

Rapporti tecnici

INGV

Analisi comparativa delle prestazioni relative al running di tre differenti applicazioni scientifiche su singolo pc, su cluster e su infrastruttura GRID

85



Direttore

Enzo Boschi

Editorial Board

Raffaele Azzaro (CT)

Sara Barsotti (PI)

Mario Castellano (NA)

Viviana Castelli (BO)

Anna Grazia Chiodetti (AC)

Rosa Anna Corsaro (CT)

Luigi Cucci (RM1)

Mauro Di Vito (NA)

Marcello Liotta (PA)

Lucia Margheriti (CNT)

Simona Masina (BO)

Nicola Pagliuca (RM1)

Salvatore Stramondo (CNT)

Andrea Tertulliani - coordinatore (RM1)

Aldo Winkler (RM2)

Gaetano Zonno (MI)

Segreteria di Redazione

Francesca Di Stefano - coordinatore

Tel. +39 06 51860068

Fax +39 06 36915617

Rossella Celi

Tel. +39 06 51860055

Fax +39 06 36915617

redazionecen@ingv.it



Rapporti tecnici INGV

ANALISI COMPARATIVA DELLE PRESTAZIONI RELATIVE AL RUNNING DI TRE DIFFERENTI APPLICAZIONI SCIENTIFICHE SU SINGOLO PC, SU CLUSTER E SU INFRASTRUTTURA GRID

Silvia Cariolo², Antonio Drago², Sebastiano Lizzio², Danilo Reitano¹, Gaetano Russo², Fabrizio Pistagna²

¹INGV (Istituto Nazionale di Geofisica e Vulcanologia, Sezione di Catania)

²Consorzio COMETA, Catania

85

Indice

1. Introduzione	pag. 5
2. Analisi comparativa delle prestazioni del modello TEPHRA su infrastruttura GRID e su Cluster Beowulf	
2.1 Il modello TEPHRA	pag. 6
2.2 Modalità di testing	pag. 6
2.3 Code utilizzate nei test di sottomissione su GRID	pag. 6
2.4 Calcolo dei tempi	pag. 7
2.5 Risultati	pag. 9
3. Analisi comparativa delle prestazioni dell'applicazione PRESA su pc e su infrastruttura GRID	
3.1 L'applicazione PRESA	pag. 12
3.2 Modalità di testing	pag. 12
3.3 Risultati	pag. 12
4. "Selettore di features basato su Algoritmi Genetici distribuiti", su pc e su infrastruttura GRID	
4.1 Algoritmi Genetici Distribuiti	pag. 15
4.2 Modalità di testing	pag. 16
4.3 Risultati	pag. 17
5. Conclusioni	pag. 19
6. Sitografia	pag. 20
7. Bibliografia	pag. 20

1. Introduzione

L'Istituto Nazionale di Geofisica e Vulcanologia (INGV) - Sezione di Catania è dal 2005 membro del consorzio COMETA (COnsorzio Multi Ente per la promozione e l'adozione di Tecnologie di calcolo Avanzato), insieme all'Istituto Nazionale di Fisica Nucleare (INFN), all'Istituto Nazionale di Astrofisica (INAF), al Consorzio SCIRE e alle tre università statali siciliane (Catania, Messina, Palermo), e partecipa al progetto PON 2000–2006 denominato *PI2S2*, "Progetto per l'Implementazione e lo Sviluppo di una e-Infrastruttura in Sicilia basata sul paradigma della GRID". Con tale progetto il consorzio COMETA si è proposto di dotare la Sicilia di una grande infrastruttura distribuita di calcolo e di storage al servizio della comunità scientifica e delle piccole e medie imprese. Il paradigma di calcolo distribuito utilizzato è quello del GRID Computing: una "griglia" computazionale delocalizzata di calcolatori interconnessi attraverso una rete (per lo più fornita dal GARR) in grado di condividere le proprie risorse, CPU e memoria di storage, e fornirle potenzialmente a qualunque utente ad essa connesso. I *nodi computazionali* della griglia implementata per il progetto PI2S2 sono dislocati geograficamente tra i tre poli di Catania (Laboratori INAF, INFN e Università), Palermo (Polo per il Supercalcolo Scientifico) e Messina (Facoltà di Ingegneria dell'Università di Messina). La GRID siciliana è poi collegata anche al circuito della GRID europea, con cui condivide peraltro il middleware, ovvero lo strato di software che gestisce tutti i servizi di base e che si interpone tra l'utente finale e il software di più basso livello. Il middleware scelto per questa infrastruttura è "GLite", su piattaforma Unix-Linux, divenuto ormai uno standard di riferimento per tutte le attività di calcolo distribuito a livello scientifico. Questa struttura, di elevato livello tecnologico, si propone come un vero e proprio Laboratorio Virtuale capace di fornire i propri servizi "on demand" alle Università e agli Enti di Ricerca coinvolti, nonché alle PMI (Piccole e Medie Imprese) che ne fanno richiesta.

Nell'ambito del progetto PI2S2 è stato condotto all'interno dell'INGV, uno studio sui vantaggi, in termini di performance, derivanti dall'esecuzione di applicazioni scientifiche su infrastruttura GRID piuttosto che su singolo pc o su un cluster. In particolare nel presente rapporto vengono esposti i risultati di un'analisi comparativa delle prestazioni su singolo pc, sul cluster locale e su infrastruttura GRID, di tre applicazioni scientifiche utilizzate per il monitoraggio delle aree sismo-vulcaniche siciliane.

Le applicazioni analizzate sono riportate di seguito:

Nome applicazione	Comparazione effettuata
TEPHRA	Esecuzione su cluster e su GRID
PRESA	Esecuzione su singolo PC e su GRID
Selettore di Features	Esecuzione su singolo PC e su GRID

2. Analisi comparativa delle prestazioni del modello TEPHRA su infrastruttura GRID e su Cluster Beowulf

2.1 Il modello TEPHRA

TEPHRA è un'applicazione che implementa un modello analitico (Bonadonna et al., 2005), scritta in linguaggio ANSI C con l'ausilio delle specifiche MPI (Message Passing Interface ver.1) per il calcolo parallelo, utilizzata presso l'Unità di Progetto Nubi Vulcaniche (UPNV) i cui risultati sono disponibili presso la Sala Operativa dell'INGV di Catania, per la simulazione della dispersione delle ceneri in caso di esplosioni vulcaniche dell'Etna (Scollo et al., 2007; Andronico et al., 2008). L'output ottenuto dalla sua esecuzione serve per la generazione della cosiddette *mappe di dispersione* che descrivono il fenomeno della deposizione delle ceneri vulcaniche sulla geografia del suolo interessato. L'importanza di tale software è legata alla necessità di trasferire giornalmente in tempi brevi gli output ottenuti dalle simulazioni (per la precisione, le mappe di dispersione da essi ricavate) al Dipartimento di Protezione Civile nazionale che, in casi di emergenza, si preoccuperà di adottare le dovute misure di sicurezza per ridurre i rischi per la popolazione, specie per quanto riguarda l'emissione di bollettini di criticità da fornire al traffico aereo del vicino aeroporto di Fontanarossa a Catania.

2.2 Modalità di testing

I test sono stati condotti utilizzando come discriminante il tempo d'esecuzione. In questo modo sono stati effettuati dei test comparativi che hanno misurato la velocità d'esecuzione dell'applicazione TEPHRA sia sull'infrastruttura GRID di PI2S2 sia sul cluster locale.

