

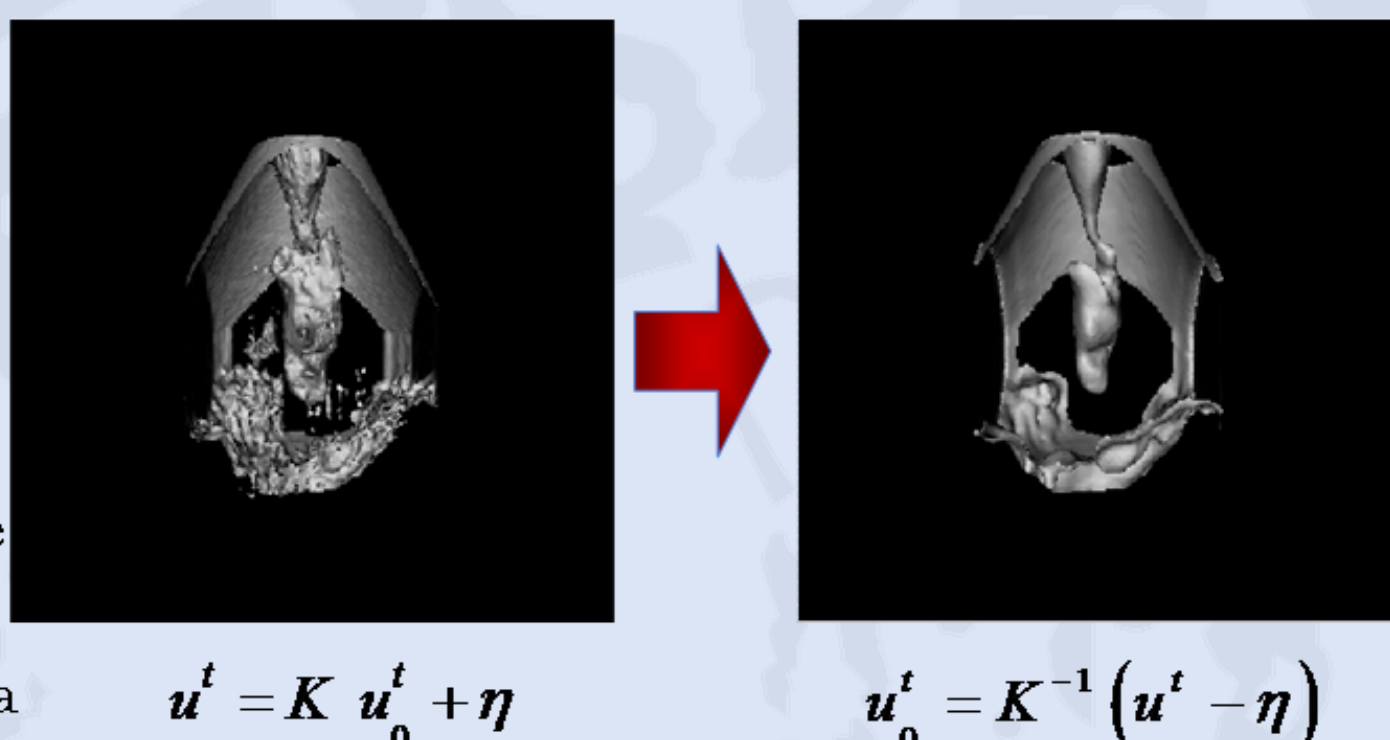
Abstract

Problema: denoising di immagini e sequenze di immagini ecocardiografiche, descritto da equazioni differenziali alle derivate parziali, non lineari, di tipo evolutivo. Un'opportuna discretizzazione di tale modello conduce ad uno schema iterativo, che richiede ad ogni passo la risoluzione di un sistema di equazioni lineari [3]. Ciascun sistema lineare viene risolto mediante metodi iterativi quali, ad esempio, il GMRES preconditionato [2]. Se n è la dimensione dell'immagine, una stima della complessità computazionale dell'applicazione è di ordine $O(n^6)$. La necessità di ottenere il risultato in tempo utile e praticabile nella diagnostica medica, poiché n è almeno dell'ordine di 10^2 , ha portato alla progettazione ed implementazione di software in ambiente di calcolo ad alte prestazioni. L'implementazione dell'algoritmo è in ambiente **PETSc** [1] (Portable, Extensible Toolkit for Scientific Computation) e si basa su una strategia di parallelizzazione che prevede una distribuzione di tipo *slice-block* di ciascun frame della sequenza. Il software parallelo consente di ridurre del 75% (da circa 24 a circa 6 minuti) il tempo effettivo di risoluzione [3].

