Sparsity, Euclid and the SKA

Jason McEwen

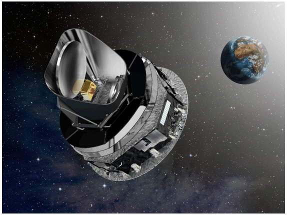
http://www.jasonmcewen.org/

Mullard Space Science Laboratory (MSSL)
University College London (UCL)

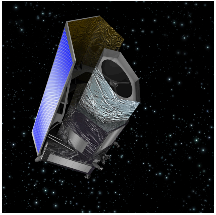
Synergistic Science with Euclid and the Square Kilometre Array
Oxford, September 2013

Big cosmology: big science, big data and big algos

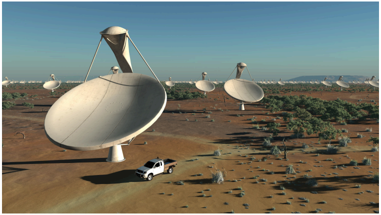
- A new era of **big cosmology** is emerging.
  - Planck: full-sky observations of the CMB at unprecedented resolution, sensitivity and frequency coverage.
  - Euclid: unprecedented survey of billion galaxies over more than one third of the sky.
  - Square Kilometre Array (SKA): sensitivity 50x that of previous radio telescopes with phenomenal data rates.
  - Others. . .



(a) Planck



(b) Euclid



(c) SKA

- New instruments must be complemented with novel analyses methodologies to extract new science from big data-sets

$\rightarrow$ sparsity.

Outline

Outline

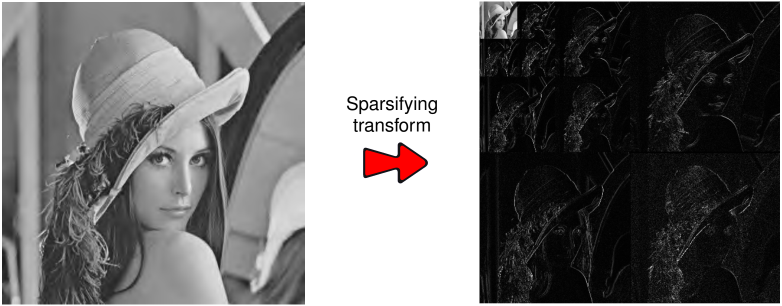
What is sparsity?

What is sparsity?

— representation of data in such a way that many data points are zero.

What is sparsity?

What is sparsity?

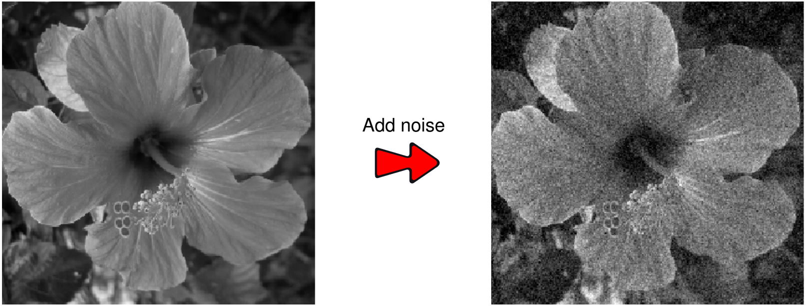


Why is sparsity useful?

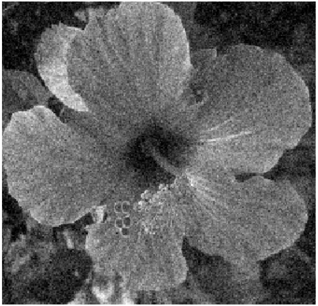
Why is sparsity useful?

— efficient characterisation of information.

