Total Variation Regularization for Edge Preserving 3D SPECT Imaging in High Performance Computing Environments *

L. Antonelli[1], L. Carracciuolo[1], M. Ceccarelli[2], L. D'Amore[1], and A. Murli[1]

[1] University of Naples, Federico II
and Center for Reserach on Parallel Computing and Supercomputers(CPS)- CNR,
Complesso M.S.Angelo Via Cintia, 80126 Naples, Italy
(laura.antonelli,luisa.damore,almerico.murli)@dma.unina.it,
carracci@pixel.dma.unina.it

[2] University of Sannio, Via Port'Arsa 11, Benevento, Italy
ceccarelli@unisannio.it

Abstract. Clinical diagnosis environments often require the availability of processed data in real-time, unfortunately, reconstruction times are prohibitive on conventional computers, neither the adoption of expensive parallel computers seems to be a viable solution.
Here, we focus on development of mathematical software on high performance architectures for Total Variation based regularization reconstruction of 3D SPECT images. The software exploits the low-cost of Beowulf parallel architectures.

1 Introduction

The problem of reconstructing a 3D image from a set of its 2D projections from different angles of view arises in medical imaging for 3D tomography. 3D tomography requires huge amount of data and long computational times. Indeed, the reconstruction times are often prohibitive (of order of hours) on conventional machines. Hence, parallel computing seems to be an effective way to manage this problem, especially in a clinical environment.

However, there are still many barriers to an effective use of High Performance Computing (HPC) technology in medical imaging. First it can be employed in medical devices only to an extent which is determined by the overall cost of the system. On the other hand, the development and set up of efficient methods is a complex task which requires the joint effort of researchers in the fields of computing, industries, and medical communities.

This work is developed within a collaborative national project among computational scientists. Within this context, our focus is development of mathematical software on high performance architectures for reconstruction of 3D SPECT images by using edge preserving regularization. The main contribution of the paper

* The activity has been developed within the Project: "Inverse Problems in Medical Imaging" partially supported by MURST, grant number MM01111258

P.M.A. Sloot et al. (Eds.): ICCS 2002, LNCS 2330, pp. 171–180, 2002.

is, from one point, the introduction of the Total Variation-based edge preserving regularization method in an accurate 3D SPECT imaging model, and from the other point, due to the enormous computing requirements of the corresponding deconvolution problem, the development of a scalable parallel algorithm for the solution of the corresponding non-linear Euler Lagrange equation. Our algorithm exploits the parallelism in terms of the number of projections (rows of the system matrix). The proposed parallel algorithm and the experiments are based on low cost architectures made of Pentium Processors connected by a Fast Switch which could be really employed in a clinical environment

2 Regularized Image Reconstruction

Single photon emission computed tomography (SPECT) is a tomographic technique which gives the spatial distribution of radioactive tracers injected into the patient's body for a variety of diagnostic purposes. Data are collected by means of a small number of large detectors which provide a sequence of M 2D projections evenly spaced over over 2π around the patient.

The basic mathematical description of SPECT imaging relates the observed data u_0 with the original image u through a linear model:

$$u_0(s,\phi,z)=\int\int\int u(\mathbf{x},z')\mathcal{K}(s-\mathbf{x}\cdot\theta,\mathbf{x}\cdot\theta^{\perp},z-z')\mathrm{d}\mathbf{x}\mathrm{d}z' \tag{1}$$

where $\mathbf{x}=\{x,y\}$ are plane coordinates where the body of the patient is represented by a disk centered at the origin, z-axis is the axis of rotation of the detectors; $\theta=\{\cos\phi,\sin\phi\}$, ϕ being the projection angle of the detector plane with respect to a fixed position, s is the radial distance and $\theta^{\perp}=\{-\sin\phi,\cos\phi\}$. The kernel κ models the physical structure of the measurement device which introduce an unavoidable blurring effect due to the fact that the rays pass through the holes of a collimator. This model is called the Blurred Radon Transform [2], in which the integration kernel κ, which is usually called *Point Spread Function (PSF)*, differs from Radon's one for two reasons: the PSF is no longer an impulse function, it is defined on the projection plane as a function of two variables, s and z, and also depends on the source-detector distance $t=\mathbf{x}\cdot\theta^{\perp}$.

