Fast Scheme for Approximating an Offset PSF Response

M. Atemkeng
Supervisors: Prof O.Smirnov, Dr C. Tasse, Dr G. Foster

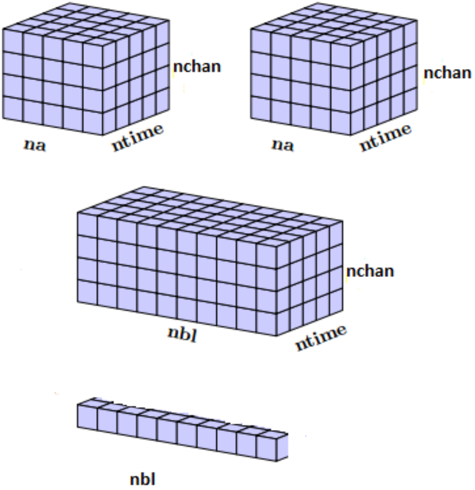
Rhodes University, Faculty of Science

SAIP

July 2, 2015

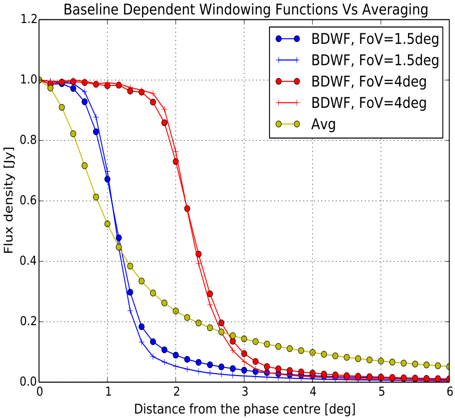
Problem statement: decorrelation

- Visibility averaging
- Time and bandwidth decorrelation
- Smearing = loss of amplitude + distortion



Problem statement: decorrelation

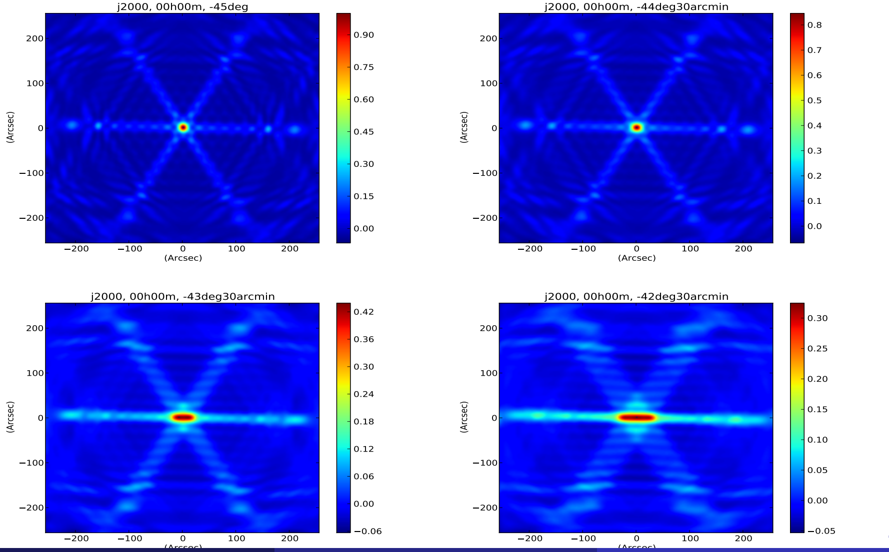
- Visibility averaging
- Time and bandwidth smearing
- Smearing = loss of amplitude + distortion



JVLA C, 100s integration and 10MHz channels width

Averaging effects or smearing

- Phase center RA=0, Dec=-45 deg



Problem solving: PSF slow derivation

- interferometric visibility

$$V(\mathbf{b}) = \int_{\Omega} I_{\nu}(\mathbf{s}) e^{-2\pi i \mathbf{bs}} d\Omega, \tag{1}$$

- $t\nu$-space: re-sampling time Δt and frequency $\Delta\nu$, t_0 and ν_0 their centre respectively

$$V(t_0, \nu_0) = \iint\limits_{\Delta t \Delta\nu} W_{t_0,\nu_0}(t,\nu) \left[\int\limits_{\Omega} I_{\nu}(\mathbf{s}) e^{-2\pi i \mathbf{bs}} d\Omega \right] dt d\nu \tag{2}$$

$$= \iint\limits_{u_{t\nu} v_{t\nu}} W_{\mathbf{b}_0}(\mathbf{b}) \left[\int\limits_{\Omega} I_{\nu}(\mathbf{s}) e^{-2\pi i \mathbf{bs}} d\Omega \right] du_{t\nu} dv_{t\nu} \tag{3}$$

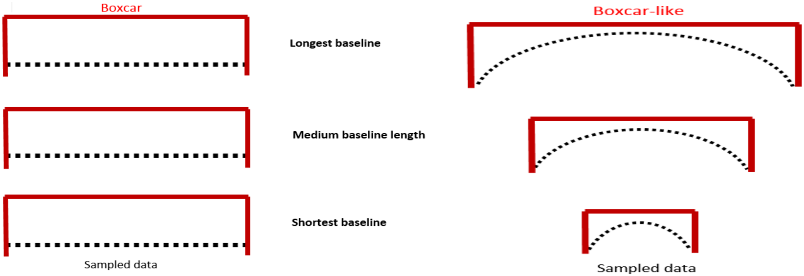
- Generalized weighting and windowing function

$$W_{t_0,\nu_0}(t,\nu) = \Pi(t - t_0, \nu - \nu_0) W(t - t_0, \nu - \nu_0) \tag{4}$$

- Boxcar and boxcar-like

$$\Pi(t - t_0, \nu - \nu_0) = \begin{cases} \frac{1}{\Delta t \Delta \nu}, & |t| \leq \Delta t/2, \quad |\nu| \leq \Delta \nu/2 \\ 0, & \text{otherwise}, \end{cases} \quad (5)$$

$$\Pi(\mathbf{b} - \mathbf{b}_0) = \begin{cases} \frac{1}{\Delta u_{t\nu} \Delta v_{t\nu}}, & |u_{t\nu}| \leq \Delta u_{t\nu}/2, \quad |v_{t\nu}| \leq \Delta v_{t\nu}/2 \\ 0, & \text{otherwise}, \end{cases} \quad (6)$$



PSF slow derivation

$$V_{\mathbf{b}_0}(\mathbf{x}) = \iint\limits_{u_{t\nu}\, v_{t\nu}} W_{\mathbf{b}_0}(\mathbf{b}) e^{-2\pi i (u_{t\nu} l + v_{t\nu} m + w_{t\nu}(n-1))} du_{t\nu} dv_{t\nu}. \quad (7)$$

- source coordinates, l, m and n. Average baseline vector, $\mathbf{x}$

$$V_{\mathbf{b}_0}(\mathbf{x}) = \left(\widetilde{G}_{\mathbf{b}} \circ \left[C_{\mathbf{b}_0} \widetilde{\Pi}_{\mathbf{b}} \right] \circ \left[C_{\mathbf{b}_0} \widetilde{W_{\mathbf{b}}} \right] \right)_{(l,m)} \quad (8)$$

$$C_{\mathbf{b}_0}(l, m) = e^{-2\pi i (u_{t_0\nu_0} l + v_{t_0\nu_0} m)}, \quad G_{\mathbf{b}}(l, m) = e^{-2\pi i [w_{t\nu}(n-1)]} \quad (9)$$

Computational load

$$V_{\mathbf{b}_0}(\mathbf{x}) = \left(\widetilde{G}_{\mathbf{b}} \circ \left[C_{\mathbf{b}_0} \widetilde{\Pi}_{\mathbf{b}} \right] \circ \left[C_{\mathbf{b}_0} \widetilde{W}_{\mathbf{b}} \right] \right)_{(l,m)} \tag{10}$$

Brute force: (CPU, Parallel computing, GPU,...)