1 Problema: Denoising di sequenze di immagini 3D

Denoising:
processo di eliminazione del rumore

K : operatore di acquisizione/formazione
 η : rumore
 t : tempo di acquisizione della sequenza



2 Modello Matematico

sequenza di input $u_0(\tau, x_1, x_2, x_3, \vartheta): \mathcal{R} \times \Omega \times I \rightarrow \mathcal{R}^+$
 con $\begin{cases} \Omega \subset \mathcal{R}^3 & \text{Dominio dell'Immagine} \\ I := [0, T] & \text{Intervallo di tempo di acquisizione della sequenza} \\ \tau \in \mathcal{R} & \text{Scala} \end{cases}$

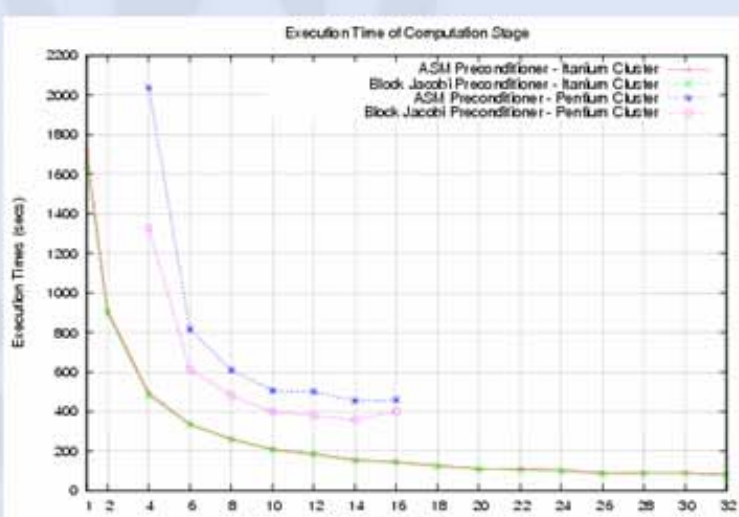
$$\frac{\partial u}{\partial \tau} = \text{clt}(u) \nabla \cdot (g(|\nabla u_r|) \nabla u)$$

 con $g(s) : \begin{cases} g(0) = 1 & \text{ed } u_r \text{ ottenuta dalla convoluzione} \\ \lim_{s \rightarrow \infty} g(s) = 0 & \text{di } u \text{ con una gaussiana 3D } u_r := G_r * u \end{cases}$