In compact form the above mathematical model of 3D SPECT imaging can be stated as follows:

$$u_0=K\,u+\eta \tag{2}$$

where η is assumed to be white noise, and K is a linear operator, the unknown vector u represents the 3D image and u_0 is the so-called *3D sinogram.* By this way it belongs to the class of linear inverse and ill posed problems. Indeed, due to the compactness of the operator $\mathcal{K}$, the computation of $\mathcal{K}^{-1}$ has the effect of noise amplification with the risk of obtaining unuseful solutions. These problem have been widely investigated, and the regularization approach produces an estimate, u^{λ} of u as the solution of

$$u^{\lambda}=\mathrm{argmin}_u\{||\mathcal{K}u-u_0||+\lambda\mathcal{R}(u)\} \tag{3}$$

where $\mathcal{R}(u)$ serves as regularization functional and λ is a parameter controlling the weight given to the regularization term relative to the residual norm.

The Total Variation (TV) norm, was first used by Rudin *et al.* [8], it is defined as

$$TV(u) = \int\int \sqrt{u_x^2 + u_y^2} \mathrm{d}x\mathrm{d}y. \tag{4}$$

The main advantage of the TV norm as regularization functional is that it does not require the solution to be continuous, thus allowing the presence of "jumps" in the solution, with a preservation of sharp edges. This is due to the fact that the TV functional is defined over the space of Bounded Variation functions [5], which can eventually contain step discontinuities. The solution of (3) with $\mathcal{R}$ given by (4) corresponds to the solution of the Euler-Lagrange equation

$$\mathcal{K}^*(\mathcal{K}u - u_0) - \lambda\nabla \cdot \left(\frac{\nabla u}{|\nabla u|}\right) = 0, \quad \frac{du}{dn} = 0 \tag{5}$$

The main difficulty in solving (5) is related to the highly non-linear second order elliptic term $\left(\frac{\nabla u}{|\nabla u|}\right)$ representing a diffusion operator with coefficient given by $\frac{1}{|\nabla u|}$. A number of approaches has been proposed for solving (5), which can be classified in three main categories

- Time marching [8]
- Fixed Point [9,4]
- Newton method [3].

In this paper we focus on the Fixed Point method as it provides the best compromise between robustness (dependence on the initial solution, parameters, ...) and speed of convergence (it can be shown to be a quasi–newton method). In particular, the solution is obtained by solving, until convergence, a sequence of linear systems arising from the discretization of the following equation

$$\left(\mathcal{K}^*\mathcal{K} - \lambda\nabla \cdot \left(\frac{\nabla}{|\nabla u^n|}\right)\right) u^{n+1} = \mathcal{K}^* u_0 \tag{6}$$

starting with $u^0 = u_0$.

It can be shown, [4], that the sequence of solutions $\{u^n\}_{n\in\mathcal{N}}$ converges as $n \to \infty$, to a global minimum of (3). The discretization of (6) leads to a linear system

$$(K^*K - \lambda L(u^n))u^{n+1} = K^* u_0 \tag{7}$$

where $L(u)$ represents the discretization of the operator $\nabla \cdot (\frac{\nabla}{|\nabla u|})$. In order to solve (7) we adopt the iterative conjugate gradient (CG) method whose main computational kernel is the multiplication of the system matrix $(K^*K - \lambda L(u^n))$ with the approximate solutions. The subject of the next section is therefore devoted to description of how parallelism has been introduced into solution of such system.

3 Parallel Reconstruction Algorithm

Our work has been essentially devoted to obtain a parallel version of the existing code by exploiting parallelism inside those modules representing the main computational bottlenecks. Actually, the most intensive calculation is due to the products which involve operators K^* and K respectively. Then in the following we mainly focus on these operations.