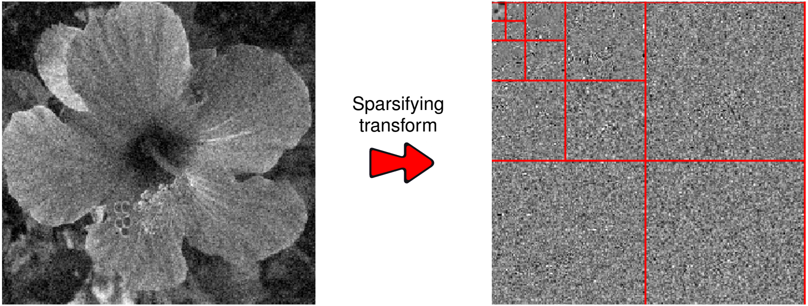
Why is sparsity useful?



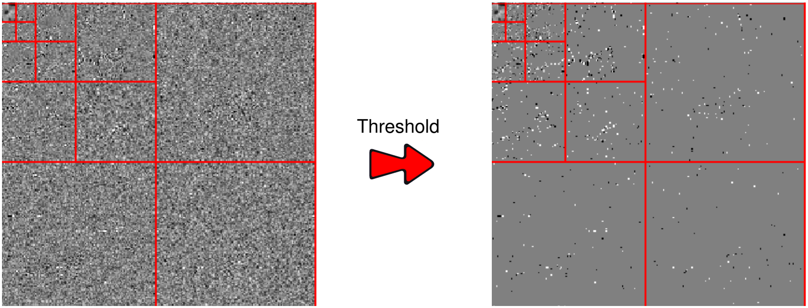
Why is sparsity useful?



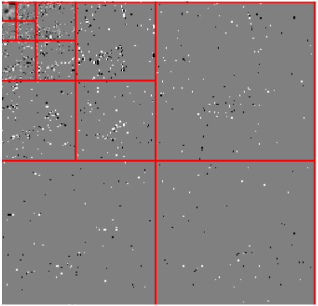
Why is sparsity useful?



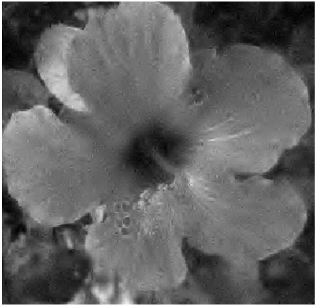
Why is sparsity useful?



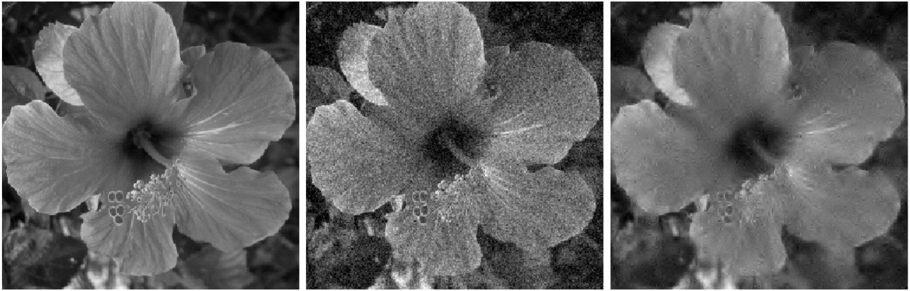
Why is sparsity useful?



Inverse transform



Why is sparsity useful?



(a) Original (b) Noisy (c) Denoised

[Credit: `http://www.ceremade.dauphine.fr/~peyre/numerical-tour/tours/denoisingwav_2_wavelet_2d/`]

Outline

How can we construct sparsifying transforms?

How can we construct sparsifying transforms?

— many signals in nature have spatially localised, scale-dependent features.

How can we construct sparsifying transforms?

Fourier (1807)

Haar (1909)

Morlet and Grossman (1981)

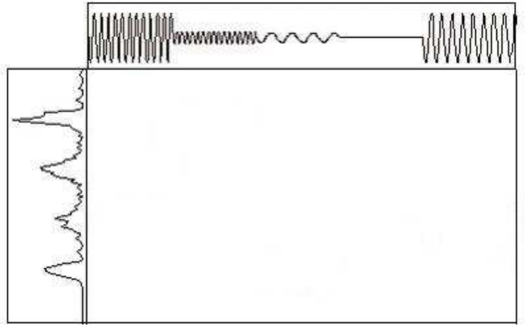


Figure: Fourier vs wavelet transform [Credit: `http://www.wavelet.org/tutorial/`]

How can we construct sparsifying transforms?