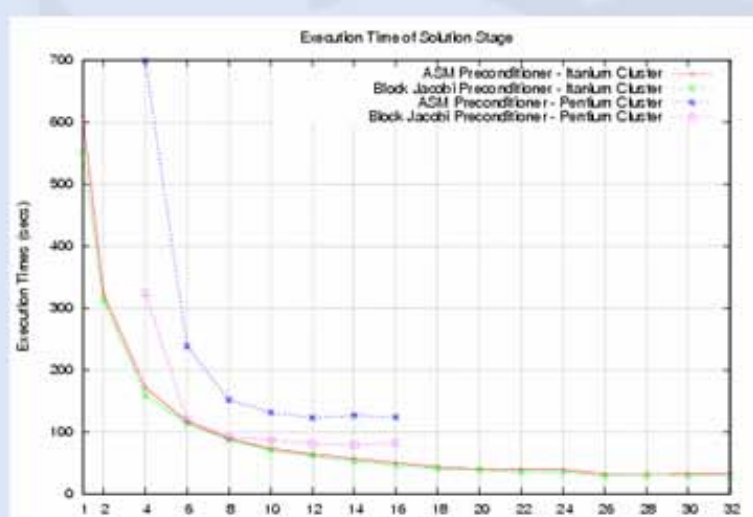
6 Risultati

Processore ITANIUM BIPROCESSOR 2 1.4 Ghz
Network Quadrics-QsNet 900 Mbytes

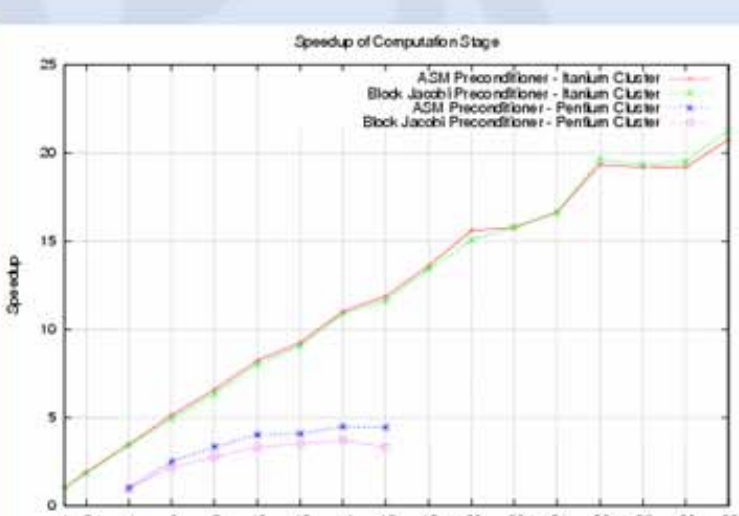
Processore Pentium 4.1.5 1,5 GHz
Switch Fast Ethernet 100 Mbytes



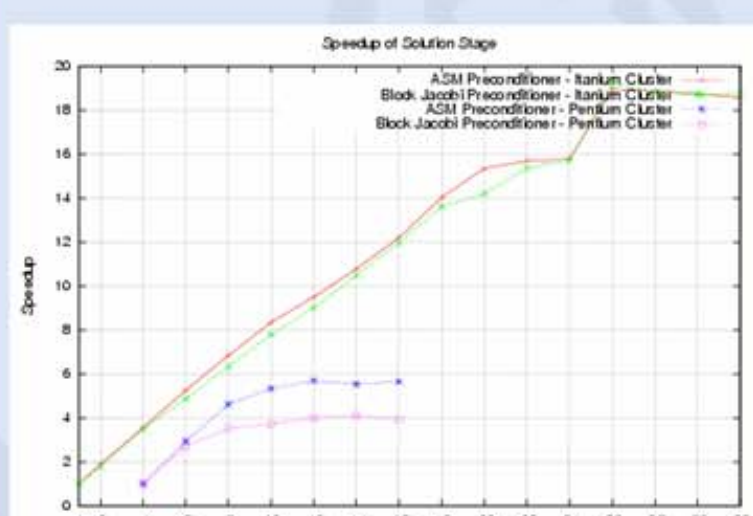
Tempo di esecuzione di un passo di scala per il denoising di tutta la sequenza assegnata



Tempo di risoluzione del secondo sistema relativo a tutti i frame e ad un passo di scala



Speed-up relativo ad un passo di scala per il denoising di tutta la sequenza assegnata



Speed-up relativo alla risoluzione del secondo sistema (un passo di scala per tutti i frame)

3 Modello Numerico

Per la discretizzazione:

- Schema Semi-Implicito per la scala
- Schema ai Volumi Finiti per lo spazio

$$\begin{cases} A_{HE} u_j^r = b_j^i \\ A_{ME} u_j^i = b_j^i \end{cases} \text{ con } A_{HE}, A_{ME} \in \mathcal{R}^{N \times N}, b_j^i \in \mathcal{R}^N \frac{\delta y}{\delta x}$$

Costruzione A_{HE}
% Ciclo sulla scala
for i=1, nscale **do**
% Ciclo sui frame
for j=0, nframe **do**
 $A_{HE} u_j^r = b_j^i$
endfor
endfor

% Matrice def. Positiva e Simmetrica (CG + BJ / ASM)

Costruzione A_{ME}

$$A_{ME} u_j^i = b_j^i$$

% Matrice def. Positiva e non Simmetrica (GMRES + BJ / ASM)

Un software parallelo per il denoising di sequenze di immagini ecografiche 3D

A. Murli¹, D. Casaburi¹, L. Carracciuolo², L. D'Amore¹, A. Galletti³



4 Necessità del Calcolo Parallelo

Fissato nframe > 10, ed $u \in \mathcal{R}^{n \times n \times n}$ → Complessità computazionale di ciascun frame $O(n^6)$
Necessità di ottenere il risultato in tempo utile e praticabile in diagnostica medica