The matrix K is a $L \times L$ block Toeplitz matrix, where L is the number of object slices. Let us introduce the vector $\{\mathbf{u}_0^i\}_{i=1,...,L}$, $\mathbf{u}_0^i \in \Re^{J \cdot M}$, M being the number of angular projections, and J the number of beans. Moreover, let each block $\mathbf{u}^i \in \Re^{N \cdot N}$ represent the 2D slice, finally $[E_i] \in \Re^{(M \cdot J) \times (N \cdot N)}$ represents the projection matrix.

By this way, product Ku can be rewritten[1] as:

$$\begin{pmatrix} \mathbf{u}_0^1 \\ \vdots \\ \mathbf{u}_0^L \end{pmatrix} = \begin{pmatrix} [E_0] & \dots & [E_{L-1}] \\ \vdots & \vdots & \vdots \\ [E_{L-1}] & \dots & [E_0] \end{pmatrix} \begin{pmatrix} \mathbf{u}^1 \\ \vdots \\ \mathbf{u}^L \end{pmatrix} \tag{8}$$

Standard acquisition parameters are $M = 120$, $J = 128$ and $L = 64$.

The way in which concurrency has been introduced was strongly induced by the physical meaning of the SPECT reconstruction process. Specifically, we observe that the $E_j \cdot \mathbf{u}^i$ products, i.e. the *projection* process of SPECT imaging on each 2D slice, can be described as a set of distinct projections each one corresponding to a projection angle. In this way the whole projection process synthesized by equation (2) is obtained by repeatedly performing the projection operation for each projection angle. Starting from this consideration, we can distribute projection angles among processors. In other words, each processors is assigned a set of projection angles ϕ over which the projection is performed. As a consequence, the whole set of sinograms and image slices is shared among the processors. The first phase of the reconstruction process performs a distribution of the sinograms among all the processor. Then, each iteration of the algorithm requires a global sum in order to keep updated the current solution of the system (7).

Let us now state the parallel algorithm using a linear algebra setting. We note that the most natural way to accomplish the parallelism as inspired by the projection/retroprojection processes is to perform parallelization of products

$$\mathbf{f}^i = \sum_{j=0}^{L-1} E_{(j+i-1) \text{ mod } L} \cdot \mathbf{u}^{j+1} \quad i = 1, ..., L$$

in a row-wise fashion. In other words, concurrency has been introduced by using a row-block cyclic distribution of the matrices $E_{(j+i-1) \text{ mod } L}$. By this way each

[1] It is worth noting that the block representation of K as described in (10) is not actually available. In practice, the matrix K is rather sparse with a sparsity between 5% and 10%. The coefficients of each row of the matrix K are given under form of look-up tables [2].

processor performs the product between a row-block of $E_{(j+i-1) \bmod L}$ and vector $\mathbf{u}^{j+1}$. In the same way, parallelization of the products

$$\mathbf{z}^i = \sum_{j=0}^{L-1} E^*_{(j+i-1) \bmod L} \cdot \mathbf{f}^{j+1} \quad i = 1, ..., L$$

is exploited in a column-wise fashion. This means that each process acts a partial update over the whole data, requiring a global sum at the end of each step.

As already underlined, the above choice was motivated by several reasons, the first is due to the structure of the look-up tables for accessing the elements of the projection matrix. Indeed, the look-up table containing the coefficients of K is given in a closed form in such a way that we can access independently to each row of K corresponding to a given angular projection over the whole data set. The second was due to the fact that a further colomnwise distribution of the matrix K, from one hand could allow a distribution of the data, but, on the other hand, it would require a further global communication step at each computation of $K\mathbf{u}$. Considering the actual dimentions of the matrices, and the obtained results, with the processors arranged as a ring, we believed this last option the most suited for our application. In addition, this choice required minor adjustment of existing projection code.

For what concerns the parallelization of the matrix product $L(u^n)u^{n+1}$, it is important to point out that it requires about the 0.1% of the whole processing time for each iteration, therefore since we adopted a global input data distribution, *i. e.* the input sinogram and the current solution are shared among all the processors, we simultaneously perform this operation locally as it does not influence the whole computing time. Other choices are of course possible, however the parallelization of this operation would introduce a further communication overhead without improving the processing time.