Fourier (1807)

Haar (1909)

Morlet and Grossman (1981)

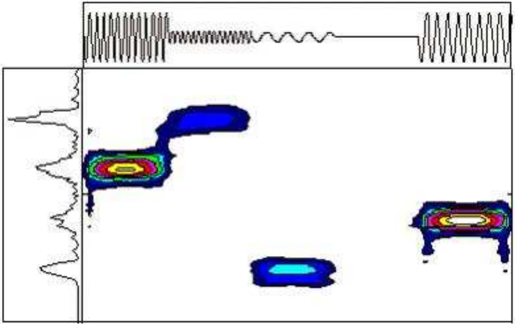


Figure: Fourier vs wavelet transform [Credit: http://www.wavelet.org/tutorial/]

How can we construct sparsifying transforms?

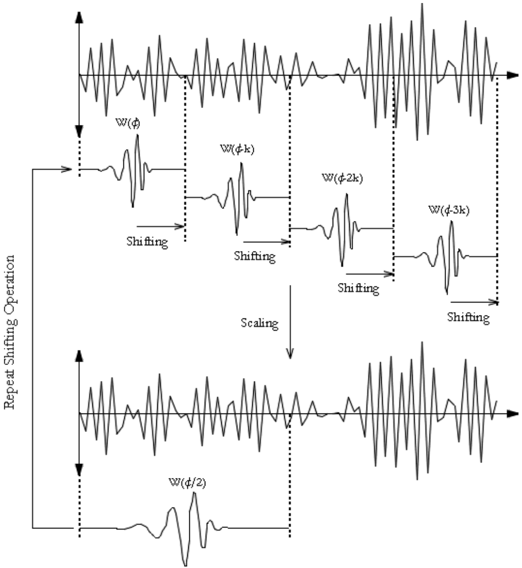
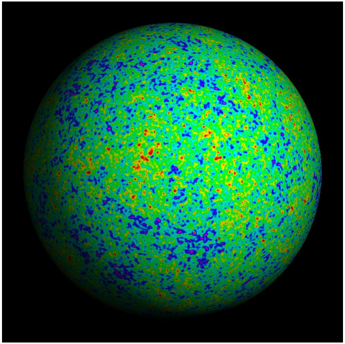


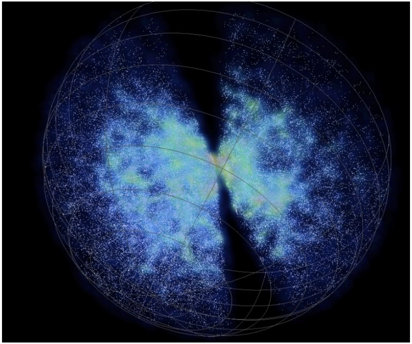
Figure: Wavelet scaling and shifting [Credit: `http://www.wavelet.org/tutorial/`]

Cosmological observations made on celestial sphere

- Cosmological observations are inherently made on the celestial sphere.
    - Observations of the cosmic microwave background (CMB) are made on the sphere.
    - Observations tracing the large-scale structure (LSS) are made on the ball.



(a) CMB (WMAP)



(b) Galaxy survey (SDSS)

Scale-discretised wavelets on the sphere

- *Exact reconstruction with directional wavelets on the sphere*
  Wiaux, McEwen, Vandergheynst, Blanc (2008) [arXiv:0712.3519]
- Alternatives: isotropic wavelets, pyramidal wavelets, ridgelets, curvelets (Starck *et al.* 2006); needlets (Narcowich *et al.* 2006, Baldi *et al.* 2009, Marinucci *et al.* 2008).