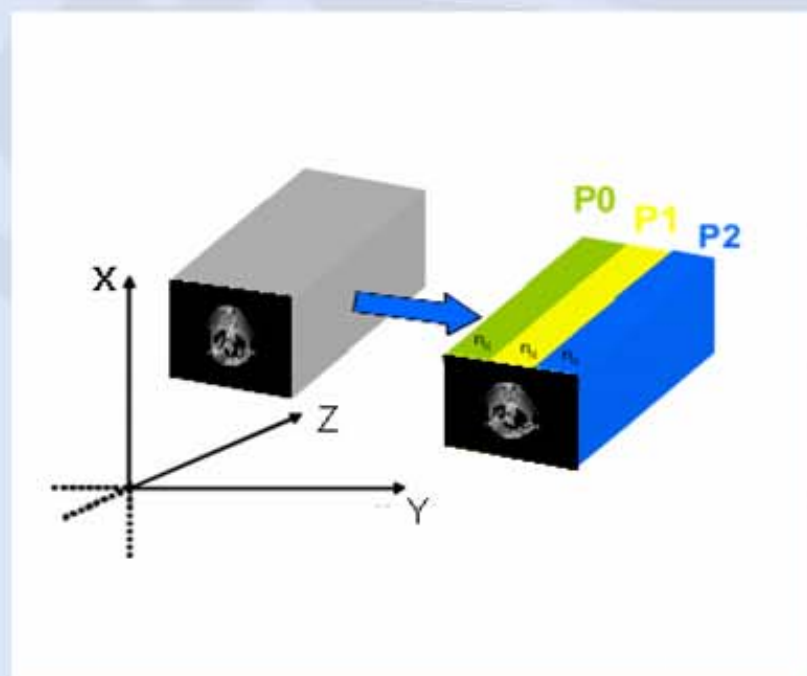
Progettazione ed Implementazione del software in ambiente di Calcolo Parallelo

Per una sequenza di 14 frame, ciascuno di dimensione $150 \times 150 \times 100$ → Complessità computazione totale $O(10^7)$

In parallelo, su architettura Beowulf, riduzione del 75% del tempo effettivo di risoluzione

5 Strategia di Parallelizzazione

Sperimentazioni effettuate su sequenza "in vivo" costituita da 14 frame 3D ciascuno dei quali di dimensione: $150 \times 150 \times 100$



Distribuzione di tipo *slice-block* di 1 frame di dimensione $n_1 \times n_2 \times n_3$ su $p = 3$ processi

$$\text{con } n_{id} = \begin{cases} n_3 / p + 1 & \text{se } id < \text{mod}(n_3 / p, p) \\ n_3 / p & \text{altrimenti} \end{cases}$$

Struttura gerarchica del toolkit **PETSc** (Portable, Extensible Toolkit for Scientific Computation)

Livello di Astrazione

Nonlinear Solvers (SNES)			Time Steppers (TS)		
Newton-based Methods	Line Search	Trust Region	Other	Euler	Backward Euler
Other				Pseudo Time Stepping	Other
Krylov Subspace Methods (KSP)					
GMRES	CG	CGS	Bi-CG-STAB	TFQMR	Richardson
Other					Chebyshev
Preconditioners (PC)					
Additive Schwarz	Block Jacobi	Jacobi	ILU	ICC	LU (Sequential only)
Other					Others
Matrices (Mat)					
Compressed Sparse Row (CSR)	Block Sparse Row (BSR)	Block Sparse Row (BSR)	Diagonal (BDIAG)	Dense	Matrix-free
Other					Other
Distributed Arrays (DA)			Index Sets (IS)		
Other			Indices	Block Indices	Stride
			Other		
Blas		Lapack		MPI	

Bibliografia:

- [1] S. Balay, K. Bushelman, W. Gropp, D. Kaushik, M. Knepley, L. Curfman McInnes, B. Smith, H. Zhang, *Petsc Users Manual*, ANL-95/11- Revision 2.1.3, Argonne National Laboratory, 2003.
- [2] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing, New York, 1996.
- [3] L. Carracciuolo, L. D'Amore, A. Murli, *Towards a Parallel Component for Imaging in PETSc Programming Environment: a Case Study in 3-D Echocardiography*, Parallel Computing, Vol. 32, gennaio 2006, pp. 67-83.

Affiliazioni: