

Poisson Denoising on the Sphere

J. Schmitt^a, J.L. Starck^a, J. Fadili^b, I. Grenier^a, J.M. Casandjian^a

^aIRFU/SEDI-SAp, Service d'Astrophysique, CEA-Saclay, 91191 Gif-sur-Yvette Cedex, France;

^bGREYC CNRS UMR 6072, ENSICAEN, 6, Bd du Maréchal Juin, 14050 Caen Cedex, France

ABSTRACT

In the scope of the Fermi mission, Poisson noise removal should improve data quality and make source detection easier. This paper presents a method for Poisson data denoising on sphere, called Multi-Scale Variance Stabilizing Transform on Sphere (MS-VSTS). This method is based on a Variance Stabilizing Transform (VST), a transform which aims to stabilize a Poisson data set such that each stabilized sample has an (asymptotically) constant variance. In addition, for the VST used in the method, the transformed data are asymptotically Gaussian. Thus, MS-VSTS consists in decomposing the data into a sparse multi-scale dictionary (wavelets, curvelets, ridgelets...), and then applying a VST on the coefficients in order to get quasi-Gaussian stabilized coefficients. In this present article, the used multi-scale transform is the Isotropic Undecimated Wavelet Transform. Then, hypothesis tests are made to detect significant coefficients, and the denoised image is reconstructed with an iterative method based on Hybrid Steepest Descent (HST). The method is tested on simulated Fermi data.

Keywords: wavelets, Poisson noise, variance stabilizing transform

1. INTRODUCTION

The Fermi Gamma-ray Space Telescope is a powerful space observatory which observes the whole universe in gamma-rays. Fermi's main instrument, the Large Area Telescope (LAT), catches photons in an energy range between 8 keV to greater than 300 GeV. As the amount of photons is small, the spherical photons counts images obtained by Fermi are degraded by quantum noise due to fluctuations on the number of detected photons. The basic photon-imaging model assumes that the number of detected photons at each pixel location is Poisson distributed. More specifically, the image is considered as a realization of an inhomogeneous Poisson process. This quantum noise makes the source detection difficult, consequently it is important to have an efficient denoising method for spherical Poisson data.

Let us first formulate the problem of Poisson denoising on the sphere. We observe a discrete dataset of counts $\mathbf{X}(\theta, \varphi)$ where $\mathbf{X}(\theta, \varphi)$ is a Poisson random variable of intensity $\Lambda(\theta, \varphi)$, i.e., $\mathbf{X}(\theta, \varphi) \sim \mathcal{P}(\Lambda(\theta, \varphi))$. Here we suppose that $\mathbf{X}(\theta, \varphi)$'s are mutually independant. The denoising aims at estimating the underlying intensity profile Λ from the observation \mathbf{X} .

Several techniques have been proposed in the literature to account for the Poisson intensity estimation problem. A major category of methods adopt a multiscale bayesian framework specifically tailored for Poisson data., independently initiated by Timmerman and Nowak¹ and Kolaczyk.² Lefkimiatis et al.³ proposed an improved statistical model for analyzing Poisson processes, based on a multiscale representation of the Poisson process in which the ratios of the underlying Poisson intensities in adjacent scales are modeled as mixtures of conjugate parametric distributions. Another approach includes preprocessing of the count data by a variance stabilizing transform (VST) such as the Anscombe⁴ and the Fisz⁵ transforms, applied either in the spatial⁶ or in the wavelet domain.⁷ The transform reforms the data so that the noise approximately becomes Gaussian with a constant variance. Standard techniques for independant identically distributed Gaussian noise are the used for denoising. Zhang proposed a powerful method called Multi-Scale Variance Stabilizing Transform (MS-VST).⁸ It consists in combining a VST with a multiscale transform (wavelets, ridgelets or curvelets), yielding asymptotically normally distributed coefficients with known variances. The choice of the multi-scale method depends on the morphology of the data. Wavelets represent more efficiently regular structures and isotropic singularities, whereas ridgelets are designed to represent global lines in an image, and curvelets represent

efficiently curvilinear countours. Significant coefficients are then detected with hypothesis tests, and the final estimate is reconstructed with an iterative scheme.

The aim of this paper is to extend the MS-VST to the analysis of spherical Poisson data images, in order to denoise Fermi photon counts images, and also a wide range of Poisson data such as medical or astronomical imaging. This extension is possible thanks to the last improvements on multi-scale transforms on the sphere. Abrial⁹ developed a range of multi-scale transforms (decimated and undecimated wavelet transform, ridgelet transform, curvelet transform) which are easy to compute with the HEALPix pixellisation and have a very simple inversion algorithm. An extensions of the algorithm to background separation is also proposed.

2. MULTI-SCALE TRANSFORMS ON THE SPHERE

New multi-scale transforms on the sphere were developed by Abrial et al.,⁹ namely the isotropic undecimated wavelet transform, the pyramidal wavelet transform, the ridgelet transform and the curvelet transform. These transforms can be inverted and are easy to compute with the HEALPix pixellisation, and are used in denoising, deconvolution, morphological component analysis and inpainting applications.

In this article, we work with the isotropic undecimated wavelet transform on the sphere (IUWT), which is based on the spherical harmonics transform and has a very simple reconstruction algorithm. At scale j , we note $a_j(\theta, \phi)$ the scale coefficients, and $d_j(\theta, \phi)$ the wavelet coefficients. Given a scale coefficient a_j , the smoother coefficient a_{j+1} is obtained by a convolution with a low pass filter $h_j : a_{j+1} = a_j * h_j$. The wavelet coefficients are defined by the difference between two consecutive resolution : $d_{j+1} = a_j - a_{j+1}$. The reconstruction is then straightforward:

$$a_0(\theta, \phi) = a_J(\theta, \phi) + \sum_{j=1}^J d_j(\theta, \phi) \quad (1)$$

Since this decomposition is redundant, the procedure for reconstruction an image from its coefficients is not unique and this can profitably used to impose additional constraints on the synthesis functions (e.g. smoothness, positivity) used in the reconstruction. A reconstruction algorithm based on a bank of filter is described in.⁹

3. PRINCIPE OF VARIANCE STABILIZING TRANSFORM

Zhang et al. proposed a 2D Poisson denoising method based on a Variance Stabilizing Transform (VST).⁸ This method is inspired of Gaussian denoising methods using sparsity in a multi-scale transform dictionary (wavelets, ridgelets, curvelets).

3.1 VST of a Poisson process

Given Poisson data $\mathbf{X} := (X_i)_i$, each sample $X_i \sim \mathcal{P}(\lambda_i)$ has a variance $\text{Var}[X_i] = \lambda_i$. Thus, the variance of \mathbf{X} is signal-dependant. The aim of a VST T is to stabilize the data such that each coefficient of $T(\mathbf{X})$ has an (asymptotically) consistent variance, say 1, irrespective of the value of λ_i . In addition, for the VST used in this study, $T(\mathbf{X})$ is asymptotically normally distributed. Thus, the VST-transformed data are asymptotically stationary and gaussian.

Anscombe transform is a widely used VST which has a simple square-root form

$$T(X) := 2\sqrt{X + 3/8} \quad (2)$$

We can show that $T(X)$ is asymptotically normal as the intensity increases.

$$T(x) - 2\sqrt{\lambda} \xrightarrow[\lambda \rightarrow +\infty]{\mathcal{D}} \mathcal{N}(0, 1) \quad (3)$$

It can be shown that Anscombe transform requires a high underlying intensity to well stabilize the data (typically for $\lambda \geq 10$).

3.2 VST of a filtered Poisson process

Given a Poisson process $\mathbf{X} := (X_i)_i$ where X_i 's are independent and $X_i \sim \mathcal{P}(\lambda_i)$, $Y_j := \sum_i h[i]X_{j-i}$ is the filtered process obtained by convolving $(X_i)_i$ with a discrete filter h . We will use Y to denote any one of Y_j . Let us define $\tau_k := \sum_i (h[i])^k$ for $k = 1, 2, \dots$. In addition, we adopt a local homogeneity assumption than $\lambda_{j-i} = \lambda$ for all i within the support of h .

We define the square-root transform T as follows:

$$T(Y) := b \cdot \text{sgn}(Y + c)|Y + c|^{1/2} \quad (4)$$

where b is a normalizing factor. Lemma 1 proves that T is a VST for a filtered Poisson process (with a nonzero-mean filter) in that $T(Y)$ is asymptotically normally distributed with a stabilized variance as λ becomes large.

LEMMA 1. **(Square root as VST)** If $\tau_1 \neq 0$, $\|h\|_2, \|h\|_3 < \infty$, then we have :

$$\text{sgn}(Y + c)\sqrt{|Y + c|} - \text{sgn}(\tau_1)\sqrt{|\tau_1|\lambda} \xrightarrow[\lambda \rightarrow +\infty]{\mathcal{D}} \mathcal{N}\left(0, \frac{\tau_2}{4|\tau_1|}\right) \quad (5)$$

where $\text{sgn}(\cdot)$ is the sign function.

4. MULTI-SCALE VARIANCE STABILIZING TRANSFORM ON THE SPHERE

4.1 Stabilization

The multi-scale variance stabilizing transform on the sphere (MS-VSTS) consists in combining the square-root VST with a multi-scale transform, here the isotropic undecimated wavelet transform on the sphere (IUWT). The recursive scheme is:

$$\text{IUWT} \begin{cases} a_j &= h_{j-1} * a_{j-1} \\ d_j &= a_{j-1} - a_j \end{cases} \implies \text{MS-VST+IUWT} \begin{cases} a_j &= h_{j-1} * a_{j-1} \\ d_j &= T_{j-1}(a_{j-1}) - T_j(a_j) \end{cases} \quad (6)$$

In (6), the filtering on a_{j-1} can be rewritten as a filtering on $a_0 = \mathbf{X}$, i.e., $a_j = h^{(j)} * a_0$, where $h^{(j)} = h_{j-1} * \dots * h_1 * h_0$ for $j \geq 1$ and $h^{(0)} = \delta$. T_j is the VST operator at scale j :

$$T_j(a_j) = b^{(j)} \text{sgn}(a_j + c^{(j)})\sqrt{|a_j + c^{(j)}|} \quad (7)$$

Let us define $\tau_k^{(j)} := \sum_i (h^{(j)}[i])^k$. According to Zhang, to have an optimal asymptotic rate for the VST, the constant $c^{(j)}$ associated to $h^{(j)}$ should be set to:

$$c^{(j)} := \frac{7\tau_2^{(j)}}{8\tau_1^{(j)}} - \frac{\tau_3^{(j)}}{2\tau_2^{(j)}} \quad (8)$$

The stabilization procedure is directly invertible as we have:

$$a_0 = T_0^{-1} \left[T_J(a_J) + \sum_{j=1}^J d_j \right] \quad (9)$$

Setting $b^{(j)} := \text{sgn}(\tau_1^{(j)})/\sqrt{|\tau_1^{(j)}|}$, if λ is constant within the support of the filter $h^{(j)}[k - \cdot]$, then we have:

$$d_j[k] \xrightarrow[\lambda \rightarrow +\infty]{\mathcal{D}} \mathcal{N}\left(0, \frac{\tau_2^{(j-1)}}{4\tau_1^{(j-1)^2}} + \frac{\tau_2^{(j)}}{4\tau_1^{(j)^2}} - \frac{\langle h^{(j-1)}, h^{(j)} \rangle}{2\tau_1^{(j-1)}\tau_1^{(j)}}\right) \quad (10)$$