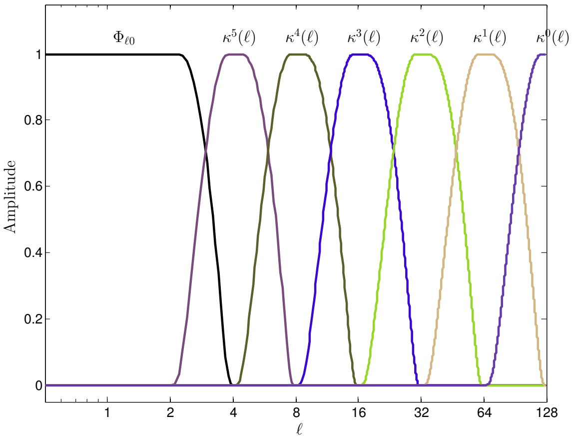


Figure: Harmonic tiling on the sphere.

Scale-discretised wavelets on the sphere

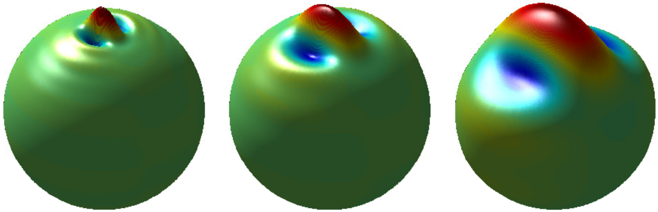

(a) $j = 4$ (b) $j = 3$ (c) $j = 2$

Figure: Scale-discretised wavelets on the sphere.

- The scale-discretised wavelet transform is given by the usual projection onto each wavelet:

$$W^{\Psi^j}(\rho) \equiv (f \star \Psi^j)(\rho) = \langle f,\, \mathcal{R}_\rho \Psi^j \rangle = \int_{\mathbb{S}^2} \mathrm{d}\Omega(\omega) f(\omega) (\mathcal{R}_\rho \Psi^j)^*(\omega)\ ,$$

- The original function may be recovered exactly in practice from the wavelet (and scaling) coefficients:

$$f(\omega) = 2\pi \int_{\mathbb{S}^2} \mathrm{d}\Omega(\omega') W^{\Phi}(\omega') (\mathcal{R}_{\omega'} L^{\mathrm{d}} \Phi)(\omega) + \sum_{j=0}^{J} \int_{\mathrm{SO}(3)} \mathrm{d}\varrho(\rho) W^{\Psi^j}(\rho) (\mathcal{R}_\rho L^{\mathrm{d}} \Psi^j)(\omega)\ .$$

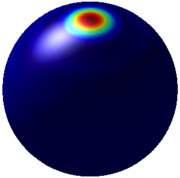
Codes for scale-discretised wavelets on the sphere

S2DW code

`http://www.s2dw.org`

Exact reconstruction with directional wavelets on the sphere
Wiaux, McEwen, Vandergheynst, Blanc (2008) [arXiv:0712.3519]

- Fortran
- Parallelised
- Supports directional, steerable wavelets



S2LET code

`http://www.s2let.org`

S2LET: A code to perform fast wavelet analysis on the sphere
Leistedt, McEwen, Vandergheynst, Wiaux (2012) [arXiv:1211.1680]

- C, Matlab, IDL, Java
- Supports only axisymmetric wavelets at present
- Future extensions:
  - Directional, steerable wavelets
  - Faster algorithms to perform wavelet transforms
  - Spin wavelets

Fourier-LAGuerre wavelets (flaglets) on the ball

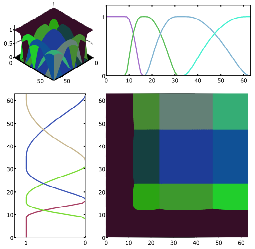


Figure: Tiling of Fourier-Laguerre space.

- *Exact wavelets on the ball*
  Leistedt & McEwen (2012) [arXiv:1205.0792]
- Extend scale-discretised wavelets on the sphere to the ball.
- Some subtleties (define translation and convolution on the radial line).
- Construct wavelets by tiling the ℓ–p harmonic plane.