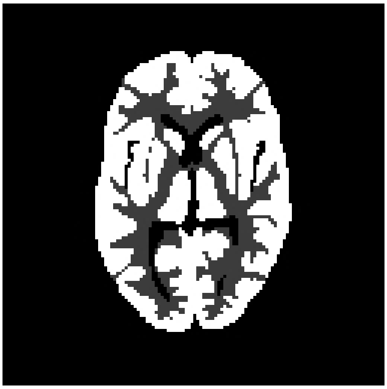

Fig. 1. The Hoffman phantom used in the experiments

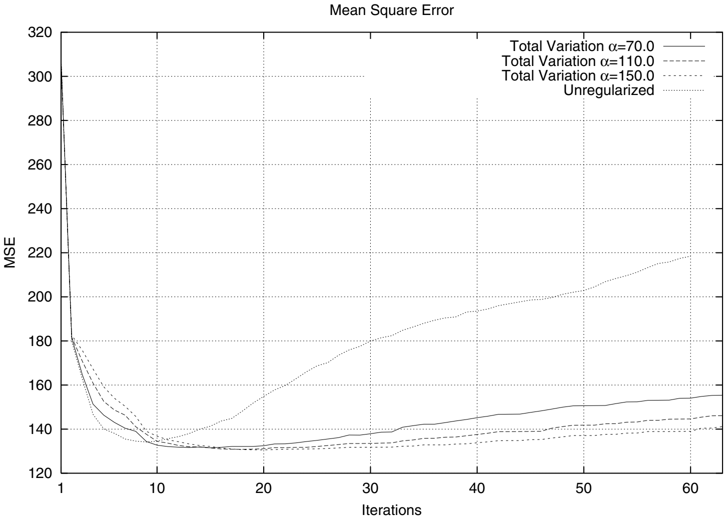

Fig. 2. The MSE between the computed and ideal solution for the unregularized problem and for the regularized problem with various regularization parameters. Note the *semiconvergence* property of the unregularized iterative algorithm.

4 Experiments and Results

The above algorithm was experimented by using a Hoffman phantom depicted in figure 1. Projection data were generated in a 128x128 image array for 120 views over $[0, \pi)$, the fully 3D projector assumes a Gaussian model of collimator blurr both in the 2D slices and along the staking direction of the slices [1]. Figure 2 demonstrates the stability of the TV-based algorithm with respect to the standard CG-based iterative algorithm [1], the figure reports the Mean Squared Error (MSE) between the computed solution and the ideal solution as function of the iterations. Note that, in order to make the comparison homogeneous, the MSE measure for the TV-based algorithm was computed at each inner iteration of the adopted CG algorithm for solving the linear system (7). The behaviour of the MSE clearly shows the classical *semiconvergence* property of iterative reconstruction algorithms, in particular the unregularized algorithm has an optimal number of iterations after which the quality of the reconstruction

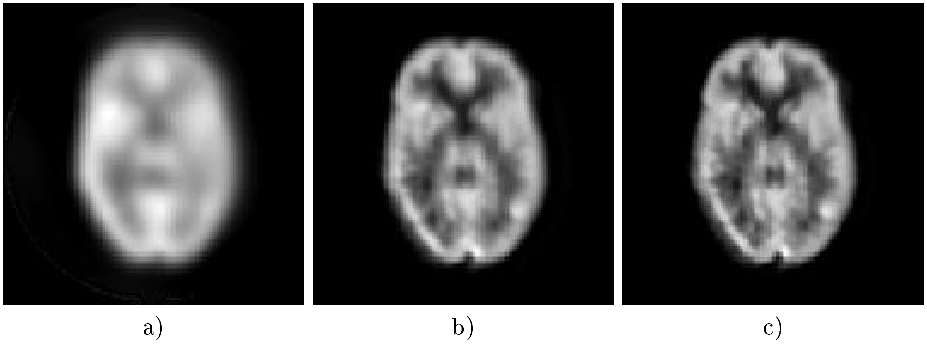

Fig. 3. The TV-regularized solutions. Figure a) is obtained with 1 TV iteration, b) refers to 4 TV iteration and c) to 7 iterations. Note that each TV iteration requires 7 inner CG iterations.

