Cividale non berrei abbastanza birra per far lavorare il cervello tutto il resto dell'anno!

Un grazie anche a un altro compagno di serate, queste più filosofiche, **Francesco** Draisci che devo ringraziare per avermi insegnato due cose: la prima è che a volte vale la pena mettere da parte l'orgoglio per qualcosa a cui tieni molto, ma la seconda è che allo stesso tempo quando un amico è un po' troppo accondiscendente

deve farti venire qualche dubbio su quello che vuole realmente. Caro Fra, spero di poter assistere presto anche alla tua laurea magistrale.

Un grande grazie ai miei amici tedeschi **Betta** Bergamini e **Enrish** Oliva per le vacanze fatte assieme, tra cui l'Irlanda uno dei viaggi più belli che abbia mai fatto, e per tutte le serate tra Mirandola, Carpi e Modena! Un grazie particolare a te Betta che dici un sacco di cose, le mimi anche e poi ti chiedi come gli altri facciano a saperle.

Un grande grazie, ma grande davvero, al mio amico fotografo, che è anche ingegnere, ma soprattutto è uno degli altri due pazzi che ha fondato con me l'Associazione. Grazie mille per tutte le ore passate insieme, tra lavoro, gioco, chiacchiere, discussioni, birre alle Terrazze... e l'elenco potrebbe essere molto più lungo... quindi Grazie **Nicholas** Zanasi.

Grazie infinite Scema! A chi mi sto riferendo?? Naturalmente a **Nicoletta** Incerti. Se leggerai mai queste righe ti chiederai per quale motivo ti sto ringraziando... Beh il motivo è semplice perché di amiche come te se ne trova una nella vita se si è fortunati! Grazie davvero per avermi dato fiducia e grazie per essere sempre un po' bambina che è, allo stesso tempo, il tuo punto di forza e il tuo punto debole! Un grazie anche per essere la mia vicepresidente "inco.sciente". Grazie davvero di tutto, senza di te non mi sarei mai posto molte domande e di conseguenza non avrei trovato le risposte e soprattutto non avrei mai vissuto un sacco di esperienze che mi sono tornate e mi torneranno utili nella vita.

Adele...a te che devo dire? Non possono essere certo queste poche righe che rendono giustizia al mega super grazie che ti voglio esprimere per essere la mia compagna da quasi nove anni in questo strano viaggio che è la vita, per esserci sempre stata in ogni singolo momento in cui avevo bisogno e per avermi sempre aiutato nelle scelte difficili. Riguardo a questa tesi, è grazie a te che ho avuto la spinta per iniziarla e l'incoraggiamento necessario per portarla a termine. Grazie per aver portato pazienza in tutti quei momenti in cui lavoravo anziché passare del tempo insieme a te. Sono molto orgoglioso di avere accanto, come dico nella dedica, *una piccola grande donna* e non credo che tutti gli uomini possano avere la fortuna di trovare una compagna a cui affidarsi ciecamente come faccio io perché mi fido di te. Grazie di tutto Adele, senza di te avrei forse portato a termine un quarto di tutto il lavoro che sono riuscito a fare in questi anni e, nel mio caso, è proprio vero che dietro ad un uomo che ottiene un successo c'è sempre una grande donna.

Un grazie speciale va ai miei nonni **Mirta** e **Vasco** per tutto il tempo e le energie che mi hanno dedicato da quando ero piccolo fino ad ora e soprattutto grazie perché credono che il loro nipote sia speciale e molto più bravo di quello che è. Se sono cresciuto così è anche grazie a loro, a tutte le estati trascorse al mare e ai pomeriggi passati tra la loro casa e il loro cortile a giocare e a diventare grande aiutato dal loro affetto.

Mamma, Papà avete visto?! finalmente c'è l'ho fatta anche io a laurearmi! La vita negli ultimi anni non è stata facile, grazie infinite **Cinzia** e **Franco** per tutti i

sacrifici che avete fatto per farmi arrivare fin qua e farmi fare tutte le esperienze che mi hanno permesso di crescere e maturare. Grazie Pa per avermi spesso ricordato *che ho sempre fatto solo il mio dovere*, in fondo credo che sia il tuo modo per dirmi bravo; mi hai insegnato molte cose che sui libri non potevo trovare: lo yoseikan budo, a sciare e perfino ad orientarmi in un bosco... grazie Papà! Mamma, grazie di appoggiarmi sempre, anche quando non mi capisci o litighiamo; se non fosse stato per te che mi hai insegnato l'amore per la curiosità e come cercare di capire cosa aspettarsi dalla vita io non sarei mai arrivato qui, non sarei mai diventato due volte dottore, e non avrei raggiunto tanti altri traguardi... quindi veramente un grande Grazie Mamma!