- The Fourier-Laguerre wavelet transform is given by the usual projection onto each wavelet:

$$W^{\Psi^{jj'}}(\boldsymbol{r}) \equiv (f \star \Psi^{jj'})(\boldsymbol{r}) = \langle f | \mathcal{T}_{\boldsymbol{r}} \Psi^{jj'} \rangle_{\mathbb{B}^3} = \int_{B^3} \mathrm{d}^3\boldsymbol{r}' f(\boldsymbol{r}') (\mathcal{T}_{\boldsymbol{r}} \Psi^{jj'})(\boldsymbol{r}') \,.$$

- The original function may be synthesised exactly in practice from its wavelet (and scaling) coefficients:

$$f(\boldsymbol{r}) = \int_{B^3} \mathrm{d}^3\boldsymbol{r}' W^{\Phi}(\boldsymbol{r}') (\mathcal{T}_{\boldsymbol{r}} \Phi)(\boldsymbol{r}') + \sum_{j=J_0}^{J} \sum_{j'=J_0'}^{J'} \int_{B^3} \mathrm{d}^3\boldsymbol{r}' W^{\Psi^{jj'}}(\boldsymbol{r}') (\mathcal{T}_{\boldsymbol{r}} \Psi^{jj'})(\boldsymbol{r}') \,.$$

- Alternatives: Spherical 3D isotropic wavelets (Lanusse, Rassat & Starck 2012)

Fourier-LAGuerre wavelets (flaglets) on the ball

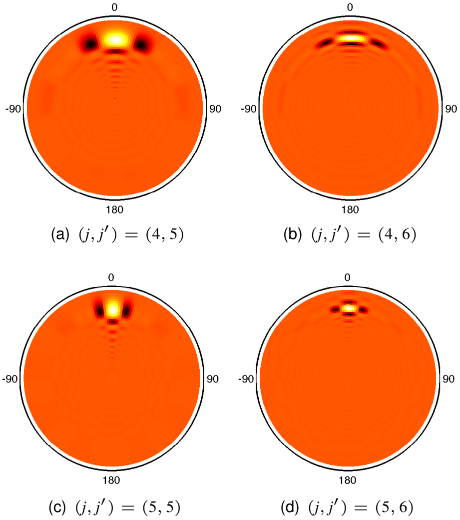
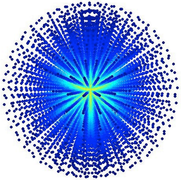


Figure: Scale-discretised wavelets on the ball.

Codes for Fourier-LAGuerre wavelets (flaglets) on the ball

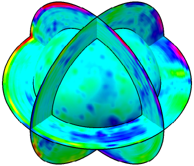


FLAG code: Fourier-Laguerre transform

`http://www.flaglets.org`

Exact wavelets on the ball
Leistedt & McEwen (2012) [arXiv:1205.0792]

- C, Matlab, IDL, Java
- Exact Fourier-LAGuerre transform on the ball



FLAGLET code: Fourier-Laguerre wavelets

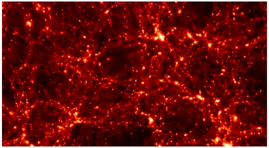
`http://www.flaglets.org`

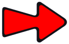
Exact wavelets on the ball
Leistedt & McEwen (2012) [arXiv:1205.0792]

- C, Matlab, IDL, Java
- Exact (Fourier-LAGuerre) wavelets on the ball – coined *flaglets*!

Large-scale structure (LSS) of the Universe

- Map Horizon simulation of large-scale structure (LSS) to Fourier-Laguerre sampling.





LSS fly through

Flaglet void finding

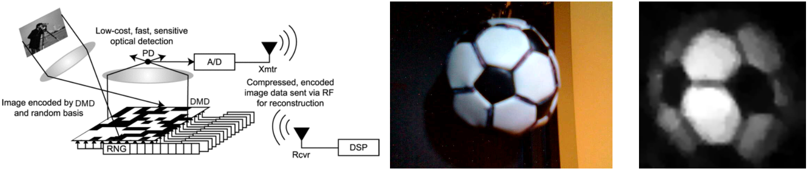
- Find voids in the large-scale structure (LSS) of the Universe.
- Perform Alcock & Paczynski (1979) test: study void shapes to constrain the nature of dark energy (*e.g.* Sutter *et al.* 2012).

LSS voids

Outline

Compressive sensing

- Next evolution of wavelet analysis $\rightarrow$ wavelets are a key ingredient.
- The mystery of JPEG compression (discrete cosine transform; wavelet transform).
- Move compression to the acquisition stage $\rightarrow$ compressive sensing.
- Deep mathematical foundation (Candes *et al.* 2006, Donoho 2006).



(a) Architecture (b) Scene (c) Recov. (20% meas.)

Figure: Single pixel camera

Compressive sensing

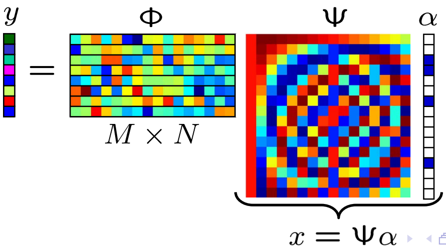
- Linear operator (algebra) representation of signal decomposition (into *atoms* of a *dictionary*):

$$x(t) = \sum_i \alpha_i \Psi_i(t) \quad \rightarrow \quad \boldsymbol{x} = \sum_i \Psi_i \alpha_i = \begin{pmatrix} | \\ \Psi_0 \\ | \end{pmatrix} \alpha_0 + \begin{pmatrix} | \\ \Psi_1 \\ | \end{pmatrix} \alpha_1 + \cdots \quad \rightarrow \quad \boxed{\boldsymbol{x} = \Psi \boldsymbol{\alpha}}$$

- Linear operator (algebra) representation of measurement:

$$y_i = \langle x, \Phi_j \rangle \quad \rightarrow \quad \boldsymbol{y} = \begin{pmatrix} - & \Phi_0 & - \\ - & \Phi_1 & - \\ & \vdots & \end{pmatrix} \boldsymbol{x} \quad \rightarrow \quad \boxed{\boldsymbol{y} = \Phi \boldsymbol{x}}$$

- Putting it together: $\boxed{\boldsymbol{y} = \Phi \boldsymbol{x} = \Phi \Psi \boldsymbol{\alpha}}$



Compressive sensing

- Ill-posed inverse problem:

$$\boldsymbol{y} = \Phi \boldsymbol{x} + \boldsymbol{n} = \Phi \Psi \boldsymbol{\alpha} + \boldsymbol{n}.$$

- Solve by imposing a regularising prior that the signal to be recovered is sparse in Ψ, *i.e.* solve the following ℓ_0 optimisation problem:

$$\boldsymbol{\alpha}^\star = \arg\min_{\boldsymbol{\alpha}} \|\alpha\|_0 \text{ such that } \|\boldsymbol{y} - \Phi\Psi\boldsymbol{\alpha}\|_2 \leq \epsilon ,$$

where the signal is synthesising by $\boldsymbol{x}^\star = \Psi \boldsymbol{\alpha}^\star$.

- Recall norms given by:

$$\|\alpha\|_0 = \text{no. non-zero elements} \qquad \|\alpha\|_1 = \sum_i |\alpha_i| \qquad \|\alpha\|_2 = \Big(\sum_i |\alpha_i|^2\Big)^{1/2}$$

- Solving this problem is difficult (combinatorial).
- Instead, solve the ℓ_1 optimisation problem (convex):

$$\boldsymbol{\alpha}^\star = \arg\min_{\boldsymbol{\alpha}} \|\alpha\|_1 \text{ such that } \|\boldsymbol{y} - \Phi\Psi\boldsymbol{\alpha}\|_2 \leq \epsilon .$$

Compressive sensing

- Ill-posed inverse problem:

$$\boldsymbol{y} = \Phi \boldsymbol{x} + \boldsymbol{n} = \Phi \Psi \boldsymbol{\alpha} + \boldsymbol{n}.$$

- Solve by imposing a regularising prior that the signal to be recovered is sparse in Ψ, *i.e.* solve the following ℓ_0 optimisation problem:

$$\boldsymbol{\alpha}^\star = \arg\min_{\boldsymbol{\alpha}} \|\alpha\|_0 \text{ such that } \|\boldsymbol{y} - \Phi\Psi\boldsymbol{\alpha}\|_2 \leq \epsilon \,,$$

where the signal is synthesising by $\boldsymbol{x}^\star = \Psi \boldsymbol{\alpha}^\star$.

- Recall norms given by:

$$\|\alpha\|_0 = \text{no. non-zero elements} \qquad \|\alpha\|_1 = \sum_i |\alpha_i| \qquad \|\alpha\|_2 = \Big(\sum_i |\alpha_i|^2\Big)^{1/2}$$

- Solving this problem is difficult (combinatorial).

- Instead, solve the ℓ_1 optimisation problem (convex):

$$\boldsymbol{\alpha}^\star = \arg\min_{\boldsymbol{\alpha}} \|\alpha\|_1 \text{ such that } \|\boldsymbol{y} - \Phi\Psi\boldsymbol{\alpha}\|_2 \leq \epsilon \,.$$

Compressive sensing

- The solutions of the ℓ_0 and ℓ_1 problems are often the same.
- Geometry of ℓ_2 and ℓ_1 problems.

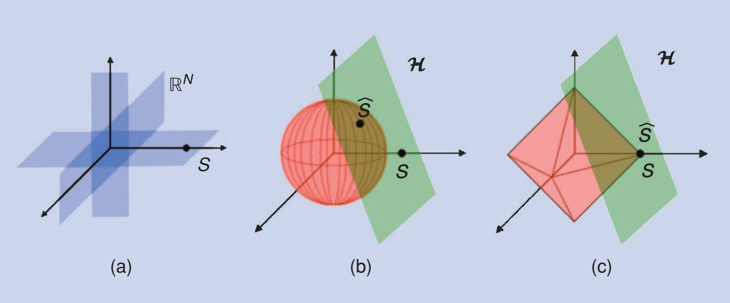


Figure: Geometry of (a) ℓ_0 (b) ℓ_2 and (c) ℓ_1 problems. [Credit: Baraniuk (2007)]

Compressive sensing

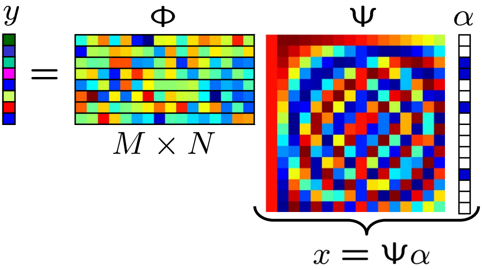
- In the absence of noise, compressed sensing is **exact!**
- Number of measurements required to achieve exact reconstruction is given by

$$M \geq c\mu^2 K \log N \; ,$$

where K is the sparsity and N the dimensionality.

- The coherence between the measurement and sparsity basis is given by

$$\mu = \sqrt{N} \max_{i,j} |\langle \Psi_i , \Phi_j \rangle| \; .$$



- Robust to noise.

Radio interferometric inverse problem

- Consider the ill-posed inverse problem of radio interferometric imaging:

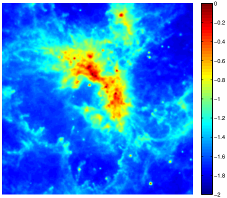
$$\boxed{\boldsymbol{y} = \Phi \boldsymbol{x} + \boldsymbol{n}} \, ,$$

where $\boldsymbol{y}$ are the measured visibilities, Φ_{p} is the linear measurement operator, $\boldsymbol{x}_{\mathrm{p}}$ is the underlying image and $\boldsymbol{n}$ is instrumental noise.

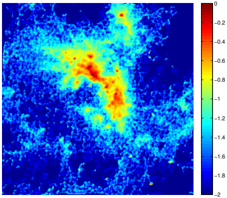
- Measurement operator $\boxed{\Phi = \mathbf{M}\,\mathbf{F}\,\mathbf{C}\,\mathbf{A}}$ may incorporate:
  - primary beam $\mathbf{A}$ of the telescope;
  - w-component modulation $\mathbf{C}$ (responsible for the spread spectrum phenomenon);
  - Fourier transform $\mathbf{F}$;
  - masking $\mathbf{M}$ which encodes the incomplete measurements taken by the interferometer.