It means that the detail coefficients issued from locally homogeneous parts of the signal follow asymptotically a centered normal distribution with an intensity-independant variance which only relies on the filter h and on the

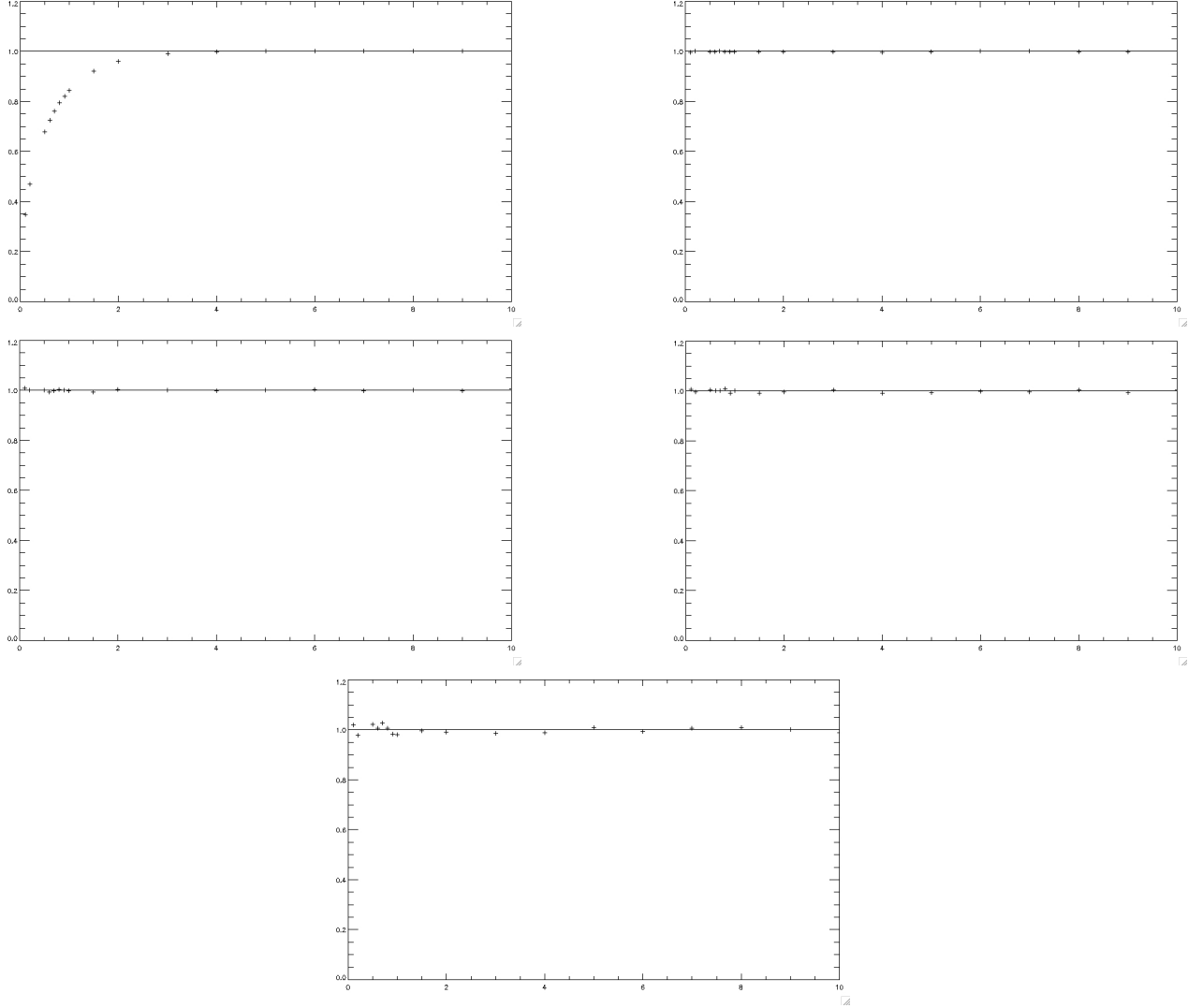


Figure 1. Normalized value $((\sigma_{(j)})_{\text{experience}}/(\sigma_{(j)})_{\text{theoretical}})$ of the stabilized variances at each scale j as function of λ : $\sigma_{(1)}$, $\sigma_{(2)}$, $\sigma_{(3)}$, $\sigma_{(4)}$ and $\sigma_{(5)}$

current scales. Consequently, the stabilized variances and the constants $b^{(j)}, c^{(j)}, \tau_k^{(j)}$ can all be pre-computed. Let us define $\sigma_{(j)}^2$ the stabilized variance at scale j for a locally homogeneous part of the signal:

$$\sigma_{(j)}^2 = \frac{\tau_2^{(j-1)}}{4\tau_1^{(j-1)^2}} + \frac{\tau_2^{(j)}}{4\tau_1^{(j)^2}} - \frac{\langle h^{(j-1)}, h^{(j)} \rangle}{2\tau_1^{(j-1)}\tau_1^{(j)}} \quad (11)$$

The $\sigma_{(j)}$ have been precomputed for a 6-scaled isotropic undecimated wavelet transform: $\sigma_{(1)} = 0.484704$, $\sigma_{(2)} = 0.0552595$, $\sigma_{(3)} = 0.0236458$, $\sigma_{(4)} = 0.0114056$, $\sigma_{(5)} = 0.00567026$. We have simulated Poisson images of different constant intensities λ , computed the IUWT with MS-VST on each image and observed the variation of the normalized value of $\sigma_{(j)}$ $((\sigma_{(j)})_{\text{experience}}/(\sigma_{(j)})_{\text{theoretical}})$ in function of λ for each scale j (Fig. 1). We see that the wavelet coefficients are stabilized when $\lambda \gtrsim 4$.

4.2 Denoising

On each scale j , we detect significant wavelet coefficients by comparing $|d_j[k]|$ with a threshold $S_j = 3, 4$ or $5\sigma_{(j)}$. We assume that:

- if $|d_j[k]| \geq S_j$, $d_j[k]$ is significant.
- if $|d_j[k]| < S_j$, $d_j[k]$ is not significant.

Then we have to invert the MS-VSTS scheme to reconstruct the estimate. Even if the direct inversion is possible (Eq. (9)), it can not guarantee a positive reconstruction, as the Poisson intensity is always nonnegative. A positivity projection can be applied on the inverse, but important structures could lose in the estimate. Thus, Zhang proposes to reformulate the reconstruction as a convex optimisation problem and solve it iteratively.

We define the multiresolution support \mathcal{M} , which is determined by the set of the detected significant coefficients:

$$\mathcal{M} := \{(j, k) | d_j[k] \text{ is significant}\} \quad (12)$$

We are looking for a denoised image \mathbf{y} so that $\mathbf{y} \geq 0$ and $\forall (j, k) \in \mathcal{M}, (\mathcal{W}\mathbf{y})_j[k] = (\mathcal{W}\mathbf{X})_j[k]$.

This problem is solved with an iterative algorithm: the reconstructed image $\mathbf{y} = \mathcal{R}\mathbf{d}$ is initialised by $\mathbf{y}^{(0)} = 0$ and, for $k = 0$ to $N_{\max} - 1$, $\mathbf{y}^{(k+1)} = P_+(\mathbf{y}^{(k)} + \mathcal{R}\mathcal{M}\mathcal{W}(\mathbf{X} - \mathbf{y}^{(k)}))$, where P_+ represents the projection on the nonnegative orthant, \mathcal{W} represents the Wavelet transform operator, and \mathcal{R} its (weak-generalized) left inverse (synthesis operator). The final estimate of the Poisson intensity is: $\hat{\mathbf{A}} = \mathbf{y}^{(N_{\max})}$.

Algorithm 1 MS-VST

Require: $a_0 := \mathbf{X}$

Detection

- 1: **for** $j = 1$ to J **do**
- 2: Compute a_j and d_j using (6).
- 3: Compare $|d_j[k]|$ with S_j for each pixel k and update \mathcal{M} .
- 4: **end for**

Estimation

- 5: Initialize $\mathbf{y}^{(0)} = 0$.
 - 6: **for** $k = 0$ to $N_{\max} - 1$ **do**
 - 7: $\mathbf{y}^{(k+1)} = P_+(\mathbf{y}^{(k)} + \mathcal{R}\mathcal{M}\mathcal{W}(\mathbf{X} - \mathbf{y}^{(k)}))$.
 - 8: **end for**
 - 9: Get the estimate $\hat{\mathbf{A}} = \mathbf{y}^{(N_{\max})}$.
-

As seen in the section 2, the reconstruction operator \mathcal{R} can be either the direct reconstruction algorithm (Eq. 1) or the reconstruction with filter bank.⁹ In the first case, reconstructed images have artefacts due to the multi-resolution support and the positivity projection. In the second case, this artefacts are smoothed, but the synthesis filters involve ringing around the most luminous sources. A solution to attenuate this ringing considerably consists in removing the finest wavelet scale. The drawback of this method is that we lose some resolution.

In order to test this method, we simulate a Poisson noise on Gamma Ray galactic background model, without (Fig. 2) and with punctual sources (Fig. 3). Maps are HEALPix images with $n_{\text{side}} = 128$.

4.3 Denoising with Hybrid Steepest Descent method

In order to get a smoother image without degrading the resolution, we propose another algorithm based on Hybrid Steepest Descent (HSD).¹⁰ We formulate the problem as a constrained minimization problem:

$$\min \|\mathcal{W}\mathbf{y}\|_1, \mathbf{y} \geq 0 \text{ and } \forall (j, k) \in \mathcal{M}, (\mathcal{W}\mathbf{y})_j[k] = (\mathcal{W}\mathbf{X})_j[k] \quad (13)$$

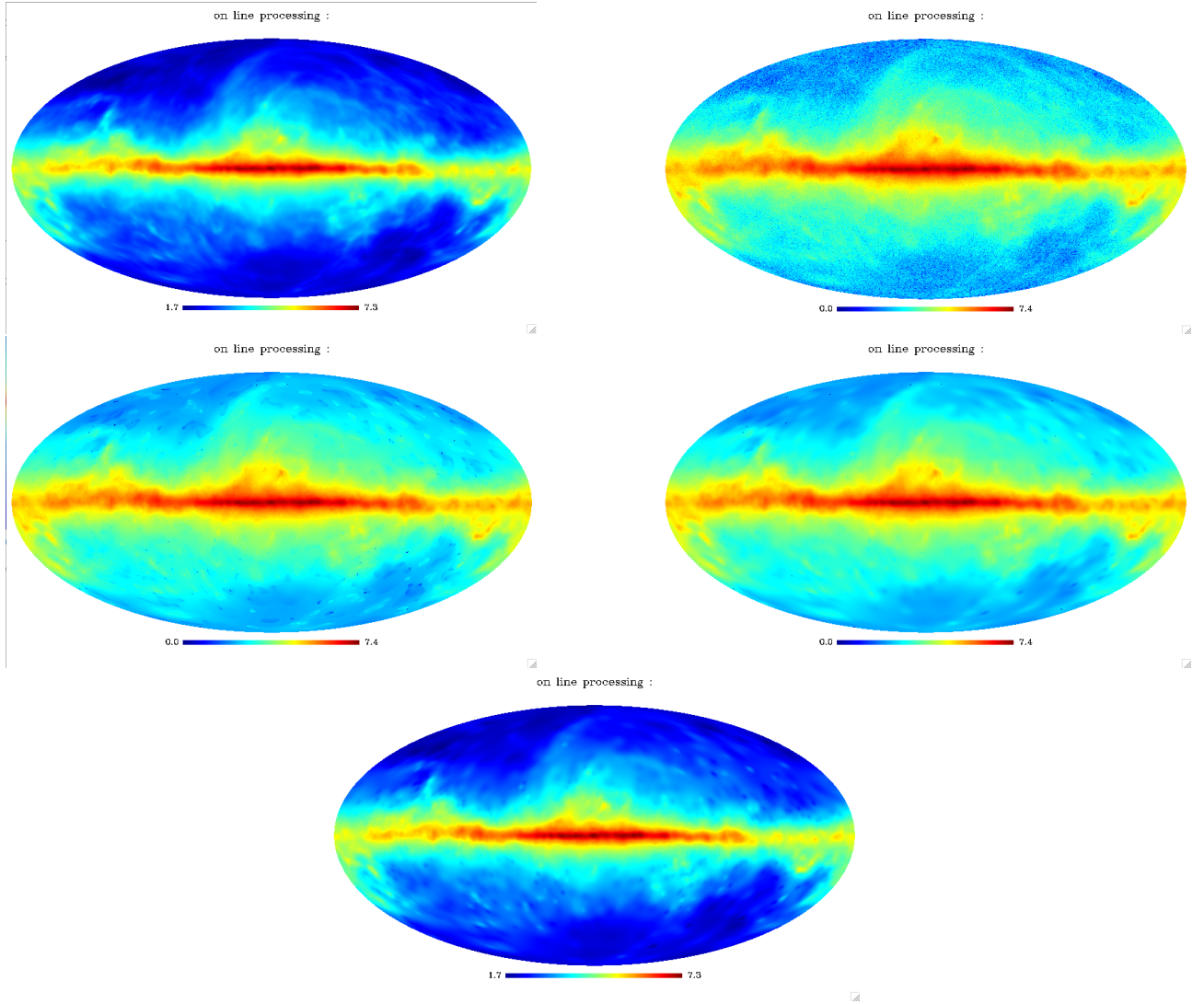


Figure 2. Denoising of a gamma ray galactic background model image with simulated Poisson noise. Pictures are in logarithmic scale. *Top Left*: Image without noise. *Top Right*: Image with simulated Poisson noise. *Middle Left*: Image reconstructed with MS-VSTS method. *Middle Right*: Image reconstructed with MS-VSTS method + filters. *Bottom*: Image reconstructed with MS-VSTS method + filters + suppression of the finest scale.

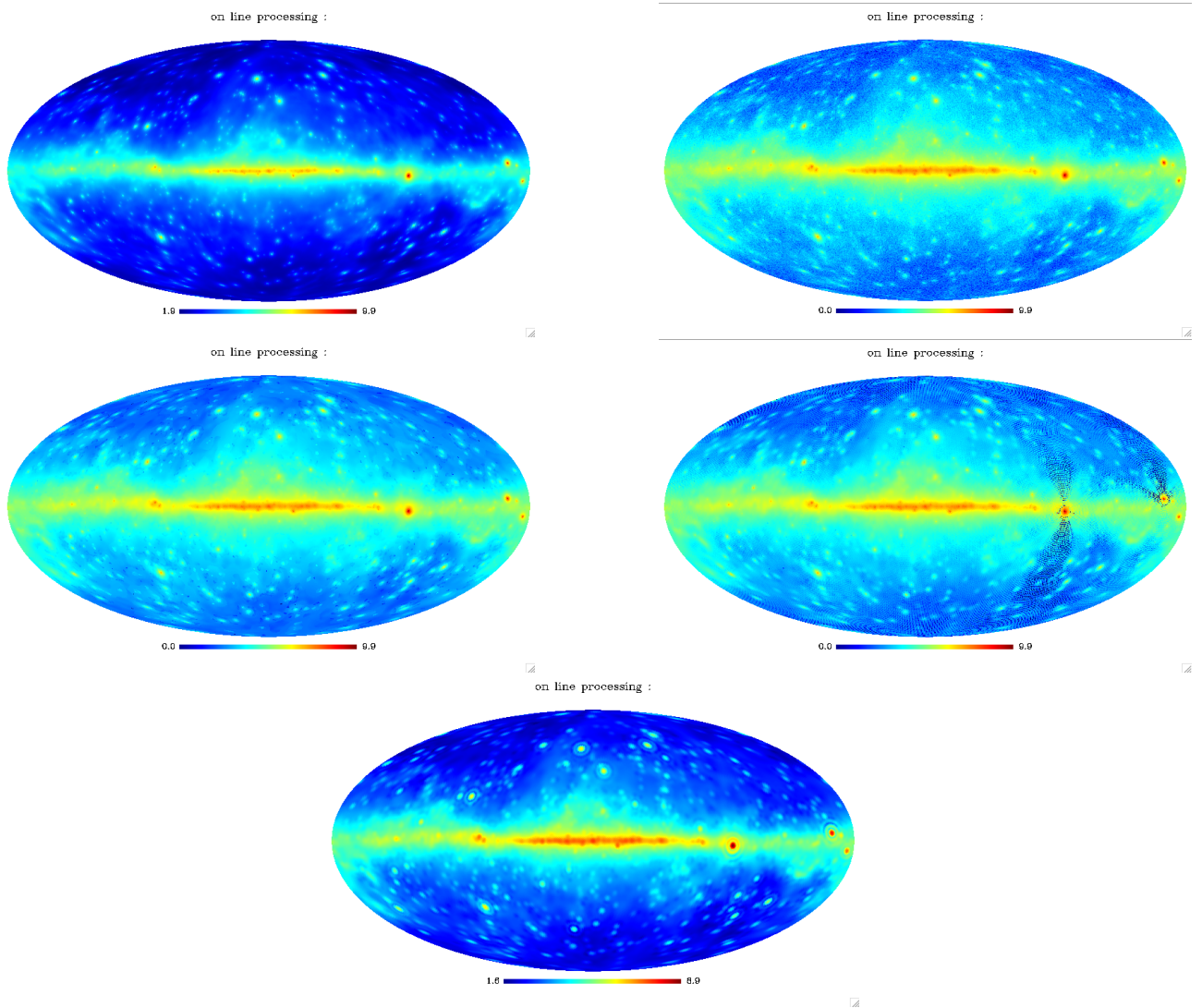


Figure 3. Denoising of a gamma ray galactic background model image with poncutal gamma ray sources with simulated Poisson noise. Pictures are in logarithmic scale. *Top Left*: Image without noise. *Top Right*: Image with simulated Poisson noise. *Middle Left*: Image reconstructed with MS-VSTS method. *Middle Right*: Image reconstructed with MS-VSTS method + filters. *Bottom*: Image reconstructed with MS-VSTS method + filters + suppression of the finest scale.

The only difference with the other algorithm is that we make a soft thresholding on the wavelet coefficients of the image at each iteration. The image is still initialised by $\mathbf{y}^{(0)} = 0$, and the new iteration scheme is, for $k = 0$ to $N_{\max} - 1$:

$$\tilde{\mathbf{y}} = P_+(\mathbf{y}^{(k)} + \mathcal{RMW}(\mathbf{X} - \mathbf{y}^{(k)})) \quad (14)$$

$$\mathbf{y}^{(k+1)} = \mathcal{RST}[\mathcal{W}\tilde{\mathbf{y}}] \quad (15)$$

Let us write $\tilde{\mathbf{d}}_j$ the wavelets coefficients of $\tilde{\mathbf{y}}$ at scale j . For each scale j , the soft thresholding consists in :

$$\text{ST}[\tilde{\mathbf{d}}_j] = \tilde{\mathbf{d}}_j - \beta_k^j \text{sgn}(\tilde{\mathbf{d}}_j) \quad (16)$$

The step sequence β_k^j satisfies:

$$\lim_{k \rightarrow +\infty} \beta_k^j = 0, \sum_{k \geq 1} \beta_k^j = +\infty \text{ and } \sum_{k \geq 1} |\beta_k^j - \beta_{k+1}^j| < +\infty \quad (17)$$

A possible choice of β_k^j is: $\beta_k^j = \sigma_{(j)} \frac{N_{\max} - k}{N_{\max} - 1}, k = 1, 2, \dots, N_{\max}$.

Algorithm 2 MS-VST + HSD

Require: $a_0 := \mathbf{X}$

Detection

- 1: **for** $j = 1$ to J **do**
- 2: Compute a_j and d_j using (6).
- 3: Compare $|d_j[k]|$ with S_j for each pixel k and update \mathcal{M} .
- 4: **end for**

Estimation

- 5: Initialize $\mathbf{y}^{(0)} = 0$.
 - 6: **for** $k = 0$ to $N_{\max} - 1$ **do**
 - 7: $\tilde{\mathbf{y}} = P_+(\mathbf{y}^{(k)} + \mathcal{RMW}(\mathbf{X} - \mathbf{y}^{(k)}))$.
 - 8: $\mathbf{y}^{(k+1)} = \mathcal{RST}[\mathcal{W}\tilde{\mathbf{y}}]$.
 - 9: **end for**
 - 10: Get the estimate $\hat{\mathbf{A}} = \mathbf{y}^{(N_{\max})}$.
-

Denoised images are much smoother with the HSD algorithm, whatever the choice of the reconstruction operator \mathcal{R} (Fig. 4).

5. BACKGROUND EXTRACTION

In some cases, it may be useful to be able to remove a background model from the Poisson data. In the case of Fermi data, the presence of the Milky Way diffuse background makes the detection of punctual sources more difficult. A model of the diffuse background is obtained thanks to infra-red galactic gas and dust maps. We introduce a background removal in our denoising algorithm.

We note \mathbf{X} the data, \mathbf{B} the background we want to remove, and $d_j^{(b)}[k]$ the MS-VSTS coefficients of \mathbf{B} at scale j and position k . We determine the multi-resolution support by comparing $|d_j[k] - d_j^{(b)}[k]|$ with S_j . Then, the reconstruction algorithm step is: $\mathbf{y}^{(k+1)} = P_+(\mathbf{y}^{(k)} + \mathcal{RMW}(\mathbf{X} - \mathbf{B} - \mathbf{y}^{(k)}))$.

We made a theoretical study where the background is a Poisson realisation whose intensity map is a sum of two Gaussians of standard deviation equal to 0.1 and 0.01 respectively, and the useful signal is a Poisson realisation whose intensity map is a Gaussian of standard deviation equal to 0.01. The method works on this theoretical case. The method was tested on simulated Fermi data with Milky Way diffuse background (Fig. 6).

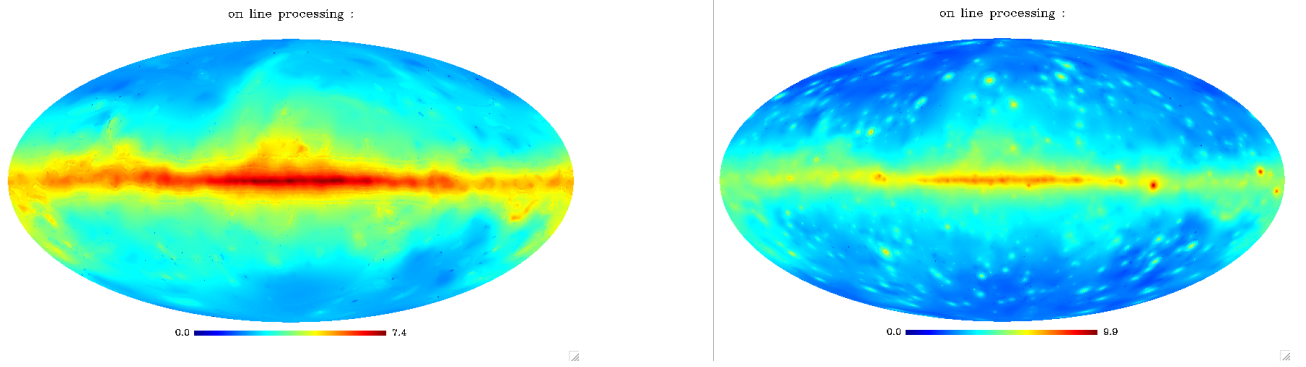


Figure 4. *Left*: Gamma Ray galactic background model reconstructed with MS-VSTS method + Hybrid Steepest Descent Algorithm in logarithmic scale *Right*: Gamma Ray galactic background model with ponctual sources reconstructed with MS-VSTS method + Hybrid Steepest Descent Algorithm in logarithmic scale

Algorithm 3 MS-VSTS + HSD + Background extraction

Require: $a_0 := \mathbf{X}$

Detection

- 1: **for** $j = 1$ to J **do**
- 2: Compute a_j and d_j using (6).
- 3: Compare $|d_j[k] - d_j^{(b)}[k]|$ with S_j for each pixel k and update \mathcal{M} .
- 4: **end for**

Estimation

- 5: Initialize $\mathbf{y}^{(0)} = 0$.
 - 6: **for** $k = 0$ to $N_{\max} - 1$ **do**
 - 7: $\hat{\mathbf{y}} = P_+(\mathbf{y}^{(k)} + \mathcal{RMW}(\mathbf{X} - \mathbf{B} - \mathbf{y}^{(k)}))$.
 - 8: $\mathbf{y}^{(k+1)} = \mathcal{RST}[\mathcal{W}\hat{\mathbf{y}}]$.
 - 9: **end for**
 - 10: Get the estimate $\hat{\mathbf{A}} = \mathbf{y}^{(N_{\max})}$.
-

6. CONCLUSION

This article presented a new method of Poisson noise removal on spherical maps called MS-VSTS, adapted from the MS-VST method developed by Zhang on 2D fluorescence microscopy images. This method is the first one to have been designed for spherical data. Before, the only solution to denoise spherical Poisson data was projecting the spherical data on several plans (e.g. faces of the HEALPix representation), applying a 2D Poisson data denoising method on each plan, and then reconstructing the denoised spherical map, which led to some visual artefacts. This method proved to be very effective for simulated Fermi data denoising. This method, presented here with the isotropic undecimated wavelet transform on the sphere, can be applied with a wide range of multi-scales transforms on the sphere like ridgelets or curvelets. Consequently, our method is particularly flexible, as we are allowed to choose the most adaptative transform according to the morphology of the Poisson data set. Our algorithm may also be extended to perform background extraction in addition to Poisson noise removal.

ACKNOWLEDGMENTS

Our results were obtained using the HEALPix pixellisation (by Górsky, Hivon and Wandelt, 1999) and the MRS environment (by Moudden, Starck, Abrial and Cardoso).

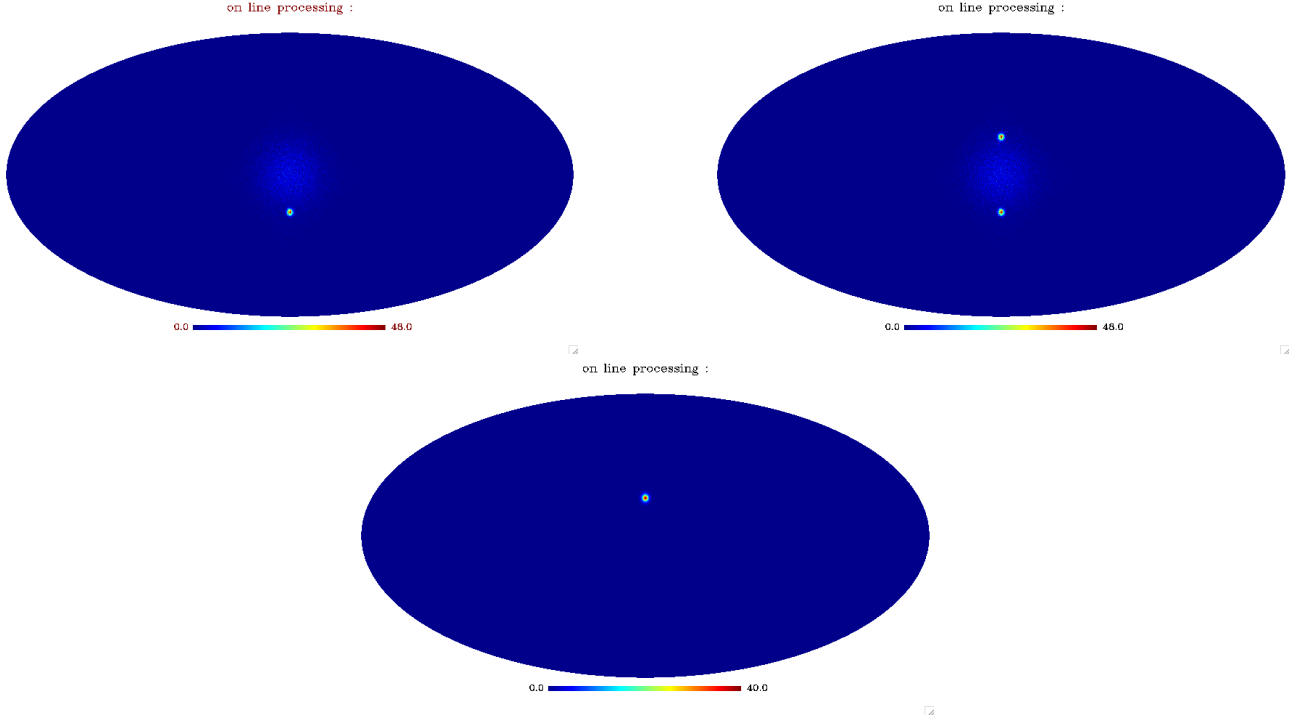


Figure 5. Theoretical experience for background extraction. *Top Left*: Background: Poisson realisation whose intensity map is a sum of two Gaussians of standard deviation equal to 0.1 and 0.01 respectively. *Top Right*: Image: the useful signal is a Poisson realisation whose intensity map is a Gaussian of standard deviation equal to 0.01. *Bottom*: Image denoised with MS-VSTS method + Hybrid Steepest Descent and extraction of background model.

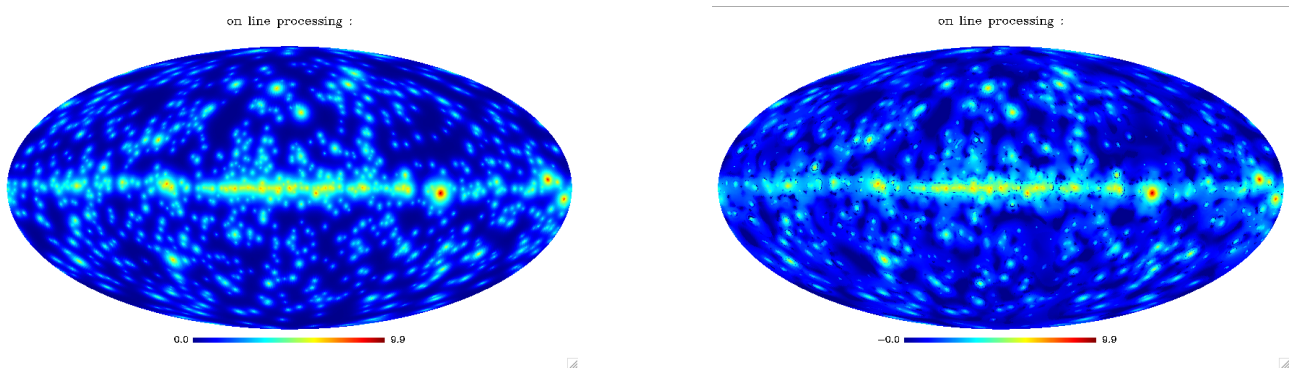


Figure 6. *Top Left*: Theoretical Gamma Ray sources in logarithmic scale. *Top Right*: Simulated Fermi map denoised with Hybrid Steepest Descent and extraction of background model.

REFERENCES

- [1] Timmerman, K. and Nowak, R., “Multiscale modeling and estimation of poisson processes with application to photon-limited imaging,” *IEEE Trans. Inf. Theory* **vol. 45**, **no. 3**, pp. 846–862 (1999).
- [2] Kolaczyk, E., “Bayesian multiscale models for poisson processes,” *J. Amer. Stat. Assoc.* **vol. 94**, **no. 447**, pp. 920–933 (1999).
- [3] Lefkimmiaits, S., Maragos, P., and Papandreou, G., “Bayesian inference on multiscale models for poisson intensity estimation: Applications to photon-limited image denoising,” *IEEE Transactions on Image Processing* **vol. 20**, **no. 20** (january 2009).
- [4] Anscombe, F., “The transformation of poisson, binomial and negative-binomial data,” *Biometrika* **vol. 35**, **no. 3**, pp. 246–254 (1948).
- [5] Fisz, M., “The limiting distribution of a function of two independant random variables and its statistical application,” *Colloquium Mathematicum* **vol. 3**, pp. 138–146 (1955).
- [6] Donoho, D., “Nonlinear wavelet methodes for recovery of signals, densitites, and spectra from indirect and noisy data,” *Proc. Symposia in Applied Mathematics* **vol. 47**, pp. 173–205 (1993).
- [7] Fryźlewicz, P. and Nason, G., “A haar-fisz algorithm for poisson intensity estimation,” *J. Comp. Graph. Stat.* **vol. 13**, pp. 621–638 (2004).
- [8] Zhang, B., Fadili, J., and Starck, J.-L., “Wavelets, ridgelets and curvelets for poisson noise removal,” *IEEE Transactions on Image Processing* **vol. 11**, **no. 6**, pp. 1093–1108 (january 2009).
- [9] Abrial, P., Moudden, Y., Starck, J., Bobin, J., Fadili, J., Afeyan, B., and Nguyen, M., “Morphological component analysis and inpainting on the sphere: Application in physics and astrophysics,” *Journal of Fourier Analysis and Applications (JFAA)* (2007).
- [10] Yamada, I., “The hybrid steepest descent method for the variational inequality problem over the intersection of fixed point sets of nonexpansive mappings,” in [*Inherently Parallel Algorithm in Feasibility and Optimization and their Applications*], pp. 473–504, Elsevier (2001).