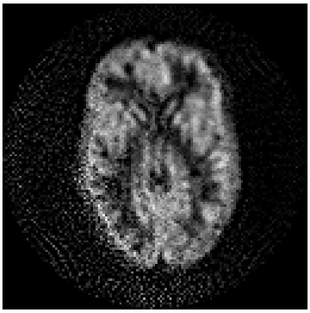

Fig. 4. The unregularized solution after 60 CG iterations.

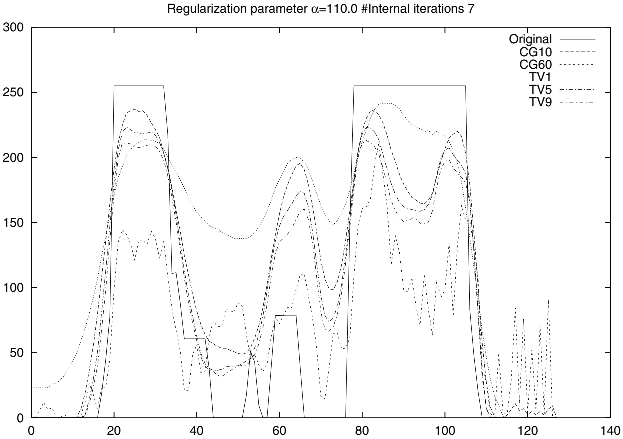

Fig. 5. Cross-section of the solution for various iteration number of the regularize and unregularized problem.

degrades, such optimal number of iterations being typically chosen with experimentation, depending on the amount of noise and the kind of images. On the contrary, the TV-regularized algorithm offers the warranty to converge to an accurate and stable solution for a wide range of regularization parameters. Neither such regularization tends to destroy image features. We plot in figure 3 the regularized solutions for different regularization parameters and iteration number. The unregularized solution is reported in figure 4. What it is evident here is the property of the TV regularization to produce a stable solution while mantaining the features of the images, *i.e.* the regularization is *discontinuity adaptive* [6] in the sense that the sharp changes (edges) are preserved. As a further example, figure 5 reports several linear profiles of the regularized solution together with the unregularized one, it shows how the TV-based reconstruction preserves the stepwise behavior of the original image.

For what concerns the computing environment, we implemented the algorithm on two Beowulf systems, available at CPS-CNR (Naples). The first, referred in the figures as *Beocomp* has 32 nodes, each node being a 450 Mhz Pentium II processor running Linux Red Hat 6.0. The second, referred as *Vega*, consisting of 18 Pentium IV running Red Hat Linux 7.2. The program uses the MPI and the Basic Linear Algebra Communication Subprograms (BLACS) communication libraries. Figures 7 and 8 report respectively the achieved speedup and the corresponding efficiency on both systems. Results show that the problem of 3D SPECT imaging can be efficiently solved on high performance architectures. The time reduction gained with the adoption of such parallel algorithm is in accordance with diagnostic times, allowing the use of accurate iterative reconstruction algorithms.

5 Conclusions

In conclusion the contribution of the paper is twofold:

- introduction of the Total Variation-based edge preserving regularization method in SPECT imaging
- development of a scalable parallel algorithm for the solution of the corresponding non-linear Euler Lagrange equation. Our algorithm exploits the parallelism both in terms of the number of projections (rows of the system matrix) and 2-D slices (columns of the system matrix).

References

1. D. Baldini, P. Clavini and A. R. Formiconi, 'Image reconstruction with conjugate gradient algorithm and compensation of the variable system response for an annular SPECT system', *Phys. Medica*, vol 14, pp. 159-173, 1998.