Radio interferometric imaging

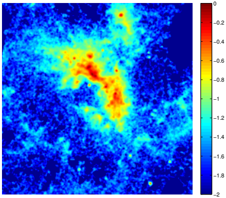
- SARA algorithm for radio interferometric imaging, building on compressive sensing techniques (Carrillo, McEwen & Wiaux 2012) [arXiv:1205.3123].



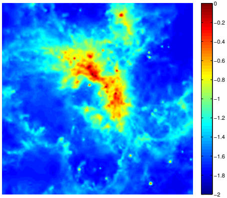
(a) Original



(b) “CLEAN”



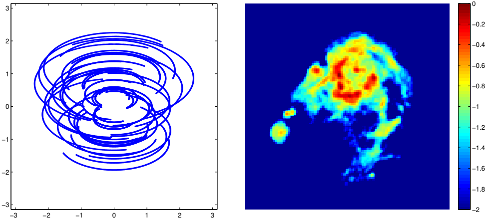
(c) “MS-CLEAN”



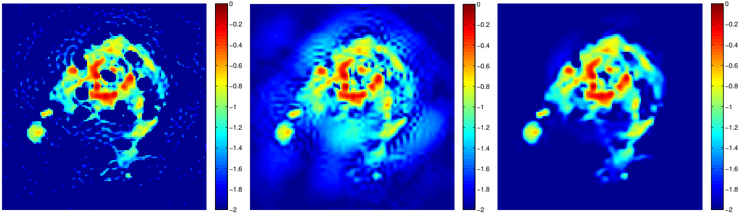
(d) SARA

Continuous visibilities

- PURIFY: realistic radio interferometric imaging with compressive sensing (Carrillo, McEwen & Wiaux 2013) [arXiv:1307.4370].
  http://basp-group.github.io/purify/



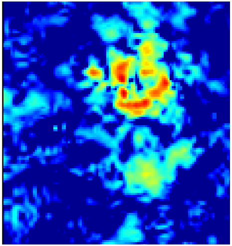
(a) Coverage (b) Original



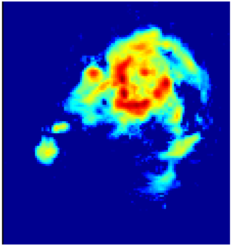
(c) "CLEAN" (d) "MS-CLEAN" (e) SARA

Wide field-of-view

- Wide fields give rise to the spread spectrum effect (Wiaux *et al.* 2009), which improves reconstruction quality.
- Recently studied in a more realistic setting (Wolz, McEwen, Abdalla, Carrillo, Wiaux 2013) [arXiv:1307.3424].



(a) No spread spectrum

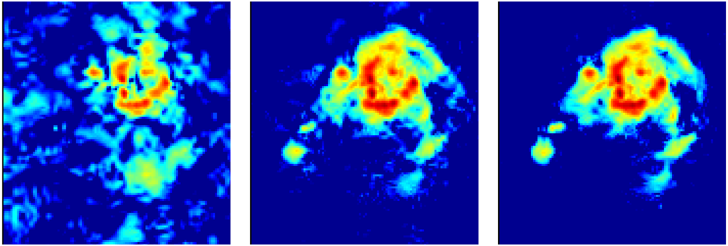


(c) Idealised spread spectrum

Figure: Reconstruction fidelity in the presence and absence of the spread spectrum effect.

Wide field-of-view

- Wide fields give rise to the spread spectrum effect (Wiaux *et al.* 2009), which improves reconstruction quality.
- Recently studied in a more realistic setting (Wolz, McEwen, Abdalla, Carrillo, Wiaux 2013) [arXiv:1307.3424].



(a) No spread spectrum (b) More realistic spread spectrum (c) Idealised spread spectrum

Figure: Reconstruction fidelity in the presence and absence of the spread spectrum effect.

Summary

For big cosmology we need novel analysis methods to deal with the data deluge of forthcoming experiments (*e.g.* Euclid, SKA, . . .)

$\rightarrow$ exploit sparsity.