Il cluster utilizzato, presente presso la sala CED della sede INGV di Catania, è di tipo Beowulf, composto da 24 server Proliant DL 145 OPTERON 248. Ciascuna di queste macchine possiede due processori da 2.6 GHz/1Mb, 1 Gb di memoria RAM ed un HDD da 80 Gb. Il sistema operativo utilizzato è SUSE Linux Enterprise Server.

I siti di calcolo dell'infrastruttura GRID di PI2S2 sono invece delle architetture IBM BladeCenter H costituite da 14 IBM Blade LS21. Ogni Blade è equipaggiato con 2 AMD Opteron 2218 dual-core, frequenza di clock 2.6 GHz. Ogni core ha 2GB di RAM DDR2 a 667 MHz (8 GB in totale per Blade). L'hard disk interno è da 73GB 10Krpm. Network Gigabit Ethernet sono usate per le connessioni verso l'esterno e CISCO Topspin InfiniBand-4X network per le funzionalità HPC. Il sistema operativo utilizzato è Scientific Linux (SLC 4), una distribuzione derivata da Red Hat Enterprise Linux 4.

I test sono stati condotti effettuando prove di sottomissione di job, in cui varia il numero di nodi coinvolti nel calcolo, secondo una sequenza di 4,8,16,20,24 e 28 nodi. Per ottenere dei dati statisticamente validi, sono state effettuate almeno 4 sottomissioni per ciascuna della classi precedenti. Da tali prove sono stati ricavati i tempi di esecuzione e di attesa (nel caso della GRID), successivamente mediati per essere confrontati tra loro.

2.3 Code utilizzate nei test di sottomissione su GRID

Il processo di sottomissione di un job sull'infrastruttura GRID è strettamente legato ad un *delay* temporale dovuto alla logica delle code di attesa, i cui tempi dipendono dal numero di job già in esecuzione o in attesa di eseguire su un determinato nodo computazionale della GRID e soprattutto dalle risorse di calcolo da essi richieste. Considerata la particolarità di tale applicazione e l'esigenza di tempestività con cui i tempi di risposta devono essere comunicati alla Protezione Civile, si è ricorsi all'utilizzo di code dedicate ad alta priorità. La progettazione di queste ultime, denominate *code Emergency*, è stata oggetto di collaborazione tra il gruppo COMETA operante presso l'INGV ed il personale del Consorzio presente all'INFN di Catania. La peculiarità delle code *Emergency* è quella di bloccare temporaneamente tutti i job in esecuzione su un determinato sito della GRID per permettere ad un dato job, nella fattispecie TEPHRA, di riservarsi con una maggiore priorità le risorse necessarie e andare subito in esecuzione.

2.4 Calcolo dei tempi

Per ogni numero di nodi fissato ed utilizzando gli stessi parametri specifici dell'applicazione e gli stessi dati d'ingresso, sono state effettuate dieci prove. Da ognuna di esse, sono stati presi in esame i risultati delle quattro prove con valori del tempo di attesa più simili, in modo da assicurare che le condizioni di funzionamento della GRID siano pressoché identiche nella prove considerate. I valori dei risultati sono stati ordinati in tabelle, delle quali si riporta un esempio, divise per piattaforma utilizzata (GRID o cluster), per numero di nodi e per tipo di coda utilizzata (nel caso di sottomissione su GRID). A partire da tali tabelle sono stati prodotti i grafici di comparazione del successivo paragrafo. Per ogni tabella vengono illustrati: numero di nodi utilizzati, orario di sottomissione, orario d'inizio dell'esecuzione, orario di fine dell'esecuzione, tempo di esecuzione, tempo di sottomissione e tempo di attesa (quest'ultimo solo nel caso di GRID). I tempi di esecuzione sono stati ottenuti mediante una differenza lineare tra il tempo registrato al termine dell'esecuzione del job ed il tempo iniziale registrato allo start-up dell'esecuzione:

$$T = T_{\text{finale}} - T_{\text{iniziale}}$$

dove il tempo finale ed iniziale sono stati registrati tramite l'utilizzo di uno script in linguaggio Bash che viene invocato sul nodo master rispettivamente all'inizio ed alla fine della sottomissione di job MPI su infrastruttura GRID (*mpi_pre.sh*, *mpi_post.sh*).

```
Type = "Job";
JobType = "MVAPICH_gcc4";
Executable = "new_tephra";
NodeNumber = 20;
Arguments="new_tephra.conf eruption_strong.in etnapoints.in
wind00.in";
StdOutput = "output.out";
StdError = "error.out";
InputSandbox={"mpi.pre.sh","mpi.post.sh","new_tephra",
wind00.in","eruption_strong.in","etnapoints.in","new_tephra.conf"
};
OutputSandbox = {"output.out", "error.out"};
RetryCount = 1;
```

Tabella 1 Script di sottomissione JDL (Job Description Language) per TEPHRA.

Quest'ultimo è uno dei file utilizzati per la sottomissione del job TEPHRA su infrastruttura GRID, tramite i comandi appartenenti alla suite del sistema *glite-wms* (workload management system) per la gestione dei job.

NODI	SUBMITTED	START SIMULATION	END SIMULATION	TEMPO ESEC.	TEMPO SOTT.	TEMPO ATTES.
8	13.03.38	13.05.39	13.08.05	<i>0.02.26</i>	<i>0.04.27</i>	<i>0.02.01</i>
8	13.03.48	13.05.39	13.08.20	<i>0.02.41</i>	<i>0.04.32</i>	<i>0.01.51</i>
8	13.04.01	13.05.44	13.08.19	<i>0.02.35</i>	<i>0.04.18</i>	<i>0.01.43</i>
8	13.04.10	13.05.39	13.07.59	<i>0.02.20</i>	<i>0.03.49</i>	<i>0.01.29</i>

Tabella 2 Tempi ottenuti dai test di sottomissione di TEPHRA su GRID su 8 nodi con code normali.

NODI	SUBMITTED	START SIMULATION	END SIMULATION	TEMPO ESEC.	TEMPO SOTT.	TEMPO ATTES.
8	12.52.43	12.54.21	12.56.32	<i>0.02.11</i>	<i>0.03.49</i>	<i>0.01.38</i>
8	12.52.52	12.54.34	12.56.49	<i>0.02.15</i>	<i>0.03.57</i>	<i>0.01.42</i>
8	12.52.57	12.54.57	12.57.01	<i>0.02.04</i>	<i>0.04.04</i>	<i>0.02.00</i>
8	12.53.01	12.55.07	12.57.38	<i>0.02.31</i>	<i>0.04.37</i>	<i>0.02.06</i>

Tabella 3 Tempi ottenuti dai test di sottomissione di TEPHRA su GRID su 8 nodi con code Emergency.

2.5 Risultati

Si riportano, di seguito, alcuni grafici che sintetizzano i risultati ottenuti.