- Slow derivation, FFT, sky map of N^2 cells, $\left[PSF\right]_{l,m}$ requires $\sim MNlog_2N$ multiplications; $\sim N^2 log_2 N$.
- Slow derivation, FFT, sky map of N_s sources, $\sim N^4 log_2 N$ multiplications for all $\left[PSF\right]_{l,m}$.

Linear Algebra and Approximation: computationally cheaper, but an approximation

- Quick derivation, FFT, sky map of N_s sources, $\sim N^2 log_2 N$ multiplications for all $\left[PSF\right]_{l,m}$.

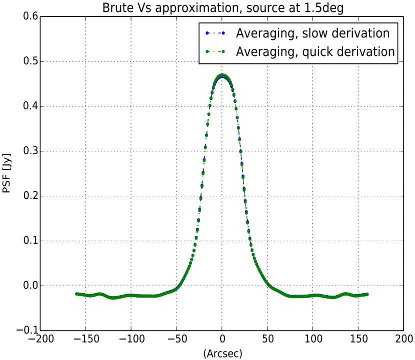
uv plane approximation

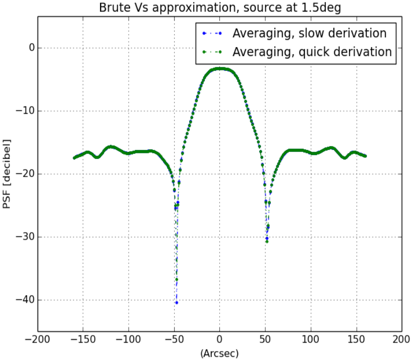
- Approximate the offset PSF (Pseudo PSF) visibilities by the centre bin

$$V_{\mathbf{b}_0}(\mathbf{x}) \simeq \left(\widetilde{G}_{\mathbf{b}_0} \circ \left[C_{\mathbf{b}_0} \widetilde{\Pi}_{\mathbf{b}_0} \right] \circ \left[C_{\mathbf{b}_0} \widetilde{W}_{\mathbf{b}_0} \right] \right)_{(l,m)} \tag{11}$$

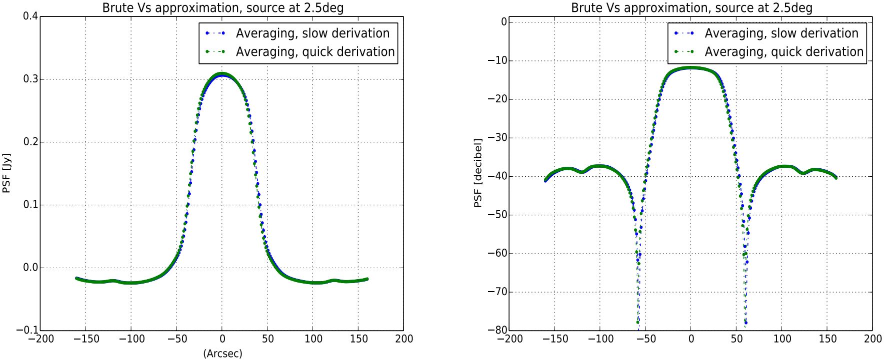
- Requires $\sim Nlog_2 N$ multiplications

Results


Brute Vs approximation, source at 1.5deg
Averaging, slow derivation
Averaging, quick derivation
PSF [Jy]
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
−200
−150
−100
−50
0
50
100
150
200
(Arcsec)


Brute Vs approximation, source at 1.5deg
Averaging, slow derivation
Averaging, quick derivation
PSF [decibel]
0
−10
−20
−30
−40
−200
−150
−100
−50
0
50
100
150
200
(Arcsec)

Results



Brute Vs approximation, source at 2.5deg
Averaging, slow derivation
Averaging, quick derivation
PSF [Jy]
0.4
0.3
0.2
0.1
0.0
−0.1
−200
−150
−100
−50
0
50
100
150
200
(Arcsec)
Brute Vs approximation, source at 2.5deg
Averaging, slow derivation
Averaging, quick derivation
PSF [decibel]
0
−10
−20
−30
−40
−50
−60
−70
−80
−200
−150
−100
−50
0
50
100
150
200

Image plane approximation

- Approximate the offset PSF from the PSF at the Phase centre (Nominal PSF)
- Find a function H and the variation in time/frequency $\Delta\Psi$ and $\Delta\Phi$

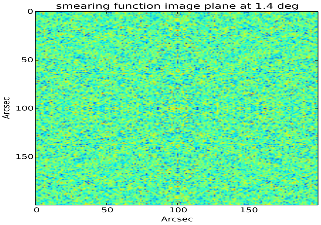
$$PSF_{l,m} \simeq \left[H(\frac{\Delta\Psi}{2})H(\frac{\Delta\Phi}{2})\right] \circ PSF_{l_0,m_0} \tag{12}$$

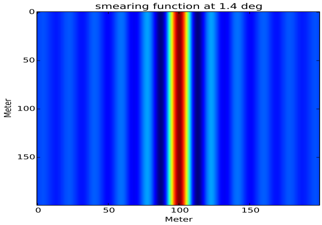
Algorithm 1 Contruct Maxtrices, $\mathbf{\Delta\Psi}$ and $\mathbf{\Delta\Phi}$

procedure VARIATION IN TIME AND FREQUENCY
    $FoV = N_{pix}\Delta l \Delta m$ //N_{pix} the number of pixels
    $\Delta u = \Delta v = 1/FoV$
    $v_0 = -(N_{pix} - 1)\Delta v/2$
    $h = (2v_0)/(N_{pix} - 1)$, $cst = \frac{-\pi}{432 \times 10^2}$
    for i from 1 to N_{pix} **do**
        $u_0 = -(N_{pix} - 1)\Delta u/2$
        for j from 1 to N_{pix} **do**
            $\theta = \arctan(u_0/v_0)$
            $du_{i,j} = cst\sqrt{u_0^2 + v_0^2}\sin\theta$
            $dv_{i,j} = cst\sqrt{u_0^2 + v_0^2}\cos\theta$
            $\Delta\Psi_{i,j} = \pi(du_{i,j}l + dv_{i,j}m)\Delta t$
            $\Delta\Phi_{i,j} = \pi\frac{\Delta\nu}{\nu}\sqrt{l^2 + m^2}\sqrt{u_0^2 + v_0^2}$
            $u_0 = u_0 + h$
        end for
        $v_0 = v_0 + h$
    end for
end procedure

$$K = sinc(\frac{\Delta\Psi}{2})sinc(\frac{\Delta\Phi}{2})$$

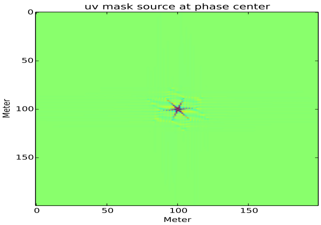
$$\widetilde{K}$$





$$\widetilde{PSF}_{l_0,m_0}$$

$$\widetilde{PSF}_{l,m}$$



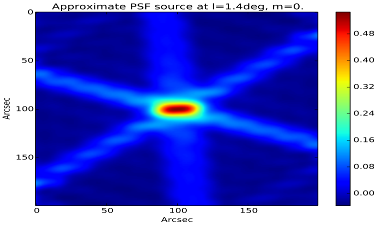
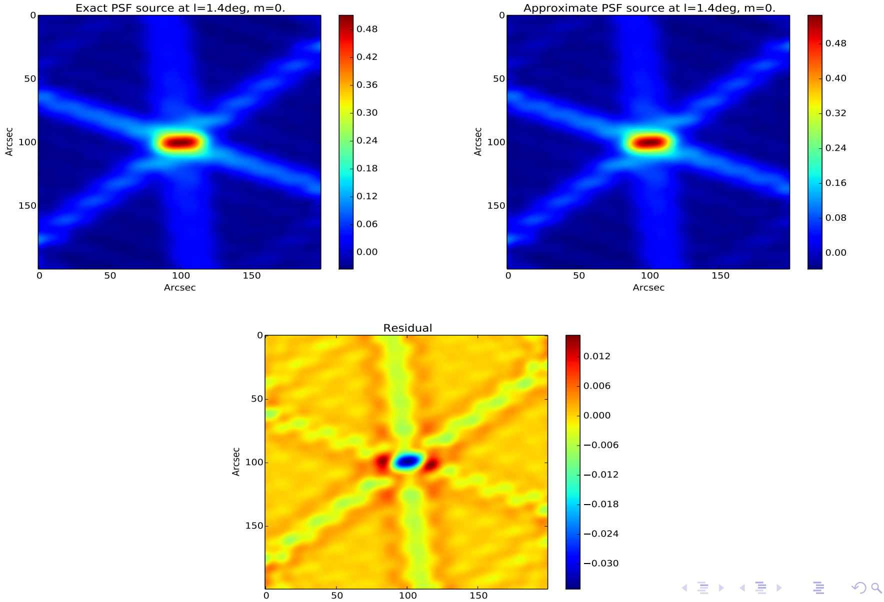


Image plane approximation: results



Computational time

uv-plane approximation

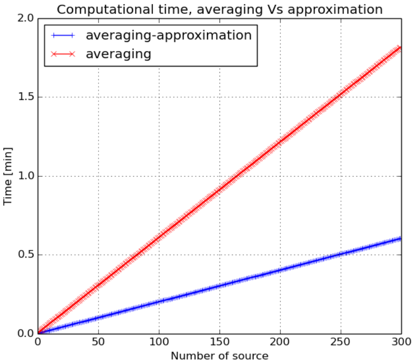
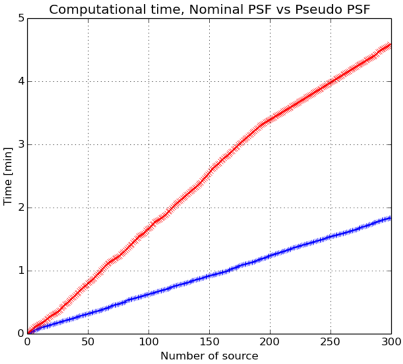
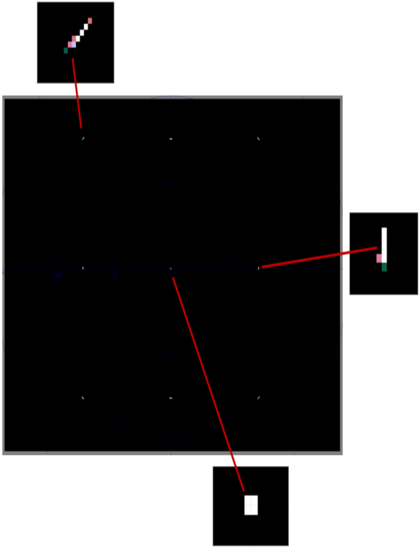


Image plane Approximation

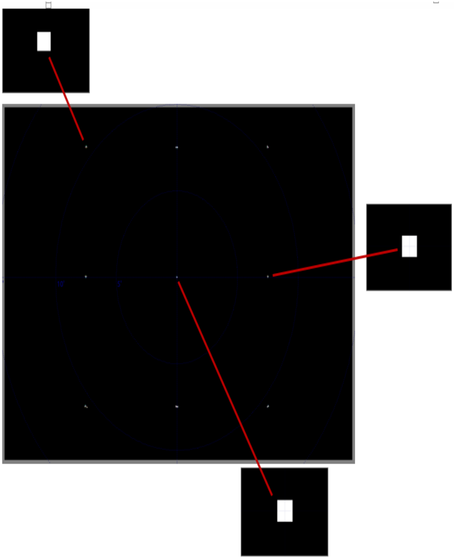


DDFacet(courtesy: C. Tasse, Observatoire de Paris)

Model image, mominal PSF



Model image, pseudo PSF



C'est la fin!!! Merci!!!