A public parallel method for astronomical image restoration *

Francisco Almeida [1] Carlos Delgado[3] Ramón J. García López[23] and Francisco de Sande[1]

[1] Depto. de Estadística, Investigación Operativa y Computación, Univ. de La Laguna, 38271–La Laguna, Spain; {falmeida, fsande}@ull.es
[2] Departamento de Astrofísica Univ. de La Laguna 38206–La Laguna. Spain; rgl@iac.es
[3] Instituto de Astrofísica de Canarias (IAC), 38205–La Laguna, Spain; delgadom@iac.es

Any recorded image is blurred whenever the instrument used to obtain it has a finite resolving power. For example, the image of a point source seen through a telescope has an angular size which is inversely proportional to the diameter of the primary mirror of the telescope. If the instrument is ground-based, the image is additionally degraded by the turbulent motions in the earth's atmosphere.

Image restoration is a useful technique relevant in many fields: medicine and astronomy among others. Basic methods applied to image restoration include filtering, speckle deconvolution and blind deconvolution. Several sequential software approaches are available to tackle the problem. Most of them are provided as commercial software packages. Among others, we can mention the implementations on IDL [1], Pixon [2] or Memsys 5 (an implementation of a maximum entropy method).

In the case of astronomy, much effort is presently devoted to the improvement of the spatial resolution of images, either via the introduction of new observing techniques (e.g. interferometry or adaptive optics [3]) or via a subsequent numerical processing of the image (deconvolution). It is, in fact, of major interest to combine both methods to reach an even better resolution. The scientific motivation of such effort is to further exploit the present available data and to reduce the uncertainty on some of the conclusions obtained from the images.

In this work we present work in progress in the design of a parallel image restoration system based on a novel algorithm. The main contribution in our approach resides in the minimization of an analytical upper bound to the algorithmic complexity of the image. The details in ground telescope images deconvolved with our algorithm are comparable than those from the Hubble Space Telescope, which is considered as the *de facto* best imaging telescope up to date.

With the new algorithm, our aim is to target astronomical images recorded at ground based telescopes and satellites. Nevertheless, the method is general enough to be applied to any kind of spatial invariant blurred image iff the statistics of the image is known. In our target scenario (astronomical images), the complexity of the problem grows with the number of pixels at the image by the

* This work has been partially supported by the EC (FEDER) and the Spanish MCyT (Plan Nacional de I+D+I, TIN2005-09037-C02-01)

number of pixels of an isolated star. Therefore, for usual image sizes (1024×1024 pixels) the computational time is in the range of months on a scalar computer.

A clear alternative to minimize the computing time for this problem is the use of High Performance Computing (HPC) techniques. HPC approaches have been successfully applied to this problem by different authors in the recent past. Image spatial domain decomposition is the most usual approach when parallelizing these class of algorithms. Both linear and non-linear image processing techniques are used to mitigate the effects of atmospheric turbulences on data collected at the Maui Space Surveillance System [4]. In [5] the authors use a domain decomposition adaptive to the spread of the blurring image. Parallelizations of both bispectrum and deconvolution algorithms are presented in [6].

The contributions of our work are manifold: the sequential algorithm is a contribution by itself. We present the parallelization of the sequential code using different alternatives. Although we plan to be portable to many parallel architectures (shared and distributed memory, hybrid and heterogeneous architectures), our preliminary results have been obtained using an OpenMP parallelization on a IBM RS6000. Our final aim is to develop a web based tool that solves the problem at real-time for standard size images. The user will provide the images while the parallel application will be offered as a usual service through a web service architecture. The web service will play the role of a public repository of original and unblurred astronomical images. Currently a MPI based distributed memory version of the parallel code is available and experimental results will be provided for the final version of this contribution, if accepted.

The image restoration problem can be stated as inverting for i_{ij} the following equation:

$$d_{ij} = \sum_{k,l} R_{ij}^{kl} i_{kl} + n_{ij} \tag{1}$$

where d_{ij} and i_{kl} are the observed and true light distribution at pixel ij respectively, R_{ij}^{kl} is the probability of detecting a photon with true position kl at pixel ij (also known as Point Spread Function PSF), and n_{ij} is the contribution from readout noise in the telescope camera at pixel ij. Usually it is assumed that R_{ij}^{kl} depends only on the differences $i-k, j-l$, and n_{ij} follows a Gaussian distribution with fixed width and zero mean.

The operation applied on each pixel of the image is the same. At a fixed iteration of the algorithm, the process can be performed in an independent manner for all the pixels of the image.

The complexity order of the evaluation is the same for each pixel. We exploit the *data parallelism* based on the decomposition of the image spatial domain that appears as the most suitable technique to parallelize the code. Therefore, the sequential code is best suited for different parallel implementations varying from message-passing to shared-memory codes and hybrid solutions combining both. The domain decomposition leads to a good restored image and a balanced parallel application.

In the last years OpenMP and MPI have been universally accepted as the standard tools to develop parallel applications. OpenMP is a standard for shared memory programming. It uses a fork-join model and it is mainly based on compiler directives that are added to the code that indicate the compiler regions of code to be executed in parallel. MPI uses an SPMD model. Processes can read and write only to their respective local memory. Data are copied across local memories using subroutine calls. The MPI standard defines the set of functions and procedures available to the programmer. Each one of these two alternatives have both advantages and disadvantages, and very frequently it is not obvious which one should be selected for a specific code. The pure cases of MPI or OpenMP programming models have been widely studied in plenty of architectures using scientific codes. The case of mixed mode parallel programming has also deserved the attention of some researchers in the last few years [7], however the effective application of this paradigm still remains as an open question.

The parallelization of the sequential version of our algorithm is quite simple in shared memory architectures. Using OpenMP, different rows of the image can be assigned to different threads as they are completely independent. This is achieved by introducing the adequate OpenMP `#pragma` with a static scheduling. Data are available from one iteration to the next through the common memory. In distributed memory architectures, both the convolution and the Lucy-Richardson like step require send/receive communication operations among each iteration to exchange the borders of the rows assigned to each processor. According to the former semantic, in hybrid architectures we will launch MPI processes on several nodes and each MPI process spawns the OpenMP threads inside each node.

We have obtained an acceptable performance for our parallelization on an IBM RS-6000. An increasing speedup is observed when the number of processors is increased. We consider that an speedup of 13.52 when using 16 processors is an acceptable performance for this preliminary version.

References

1. IDL The Data Visualization and Analysis Platform, `http://www.rsinc.com/idl/`.
2. R. C. Puetter, A. Yahil, The pixon method of image reconstruction, in: D. M. Mehringer, R. L. Plante, D. A. Roberts (Eds.), Astronomical Data Analysis Software and Systems VIII, Vol. 172 of ASP Conference Series, 1999, pp. 307–316.
3. P. Lená, Observational Astrophysics, Springer Verlag, 1988.
4. K. Schulze, M. Murphy, D. Tyler, The maui image manager and online systems archive (mimosa): Inte lligent observatory data management and processing, in: Proceedings of the 1999 AMOS Technical Conference, 1999.
5. A. Bevilacqua, E. L. Piccolomini, Parallel image restoration on parallel and distributed computers., Parallel Computing 26 (4) (2000) 495–506.
6. K. Schulze, A comparison of parallelization techniques for bispectrum and blind deconvolution, in: Proceedings of the 2002 AMOS Technical Conference, 2002.
7. L. Smith, M. Bull, Development of mixed mode MPI/OpenMP applications, Scientific Programming 9 (2–3) (2001) 83–98.