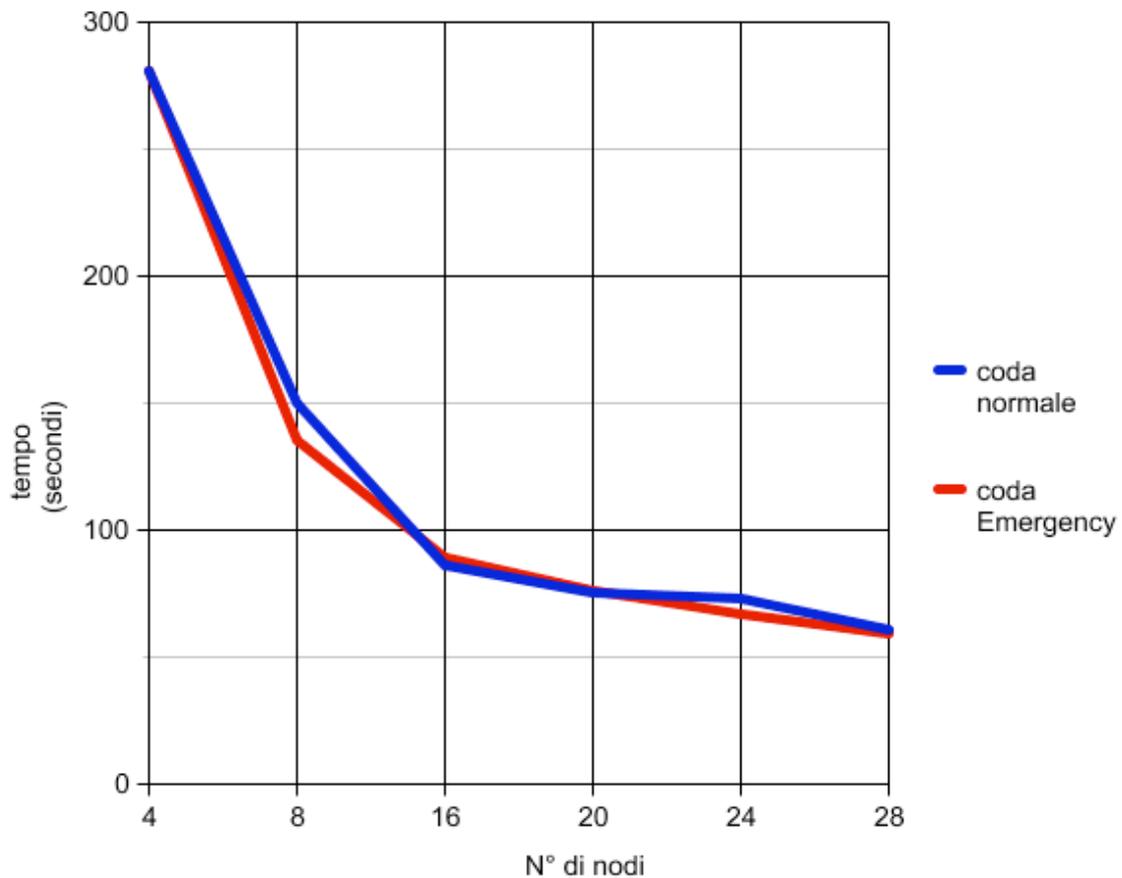


Figura 1 Grafico dei tempi medi di esecuzione ottenuti da una simulazione di TEPHRA su GRID, al variare del numero di nodi. Le sottomissioni sono state effettuate sia su code normali (in blu) sia su code Emergency (in rosso) sebbene, come era prevedibile, tale scelta non influenzi i tempi di esecuzione che risultano pressoché uguali.

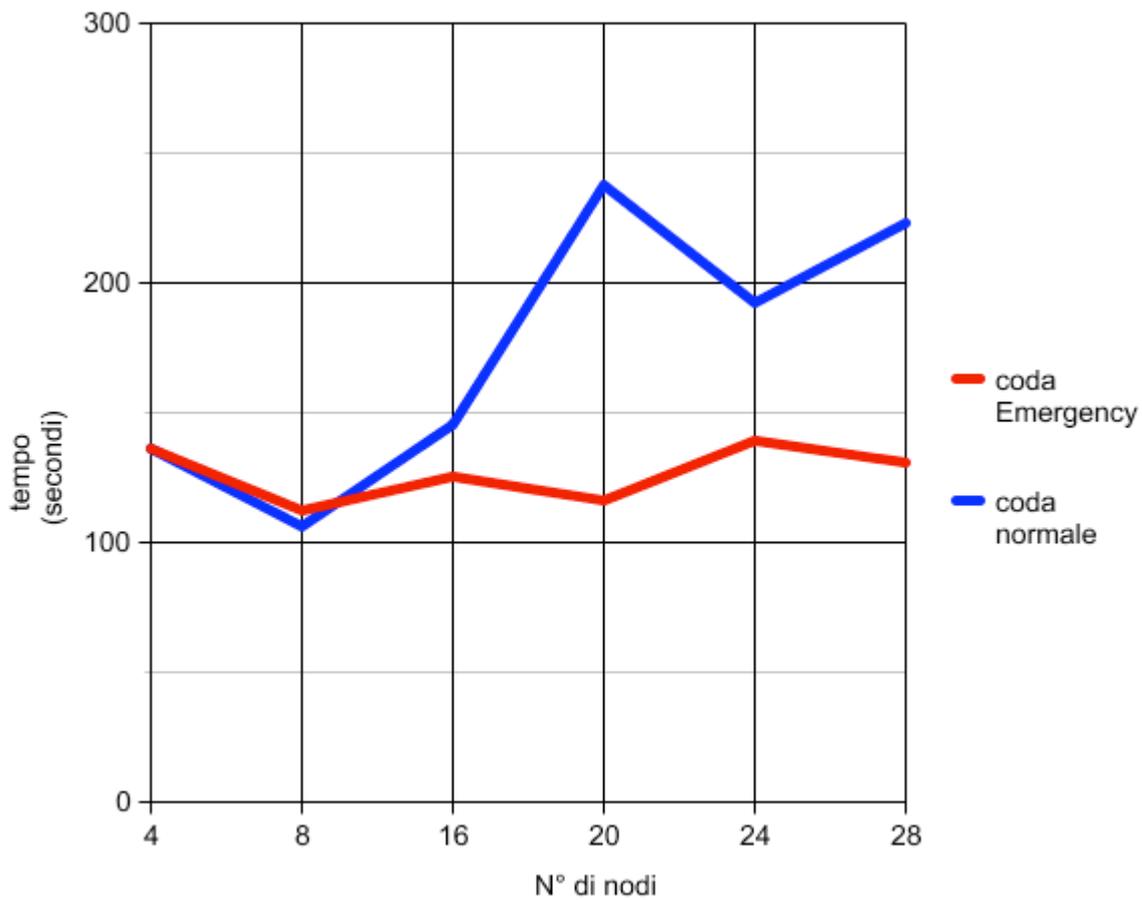


Figura 2 Grafico dei tempi di attesa di un job di TEPHRA su GRID al variare del numero di nodi. Questa figura evidenzia i miglioramenti apportati dall'utilizzo della coda Emergency (in rosso) nell'abbattimento dei tempi di attesa, rispetto alle normali code (in blu).