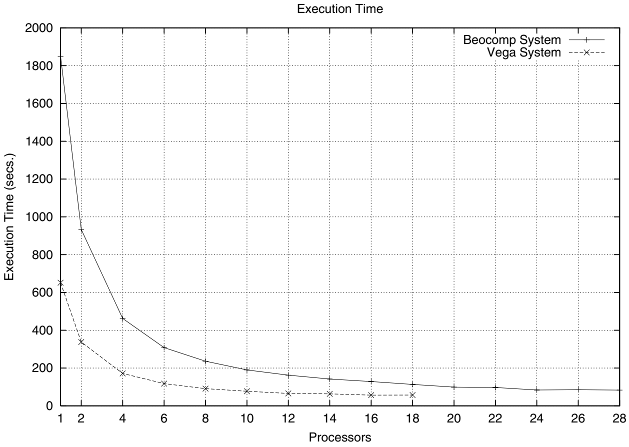

Fig. 6. Execution times as function of the number of processors

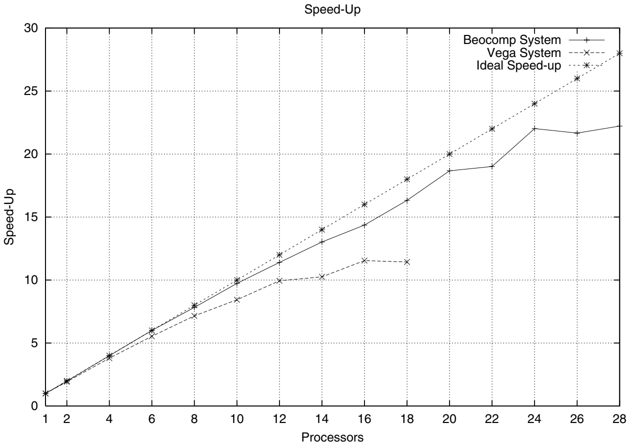

Fig. 7. Speedup as function of the number of processors

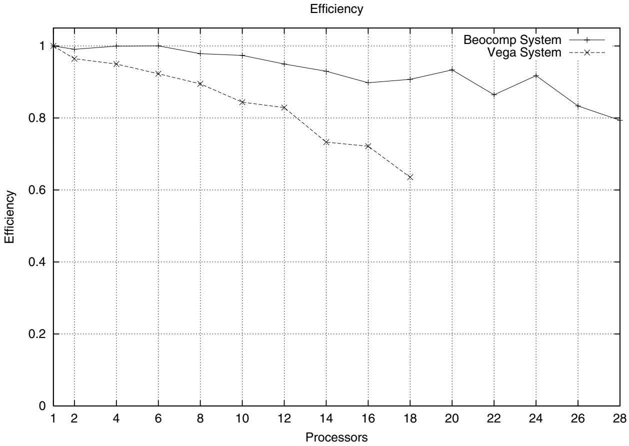

Fig. 8. Efficiency as function of the number of processors

2. P. Boccacci, P. Bonetto, P. Calvini and A. R. Formiconi, 'A simple model for the efficient correction of collimator blur in 3D SPECT imaging', *Inverse Problems*, vol. 15, pp. 907-930, 1999.
3. T. Chan, G. Golub and P. Mulet, 'A primal-dual method for total variation-based image reconstruction', UCLA CAM Report n. CAM-95-43, 1995.
4. P. Charbonnier, L. Blanc-Ferlaud, G. Aubert, M. Barlaud, 'Deterministic edge-preserving regularization in computed imaging', *IEEE Trans. on Image Processing*, vol. 6, pp. 298-311.
5. E. Giusti, *Minimal surfaces and functions of bounded variation*, Birkhauser, Boston, 1984.
6. S. Z. Li, 'On discontinuity-adaptive smoothness priors in computer vision", *IEEE Transactions on Pattern Analysis and machine Intelligence*, vol. 17, pp. 576-586, 1995.
7. A. Passeri, A. R. Formiconi, M. T. De Cristofaro, A. Pupi and U. Meldolesi, 'High performance computing and networking as tools, for accurate emission computed tomography reconstruction', *Europ. Journal of Nuclear Medicine*, vol. 24, n. 4, pp. 390-397, 1997.
8. L. Rudin, S. Osher, and E. Fatemi, 'NonLinear total variation based noise removal algorithms', *Physica D*, 1992, **60**, pp. 259–268.
9. C. R. Vogel and M. E. Oman, 'Iterative methods for total variation denoising', *SIAM J. Sci. Statist. Computation*, vol. 17, pp. 227-238, 1996.