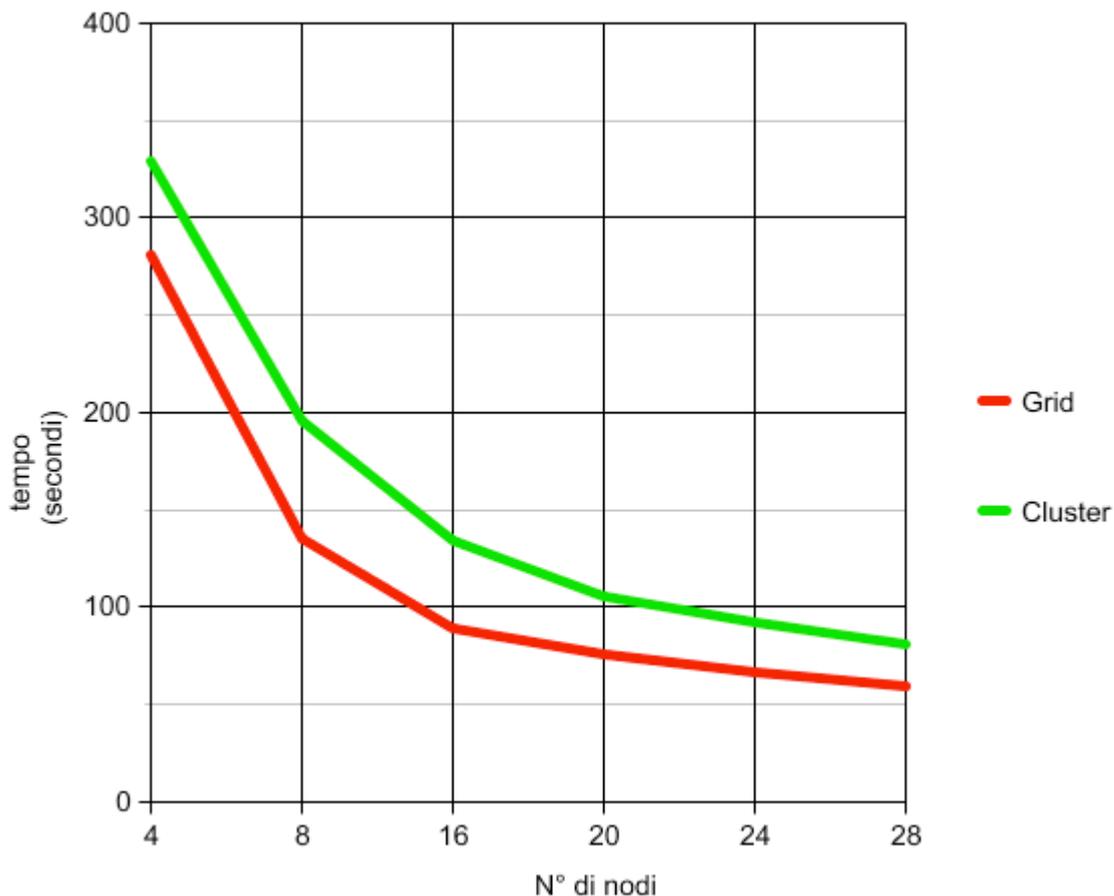


Figura 3 Grafico dei tempi medi di esecuzione di una simulazione di TEPHRA su GRID su coda Emergency (in rosso) e sul cluster locale (in blu) al variare del numero di nodi utilizzati. Si nota come i tempi di esecuzione siano maggiori sul cluster, rispetto a quelli ottenuti su GRID.

I job delle simulazioni di TEPHRA, sottomessi su coda *Emergency*, hanno garantito performance competitive, in termini di tempi d'esecuzione, rispetto ai test eseguiti su cluster locale, e in più non risentono dei tempi di attesa a cui normalmente si va incontro sottomettendo un job sull'infrastrutture GRID. Tali code riservate, dapprima sperimentate su un unico sito GRID, attualmente sono in fase d'implementazione sull'intera infrastruttura. A breve saranno effettuati dei test per validare un utilizzo *massimo* della intera infrastruttura GRID in caso di gravi eventi calamitosi.

3. Analisi comparativa delle prestazioni dell'applicazione PRESA su pc e su infrastruttura GRID

3.1 L'applicazione PRESA

La Sezione di Catania dell'INGV, dal gennaio 2001 ad oggi, gestisce una rete di tredici stazioni di monitoraggio video, costituita da telecamere fisse, sensori operanti nel campo del visibile, dell'infrarosso e del termico con lo scopo di monitorare l'attività dei principali vulcani attivi siciliani, con particolare riferimento ai vulcani Etna e Stromboli e all'attività fumarolica dell'isola di Vulcano. I segnali video, registrati in continuo con appositi apparati di videoregistrazione, sono inviati in tempo reale alla Sala Operativa della Sezione di Catania dove vengono archiviati su supporti ottici e visualizzati su monitor in tempo reale.

L'obiettivo dell'applicazione denominata PRESA (**P**attern **R**ecognition to monitor **E**xplosive **S**trombolian **A**ctivity), è in breve il triggering e la successiva classificazione delle esplosioni più significative registrate sul vulcano Stromboli utilizzando, quali dati di input, i frame ripresi dalle telecamere operanti nello spettro infrarosso posizionate sul suddetto vulcano. Ciò, sia per fini di monitoraggio e sorveglianza, sia per analisi statistiche dei dati archiviati, al fine di migliorare la comprensione dei fenomeni in atto durante un evento dell'attività vulcanica stromboliana.

3.2 Modalità di testing

Diversi *running* dell'applicazione PRESA sono stati effettuati sia su GRID, sia in locale su pc ove l'applicazione test è stata sviluppata. In particolare, per quanto riguarda l'esecuzione dell'applicazione su GRID, sono state effettuate delle simulazioni in tre diverse modalità di sottomissione, adattando di volta in volta il codice:

- job singolo (sequenziale)
- collezione di job
- job DAG (Direct Aciclyc Graph)

È stata innanzitutto testata la sottomissione dell'applicativo come job singolo, in maniera sequenziale, senza sfruttare il parallelismo. Successivamente è stata testata una sottomissione a "collezione" di job, dividendo il codice in tre moduli indipendenti l'uno dall'altro, lanciati sulla GRID simultaneamente su job diversi. Infine è stata sperimentata una sottomissione di tipo DAG (Direct Aciclyc Graph), un grafo direzionale privo di cicli in cui ogni vertice è costituito da un job riguardante un determinato task dell'applicazione. Il codice è stato suddiviso su più moduli, secondo una precisa gerarchia di dipendenze input-output, e quindi distribuito su job diversi lanciati simultaneamente e in grado di scambiarsi dati fra loro. Nel caso specifico l'algoritmo di processamento delle immagini è stato suddiviso in quattro parti, tre delle quali possono eseguire in parallelo ed un quarto job finale che invece attende che i primi abbiano terminato prima di poter eseguire, a partire dai loro output, la sua parte di codice.

Queste ultime due modalità di sottomissione (collection e DAG) permettono di eseguire, senza troppe modifiche al codice, un'applicazione sequenziale in maniera pseudo-parallela.

3.3 Risultati

Per quanto riguarda i tempi di attesa, ovvero il tempo che intercorre tra il momento in cui il job viene sottomesso su GRID e l'istante in cui il job passa allo stato di esecuzione, si è evinto dai test effettuati che il job singolo è mediamente quello che attende per minor tempo rispetto alla collezione e rispetto a quello DAG. Tale risultato è facilmente spiegabile: il job singolo deve attendere su una sola coda; il job collection, invece, essendo costituito da tre job, ha una probabilità maggiore che uno (o più) di questi "atterri" su un nodo di computazione con una coda lunga. Il tempo medio di attesa risulta dunque essere penalizzato da questi casi meno fortunati in cui un job attende per un tempo più lungo rispetto agli altri. Diverso è il

comportamento nel caso del DAG, per il quale va considerato un tempo aggiuntivo di attesa dovuto al fatto che l'inizio della coda del job finale è sicuramente subordinata all'ultima delle code dei tre predecessori.

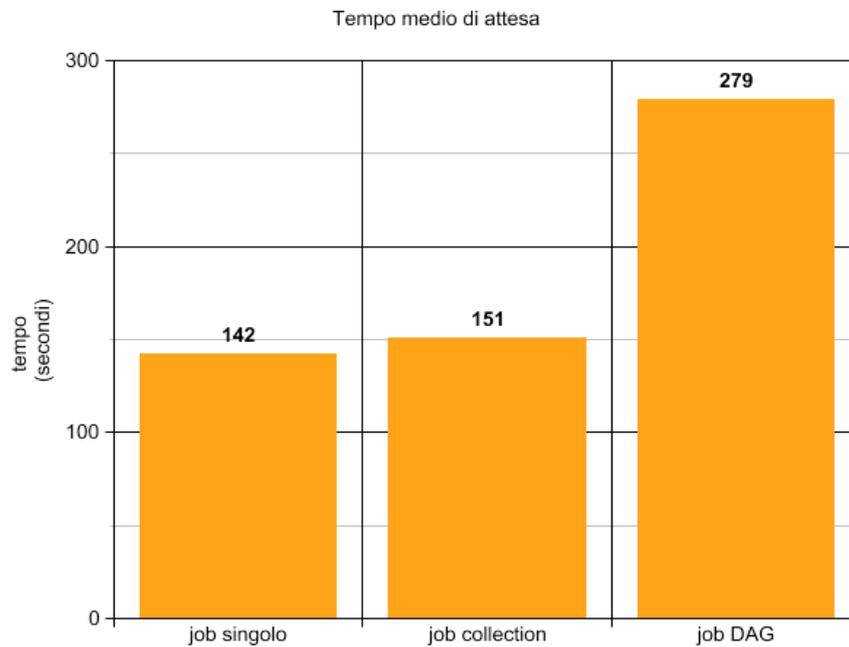


Figura 4 Tempi medi di attesa dei job PRESA su GRID, nei tre distinti casi sopraindicati.

Dal grafico in figura 4 si può notare in particolare come il tempo medio di attesa del job collection sia solo lievemente superiore a quello del job singolo.

Passando all'analisi dei tempi di esecuzione è possibile notare come questi siano positivamente influenzati dalla possibilità dei job di eseguire in parallelo su nodi differenti. Per questo motivo i tempi medi di esecuzione sono nettamente più bassi sia per quanto riguarda il job collection sia per il job DAG rispetto al job singolo.

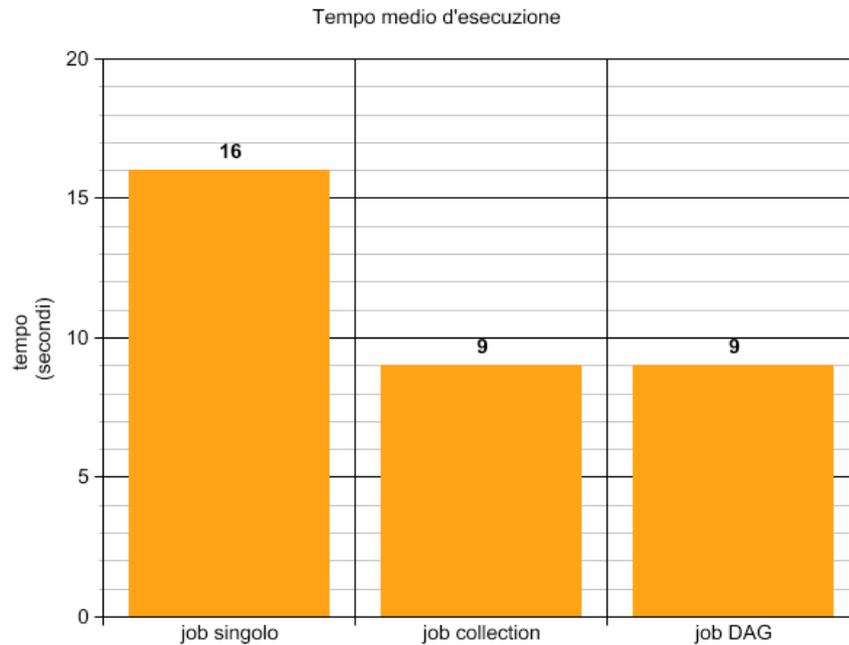


Figura 5 Grafico dei tempi medi di esecuzione del job di PRESA su GRID.

Considerando in generale i tempi di esecuzione del job collection si può notare come, per ogni simulazione essi siano, a meno di casi eccezionali, proporzionalmente inferiori rispetto a quello fatto registrare dal job singolo. Dal grafico si può notare come i tempi medi di esecuzione del job collection e del job DAG siano quasi dimezzati rispetto a quelli del job singolo. Inoltre il job DAG, in quanto a tempo di esecuzione medio, non introduce alcun ritardo rispetto al job collection.

Per quanto riguarda il tempo di elaborazione totale, definito come il tempo che intercorre tra l'istante di sottomissione del job e l'istante in cui termina l'esecuzione dell'ultimo job, è evidente come il job singolo ottenga i tempi più bassi sia rispetto al job collection sia, ovviamente, al job DAG.

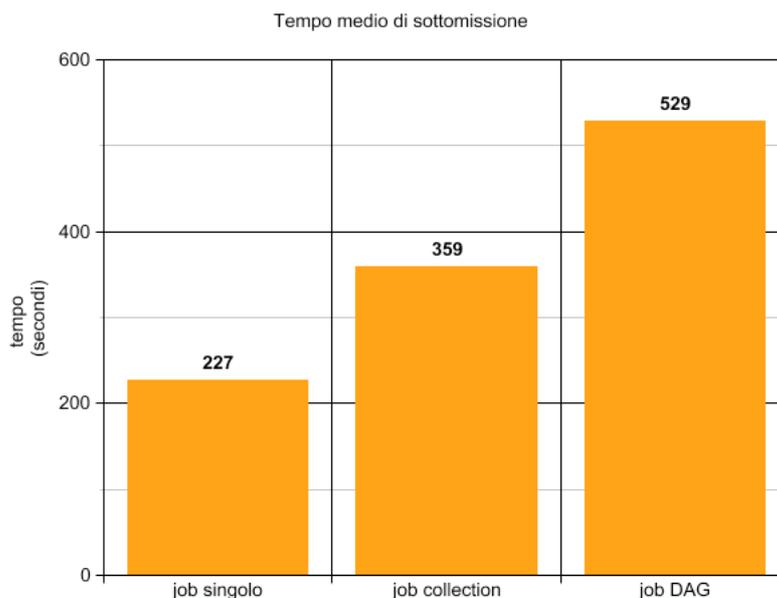


Figura 6 Grafico dei tempi medi di sottomissione dei job PRESA su GRID.

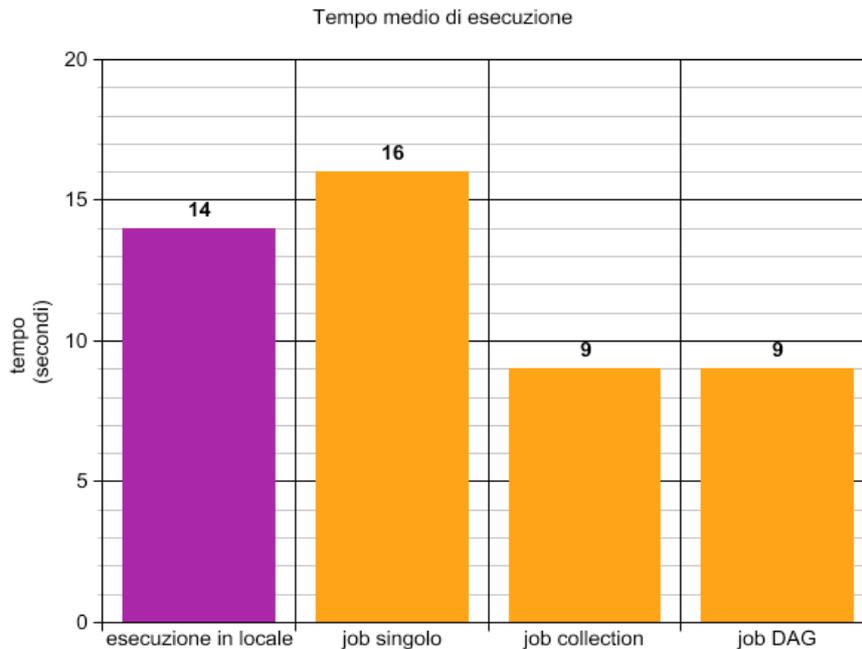


Figura 7 Grafico del tempo medio di esecuzione del job PRESA su GRID (nelle tre diverse tipologie) e su singolo pc.

Il grafico in figura 7 mostra chiaramente come, considerando anche solamente il tempo di esecuzione, senza considerare, quindi, i tempi di attesa, l'esecuzione su macchina locale sia addirittura più breve di quella del job singolo. Ciò perché il codice dell'applicazione resta comunque sequenziale e quindi non riesce a sfruttare le possibilità di parallelismo messe a disposizione dall'elevato numero di core presenti su GRID. Piuttosto in questo caso, su GRID, vi è un ulteriore ritardo introdotto dal download dei dati in input dallo storage element in cui essi sono salvati, mentre in locale tali dati si trovano sulla stessa macchina in cui viene eseguito il software.

4. “Selettore di *features* basato su Algoritmi Genetici distribuiti”, su pc e su infrastruttura GRID

4.1 Algoritmi Genetici Distribuiti

Nell'ambito dello studio relativo all'analisi dei segnali infrasonici (particolari segnali sismici legati all'attività esplosiva dei vulcani e registrati da particolari sensori microfonici), è stato implementato un nuovo software basato sull'utilizzo di sistemi esperti. Esso è in grado di analizzare, in modo autonomo, i segnali in questione e di fornirne precise informazioni. In particolare il sistema, diviso in più moduli, opera i seguenti passi:

- a) un modulo effettua un riconoscimento ed una classificazione degli eventi in famiglie;
- b) un modulo chiamato "*estrattore di features*", estrae da ogni evento, delle particolari misure a partire dalle forme d'onda;
- c) un ultimo modulo si occupa di effettuare tra una moltitudine di possibili features estratte, quelle che meglio caratterizzano la classificazione di cui sopra.

Per effettuare la selezione di cui al punto precedente è stata utilizzata la teoria degli algoritmi genetici (GA), adattata al caso particolare di “features selection”. In generale, il problema della riduzione o

selezione delle features è applicato quando si vogliono ottenere dei benefici nella manipolazione dei dati che si hanno a disposizione. Tali benefici riguardano, per esempio, una più semplice elaborazione dei dati, una riduzione dello storage necessario per la memorizzazione degli stessi, la riduzione dei tempi di calcolo ed un miglioramento nei risultati ottenuti. Nel dettaglio l'individuazione delle features ottime, ha i seguenti obiettivi:

- individuare quelle a maggiore contenuto informativo, ovvero quelle che determinano una migliore separazione delle classi analizzate;
- eliminare quelle che risultano ridondanti e irrilevanti, a vantaggio dei tempi di calcolo.

In tale ambito effettuare un benchmark sui tempi di esecuzione e scheduling di algoritmi genetici, eseguiti sia su singolo pc che su infrastruttura GRID rappresenta un passo fondamentale nella valutazione delle caratteristiche proprie e delle performance dell'algoritmo proposto, poiché in letteratura non ci sono, al momento, evidenze sull'utilizzo dei GA per tali applicazioni.

4.2 Modalità di testing

Ogni job, su GRID, esegue una simulazione formata da un numero N_p di prove, ognuna delle quali esegue l'algoritmo genetico in maniera distribuita, in quanto, grazie all'utilizzo delle librerie MPI per il calcolo parallelo nell'implementazione dell'algoritmo ogni nodo della GRID processa una singola classe di eventi.

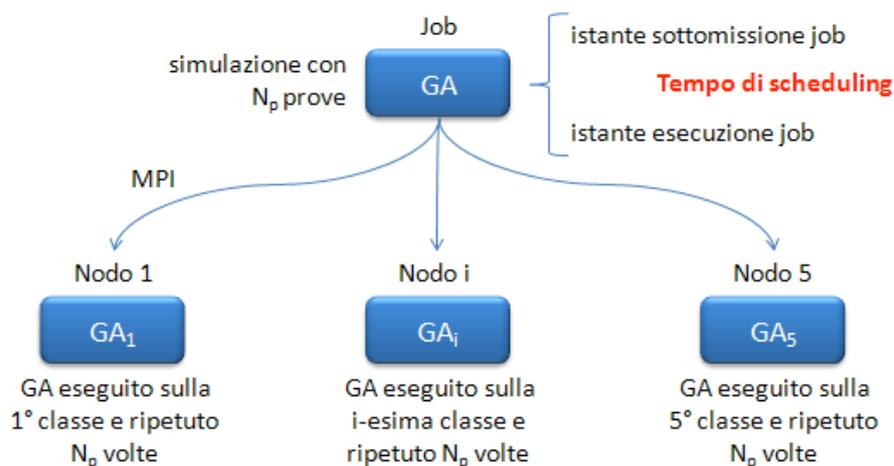


Figura 8 Schema di esecuzione di un job sulla GRID.

Sono state effettuate diverse simulazioni, formate da un numero diverso di prove ed eseguite da ogni job lanciato sulla GRID. Ogni simulazione, per ogni classe, produce informazioni riguardanti il tempo di esecuzione, gli andamenti della distanza ottima, la fitness ottima e il numero delle features ottime.

Anche in questo caso occorre ricordare che, tenuto conto dei tempi di attesa, vale la formula:

$$\text{tempo esecuzione}_{\text{vero}} = \text{tempo di esecuzione}_{\text{GA}} + \text{tempo di schedulazione}_{\text{JOB}}$$

4.3 Risultati

I grafici illustrano i risultati ottenuti analizzando i dati delle singole classi di eventi. Nonostante il numero di simulazioni sia lo stesso, il tempo di calcolo risulta differente per le varie istanze, poiché varia il numero di elementi che compongono la classe stessa.

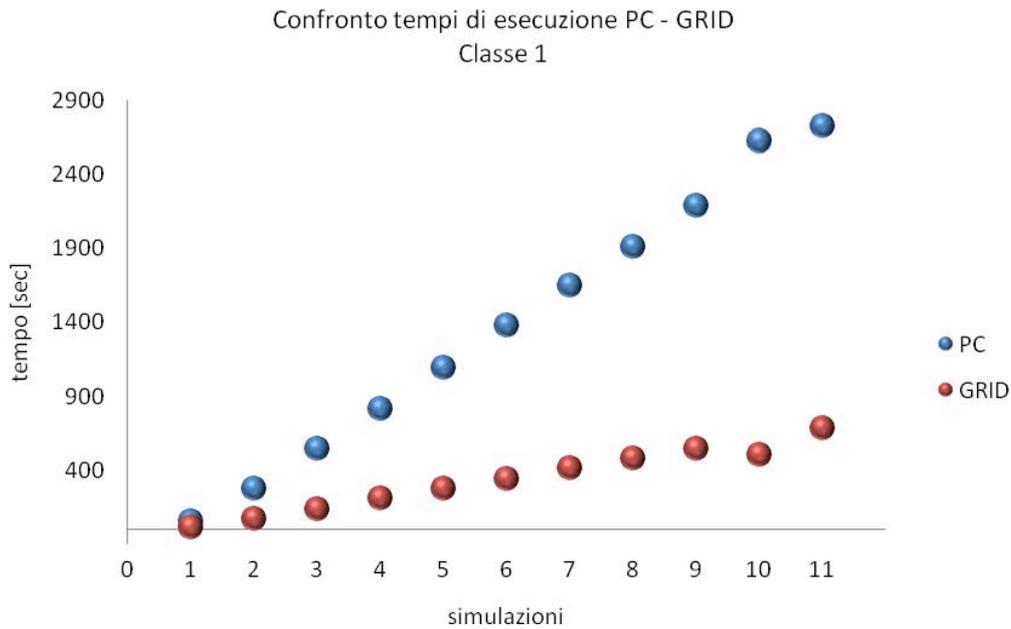


Figura 9 Confronto dei tempi di esecuzione su PC e su GRID per la Classe 1.

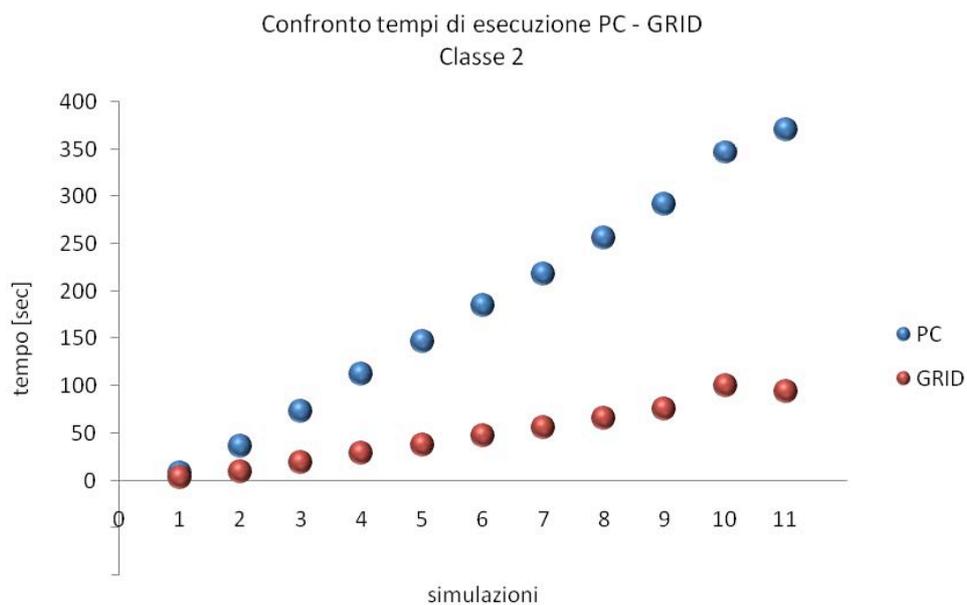


Figura 10 Confronto tempi di esecuzione tra PC e GRID per la Classe 2.

Come si può notare, rispetto a quelli relativi all'esecuzione su singolo PC, i tempi di esecuzione del GA sulla GRID sono considerevolmente minori. Tale riduzione diventa più accentuata con simulazioni più pesanti, formate cioè da un numero di prove maggiori. Si evidenzia quindi che, utilizzando la GRID, risulta possibile ottenere un notevole miglioramento nei tempi di esecuzione dell'algoritmo.

Si introducono ora, sui dati precedenti, anche i tempi di attesa per la schedulazione del job.

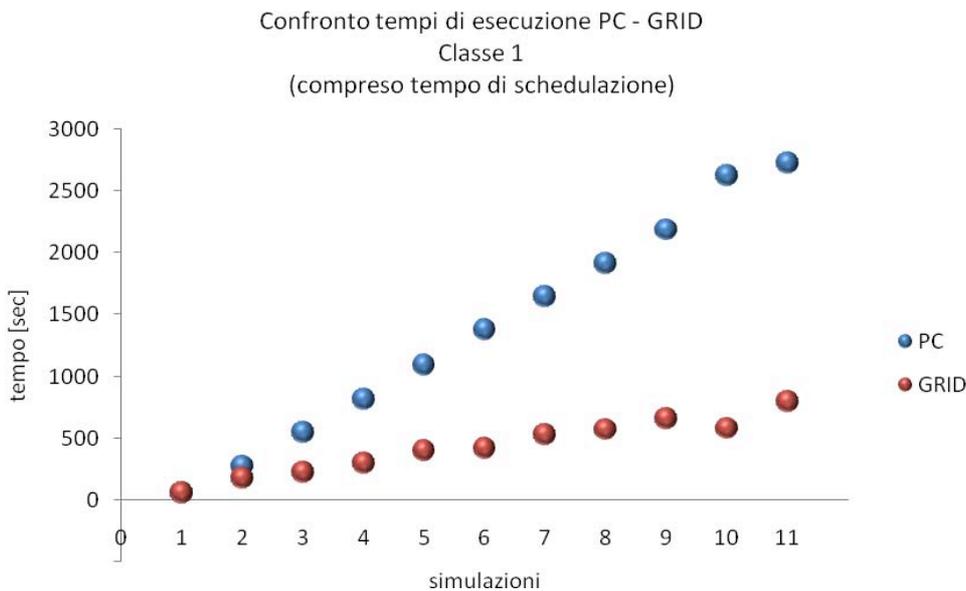


Figura 11 Confronto dei tempi di esecuzione su PC e su GRID per la Classe 1 comprensivi dei tempi di schedulazione.

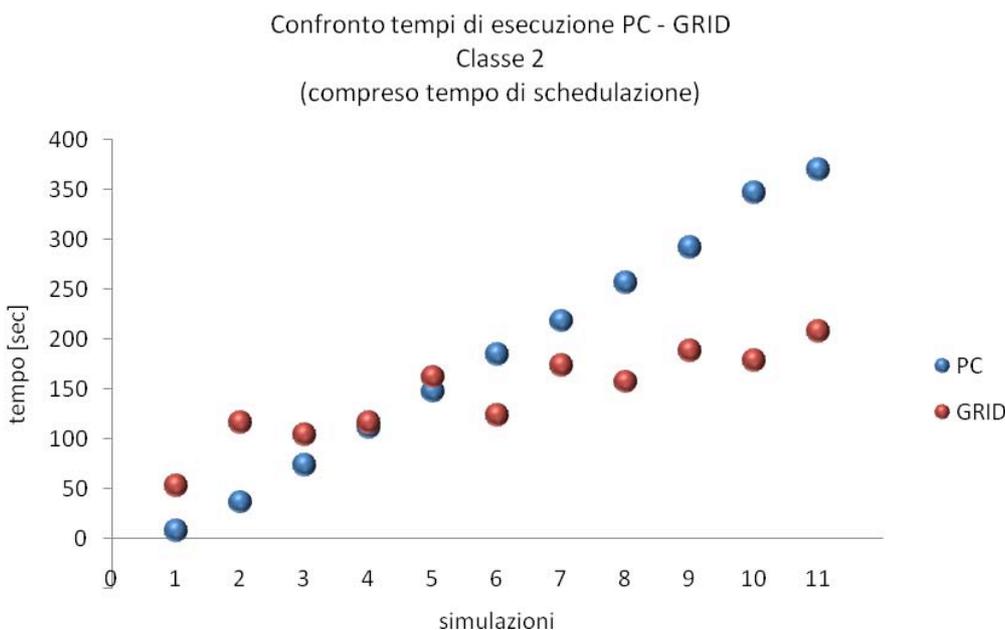


Figura 12 Analogo confronto della figura 11 per la Classe 2.

Come si può notare, per certe simulazioni i tempi di esecuzione sulla GRID risultano maggiori rispetto ai tempi di esecuzione su singolo PC. Ciò dipende proprio dal tempo di attesa in coda per la schedulazione del job che in certi casi può risultare molto lungo ed influire pesantemente sul tempo di esecuzione complessivo dell'algoritmo. Tuttavia, se l'algoritmo richiede un carico computazionale piuttosto elevato, si ha comunque un miglioramento sui tempi di esecuzione della GRID.

5. Conclusioni

In questo lavoro sono state valutate le prestazioni di tre differenti algoritmi su singolo pc, su un cluster locale e sulle macchine della GRID del progetto PI2S2. Nonostante ci si riferisca a tre software di analisi che hanno peculiarità del tutto differenti, è possibile individuare degli elementi di carattere generale quando ci si avvicina al mondo del calcolo computazionale avanzato. In prima istanza è bene ricordare che il software può essere nativamente sviluppato per ambiente di griglia o reso parallelo con artifici computazionali. Infatti, nei casi trattati, la scelta di software che accettano specifiche MPI è da preferire per le prestazioni ottenibili, anche se risulta opportuno valutare sempre la possibilità di parallelizzare software funzionante in modalità sequenziale. E' infatti possibile notare che all'aumentare del numero di nodi e di esecuzioni contemporanee (metodo del *job farming*) si possano comunque trarre vantaggi dall'utilizzo del calcolo avanzato anche per algoritmi sequenziali. L'aspetto evidenziato in tutti i casi rimane la parte di tempo di attesa dovuto alla schedulazione delle code dei singoli job. Il tempo di schedulazione medio rimane comunque un offset da sommare al tempo di esecuzione vero e proprio. In taluni casi l'attesa viene praticamente compensata dalla maggiore velocità di esecuzione dei job all'interno della GRID. Nei casi in cui risulti prioritario fornire informazioni in tempi ristretti, non solo per scopi di ricerca ma, ad esempio, nei casi di valutazione dei rischi connessi alla sorveglianza vulcanica (come nel caso di creazione di mappe di dispersione delle ceneri vulcaniche o di riconoscimento di eventi eruttivi) la progettazione e la realizzazione della cosiddetta *coda Emergency* rappresenta un ottimo sistema di impiego delle potenzialità della GRID poiché essa limita, in modo considerevole, i tempi di attesa dei job. I risultati incoraggianti sulle prove effettuate hanno suggerito agli autori successivi test sull'intera griglia. Tali test saranno utili anche per una migliore valutazione di importanti fattori quali, ad esempio, la larghezza di banda, la velocità di comunicazione dei nodi e l'accesso a sistemi di storage distribuiti.

6. Sitografia

- [1] <http://www.pi2s2.it>
- [2] <http://www.cas.usf.edu/~cconnor/vg@usf/tephra.html>
- [3] <http://globus.org/>
- [4] <http://www.ct.ingv.it/>
- [5] <https://grid.ct.infn.it/twiki/bin/view/PI2S2/>
- [6] <http://mathworld.wolfram.com/AcyclicDigraph.html>
- [7] <http://www.gridcomputing.com/>

7. Bibliografia

- [1] A.Cannata, P. Montalto, E. Privitera, G. Russo, S. Gresta. "Tracking eruptive phenomena by infrasound: May 13, 2008 eruption at Mt. Etna", submitted to GRL.
- [2] Quadro di sintesi e aggiornamento al 19 Maggio 2008 sullo stato di attività sismica. Rapporto interno a cura della UFS dell'INGV – Catania.
- [3] A. Cannata, P. Montalto, E. Privitera, G. Russo."Characterization and location of infrasonic sources in active volcanoes: Mt. Etna, September-November 2007", submitted to JGR.
- [4] Andrea Cannata, Placido Montalto, Eugenio Privitera, Danilo Reitano, Gaetano Russo."Classificazione di eventi infrasonici sul vulcano Etna" Presentazione orale Grid Open Day alla sezione di Catania dell'INGV, 2 Aprile 2008.
- [5] A.Cannata, P. Montalto, E. Privitera, G. Russo, S. Gresta. "Tracking eruptive phenomena by infrasound: May 13, 2008 eruption at Mt. Etna", Sezione Poster, AGU 2009.
- [6] A. Cannata, P. Montalto, E. Privitera, G. Russo."Characterization and location of infrasonic sources in active volcanoes: Mt. Etna, September-November 2007",Sezione Poster, AGU 2009.
- [7] M. Soryani, N. Rafat: "Application of genetic algorithms to feature subset selection in a farsi OCR", Proceedings of word academy of science, 2006.
- [8] R. C. Gonzalez, R.E. Woods, Digital Image Processing 2nd Edition, Prentice Hall, New Jersey, 2002.
- [9] R.O. Duda, P.E. Hart, D.G. Stork, Pattern classification 2ed., Wiley, 2000.
- [10] M. Migliardi, Grid Computing: da dove viene e che cosa manca perché diventi una realtà?, MondoDigitale n. 2, giugno 2004.
- [11] S. Cariolo, "Pattern Recognition to monitor explosive strombolian activity", Grid Open Day alla sezione di Catania dell'INGV, 2 Aprile 2008.
- [12] C. Jacob, Immagini Digitali per la Comunicazione Visiva e i Beni Culturali, 2004.

- [13] I.T. Young, J.J. Gerbrands, L.J. van Vliet, Image Processing Fundamentals, Delft University of Technology, 1995.
- [14] L. Villari, Una ricerca interdisciplinare per mitigare l'ira dei vulcani, Ricerca & Futuro n.8, Rivista del Consiglio Nazionale delle Ricerche, 1998.
- [15] I. Foster, C. Kesselman, S. Tuecke, The Anatomy of the Grid: Enabling Scalable Virtual Organizations, International Journal of Supercomputer Applications, 2001.
- [16] I. Foster, C. Kesselman, The Grid: Blueprint for a New Computing Infrastructure, 1999.
- [17] A. Moore, K-means and Hierarchical Clustering - Tutorial Slides.
- [18] Jian-Jiun Ding, The class of "Advanced Digital Signal Processing", the Department of Electrical Engineering, National Taiwan University (NTU), Taipei, Taiwan, 2008.
- [19] Weisstein, Eric W., "Acyclic Digraph.", MathWorld--A Wolfram Web Resource.
- [20] L. Djerou, N. Khelil, M. Batouche, Image Segmentation by Self-Organised Region Growing, Computer Information Systems and Industrial Management Applications, 2008.
- [21] C. Bradford Barber, D. P. Dobkin, H. Huhdanpaa, The QuickHull Algorithm for Convex Hulls, ACM Transactions on Mathematical Software, 1996.

Coordinamento editoriale e impaginazione

Centro Editoriale Nazionale | INGV

Progetto grafico e redazionale

Laboratorio Grafica e Immagini | INGV Roma

© 2008 INGV Istituto Nazionale di Geofisica e Vulcanologia

Via di Vigna Murata, 605

00143 Roma

Tel. +39 06518601 Fax +39 065041181

<http://www.ingv.it>



Istituto Nazionale di Geofisica e Vulcanologia