

SPECTRAL CHARACTERISTICS OF LOW-FREQUENCY VARIABILITY
IN COMPACT EXTRAGALACTIC RADIO SOURCES

by

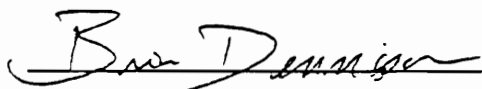
Maria J. Pantazopoulou

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY

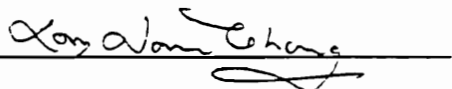
in

Physics

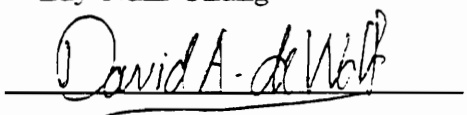
APPROVED



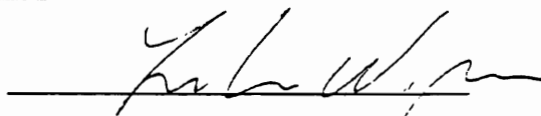
Brian Dennison, Chairman



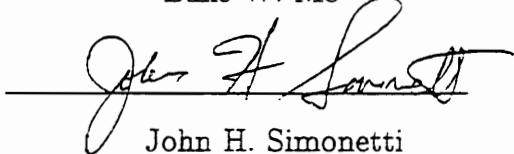
Lay Nam Chang



David A. de Wolf



Luke W. Mo



John H. Simonetti

August, 1996

Blacksburg, Virginia

Keywords: radio sources, scintillations, interstellar medium

SPECTRAL CHARACTERISTICS OF LOW-FREQUENCY VARIABILITY IN COMPACT EXTRAGALACTIC RADIO SOURCES

by

Maria J. Pantazopoulou

Brian Dennison, Chairman

Department of Physics

(ABSTRACT)

We examine the refractive scintillation hypothesis of low frequency variability via numerical simulation in order to account for the spectral characteristics of the observed fluctuations. Plane waves from extragalactic radio sources propagating through the interstellar medium, a medium with fluctuations of the refractive index due to electron density irregularities, emerge from that medium corrugated. Since fluctuations present on scales greater than the Fresnel scale act refractively, the emerging wavefront has a curvature which produces a refractive amplification or deamplification in the flux density. We develop a numerical algorithm to characterize the phase fluctuations in the wavefront and we simulate the resulting intensity distribution in the limit of geometrical optics. We then produce light curves by taking trajectories in the simulated intensity distribution plane and we compare our results with statistical properties of existing data from a 5-year monitoring program of 32 extragalactic sources at frequencies 0.318, 0.430, 0.606, 0.880 and 1.4 GHz. We find that the refractive scintillation hypothesis is in good agreement with the data at comparable timescales and that the variability in the simulated light curves diminishes within an octave of frequency, in agreement with those of the observed light curves with the same behavior.

Acknowledgements

I would like to thank my advisor professor Dennison for all his help, as well as professor Simonetti, and the members of my committee professors Lay Nam Chang, David de Wolf, Luke Mo for their useful sugestions.

Table of Contents

1	Introduction	1
2	Flux variability at low frequencies in extragalactic sources	3
2.1	The spectrum of synchrotron radiation	5
2.2	Absorption by ionized Hydrogen	10
2.3	Expanding source models	11
2.4	Refractive effects	12
2.4.1	Phase and curvature of the wavefront	17
2.4.2	Range of the power law index and image wander	17
3	Numerical simulation	21
3.1	Prewhitening	22
3.2	Construction of the phase screens	22
3.3	The inner and outer scales	28
4	Wave propagation through the phase screen	30
4.1	Focal lengths	33
4.2	The interpolation algorithms	34
5	Comparison with data	55
	References	83
	Appendix A	85
A.1	Plasmas	85
A.2	Evaluation of $\langle \phi^2 \rangle$	85
A.3	Evaluation of $\langle (\nabla^2 \phi)^2 \rangle$	87
A.4	Evaluation of $\langle (\nabla \phi) \cdot (\nabla \phi) \rangle$	88
	Appendix B	89
B.1	Documentation of the code	89
B.2	Fortran listing of read-data-spline.f	93
B.3	Fortran listing of structure-function.f subprogram	106

B.4	Fortran listing of Phaethon.f	112
B.5	Listing of papameter and include files	131
B.6	Fortran listing of cell-interpol.f subprogram	134

List of Figures

1. Radiation of an electron in a magnetic field	5
2. Source 1117 + 14 radio spectra	8
3. Source 1611 + 34 radio spectra	9
4. Source 0235 + 16 radio spectra	10
5. Refraction in ISM	18
6. The random 2-d screen	25
7. Log P versus log k for the phase screen	26
8. The phase screen and contours $\alpha = 3.6$	27
9. Phase screen contours $\alpha = 2.1, 3.1$	28
10. Position of rays on the observer's plane	32
11. Focal lengths	34
12. Intensity at the observer's plane at 318, 430, 606, 880 and 1400 MHz	36
13. Images of intensity at the observer's plane at 318 and 430 MHz	37
14-18 Simulated light curves at the observer's plane at all 5 frequencies	38-42
19. Images of intensity at the observer's plane at 318, and 430 MHz	43
20-24 Simulated light curves at the observer's plane at all 5 frequencies	44-48
25. Images of intensity at the observer's plane at 318 and 430 MHz	49
26-30 Simulated light curves at the observer's plane at all 5 frequencies	50-54
31. Auto and cross structure-functions from the model for $\alpha = 3.6$	56
32. Auto and cross structure-functions from the model for $\alpha = 2.1$	57
33. Auto and cross structure-functions from the model for $\alpha = 3.1$	58
34. Auto and cross structure-functions for the source 1117 + 14	59
35. Auto and cross structure-functions for the source 0202 + 14	60
36. Auto and cross structure-functions for the source 0256 + 07	61
37. Auto and cross structure-functions for the source 1345 + 12	62
38. Auto and cross structure-functions for the source 1611 + 34	63

39. Auto and cross structure-functions for the source 0056 + 00	64
40. Auto and cross structure-functions for the source 0116 + 31	65
41. Auto and cross structure-functions for the source 0446 + 11	66
42. Auto and cross structure-functions for the source 0723 - 00	67
43. Auto and cross structure-functions for the source 1039 + 02	68
44. Auto and cross structure-functions for the source 1055 + 01	69
45. Auto and cross structure-functions for the source 2230 + 11	70
46. Auto and cross structure-functions for the source 1422 + 20	71
47. Auto and cross structure-functions for the source 1548 + 05	72
48. Auto and cross structure-functions for the source 1901 + 31	73
49. Auto and cross structure-functions for the source 2144 + 09	74
50. Auto and cross structure-functions for the source 2319 + 27	75
51. Cross structure-functions minima	80
52. Effective screen distance and source angular size versus galactic latitude .	81

List of Tables

1. Root mean square variabilities and characteristic scales of observed sources . 75
2. Root mean square variabilities from the model 76
3. Characteristic scales and minima of cross structure- functions from the model 77
4. Minima of cross structure-functions of all frequencies 78

Chapter 1

Introduction

Many compact radio sources, quasars and the nuclei of some galaxies, show large intensity variations on time scales ranging from less than a week to years. These variations appear as bursts, first at short wavelengths and then at longer wavelengths with reduced amplitudes. There are also variations that appear simultaneously at all wavelengths with diminishing amplitudes as we go to shorter wavelengths, and with no evidence for any periodic phenomena. Many theoretical efforts have been made in order to interpret the observed spectra from radio sources, some involving intrinsic mechanisms in the source itself, and some involving propagation effects along the line of sight. In this work we examine the variability of radio sources at low frequencies, below 1.4 GHz, known in the literature as low-frequency variability in extragalactic radio sources. In the following chapters, we show that intrinsic models are not always consistent when trying to explain the low frequency variability, and we examine wave propagation effects along the line of sight, in particular refractive effects, resulting from fluctuations in the density of the ionized gas in the interstellar medium.

Plane waves from extragalactic radio sources propagating through a medium with fluctuations of the refractive index (electron density irregularities) emerge from that medium corrugated. Since fluctuations present on scales larger than the Fresnel scale act refractively, the emerging wavefront has a curvature, which in turn produces a convergence or divergence of rays, in other words a refractive amplification or deamplification. Thus the medium causes phase perturbations in the wavefront, modulating the intensity by refractive focusing, which are noticed as intensity fluctuations as the observer passes through the pattern. The purpose of this work is:

1. To develop a numerical simulation to characterize these phase fluctuations,
2. to simulate the resulting intensity patterns in the limit of geometrical optics, and

3. to compare the model with the statistical properties of the data (Mitchell et al. 1994), and evaluate spectral properties of the variations.

In general we get spectral representations of sources via Fourier transforms. The term plane waves usually is an idealization that does not really exist in the real world. That is, waves are nonplanar in nature because they are generated by finite sources such as antennas and scatterers. But this kind of waves can always be expanded in terms of plane waves through Fourier transforms. In our case, the waves we are dealing with are plane to an excellent approximation as they are incoming upon the galaxy.

Chapter 2

Flux variability at low frequencies in extragalactic sources

Low-frequency variability in extragalactic radio sources, was discovered in the quasars CTA 102 (Sholomitskii 1965, Hunstead 1972) and 3C454.3 (Hunstead 1972).

Sources with 3 to 30 percent variations in flux density observed at frequencies less than 1.4 GHz, on months to year time scales, would have brightness temperatures well exceeding the inverse Compton limit $T_{bc} = 10^{12} K$, were the variations intrinsic. It is known that in a compact bright source of synchrotron radiation, the collisions of fast electrons with radio photons may lead to a redistribution of the radiation spectrum (Pacholczyk, 1970). Classical Compton scattering refers to the scattering of photons off electrons in the electron rest-frame. But we are interested in processes in a reference frame in which the electrons have relativistic energies, so there is need to look at the Compton scattering in the moving electron's frame. This scattering process is known as inverse Compton scattering, since energy now is transferred from electrons to photons. Thus the inverse Compton process tends to increase the radiation energy density in the source. The maximum brightness temperature of an opaque synchrotron source is limited due to this process to the above T_{bc} , corresponding to the case where the energy loss by synchrotron radiation is equal to the energy loss by inverse Compton scattering (Verschuur and Kellerman ,1974).

Intrinsic mechanisms involve synchrotron radiation, for sources at cosmological redshifts (e.g. $z \approx 1$), the linear scale is given by $c\tau$, where τ is the time scale of the variations. The Rayleigh-Jeans law relates S_ν , the flux density, to the brightness temperature T_b ; that is $S_\nu = \frac{2kT_b}{\lambda^2} \pi \theta^2$ where θ the angular radius of the source. Now $\theta \approx \frac{c\tau}{D} \approx \frac{c\tau H_0}{cz}$, where D is the distance to the source, and H_0 the Hubble constant. For $\tau \approx 1 y$ and variation in the flux density $\delta S_\nu = 1 Jy$, we solve for the

approximate brightness temperature

$$T_b \approx \left(\frac{\lambda^2}{2\pi k}\right) \delta S_\nu \left(\frac{D}{c\tau_*}\right)^2 \approx 10^{15} \text{ K} \quad (1)$$

The time scale of the variations τ , in the observations, is related to the time scale of the variation at the source by $\tau = (1 + z)\tau_*$. Relativistic corrections to the angular size of the source are not taken into account since the value of T_b is estimated to order of magnitude. Since for the linear sizes set by τ , T_b exceeds by far 10^{12} K and since the change in flux is over a narrow range of frequencies, about an octave, while variations due to intrinsic events should be observed over a wide frequency range, one can seriously question intrinsic mechanisms.

2.1 The spectrum of synchrotron radiation

Synchrotron radiation is the radiation emitted by accelerated relativistic electrons. A single electron spiraling in a magnetic field at ultra-relativistic velocities has its radiation concentrated in a cone of half-angle $\delta \approx \frac{1}{\gamma}$ (Jackson 1975, Pacholczyk, 1970), where γ is the Lorenz factor. An observer sees emission only when the cone is pointed toward him (see figure 1).

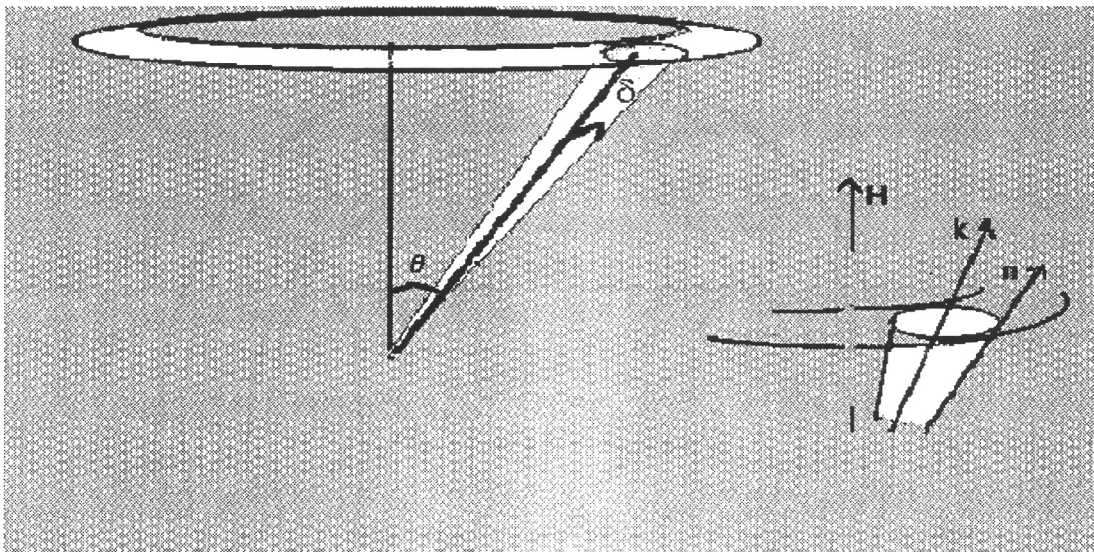


Fig.1 Radiation of an electron in a magnetic field.

At large distances from the moving charge, the power radiated, \tilde{p}_Ω , into a solid angle $d\Omega$, received by the observer at distance R_0 from the element of volume containing the charge, can be calculated (Jackson 1975, Pacholczyk, 1970).

$$\tilde{p}_\Omega = \frac{c}{4\pi} \mathcal{E}^2 R_0^2 \quad (2)$$

Where \mathcal{E} is the intensity of the electric field. \mathcal{E} , as well as H , the intensity of the magnetic field, can be computed from the Lienard-Wiechert potentials. The electric field of the emitted radiation is a periodic function of time, with period

$T = 2\pi \sin^2 \theta / \omega_H$, where θ is the angle between the electron trajectory and the magnetic field (the pitch angle), and $\omega_H = eH/\gamma mc$ is the electron gyrofrequency. Since for ultrarelativistic motion most of the radiation is emitted within a small angle δ around the direction \vec{n} of the electron velocity, an observer located along a direction \vec{k} sees pulses of radiation when the direction of the instantaneous velocity n coincides with k . The electric vector $\vec{\mathcal{E}}(t)$ can be expanded into Fourier series of monochromatic waves of frequencies $n(\omega_H/\sin^2 \theta)$, where n is the harmonic number. One can calculate then the emission coefficient ε_ν from a stationary electron cloud, with a distribution function $N(E, \vec{k}, \vec{R})$, where E is the electron energy, under the assumption that the distribution does not vary sharply within a small angle $\approx \gamma^{-1}$ (Pacholczyk 1970).

$$\varepsilon_\nu = \frac{\sqrt{3}e^3}{4\pi m_e c^2} H \sin \theta \int_0^\infty N(E) F(x) dE \quad (3)$$

where $x = \nu/\nu_c$, and $\nu_c = \frac{3}{4\pi} \omega_H \sin \theta \gamma^3$ is the critical frequency (that is when \vec{n} and \vec{k} coincide). $F(x)$ is a special function tabulated by Pacholczyk (1970) that has a broad peak near $\nu \approx 0.28\nu_c$. For monoenergetic electrons, with $N(E) = N_0 \delta(E - E_0)$, equation 3 becomes

$$\varepsilon_\nu = \frac{\sqrt{3}e^3}{4\pi m_e c^2} H \sin \theta N_0 F(x) \quad (4)$$

For a power-law distribution of the electrons in the cloud, $N(E) = N_0 E^{-\xi}$, the emission coefficient is given by

$$\varepsilon_\nu = C_5(\xi) N_0 (H \sin \theta)^{\frac{\xi+1}{2}} \left(\frac{\nu}{2(3e/(4\pi m^3 c^5))} \right)^{\frac{1-\xi}{2}} \quad (5)$$

where $C_5(\xi)$ is, again, a special function tabulated by Pacholczyk (1970). Consequently, from equation 5, one can see that for $\xi > 1$, the radio spectrum obeys a power law

$$S(\nu) \propto \nu^{-\frac{-(\xi-1)}{2}} = \nu^{-\beta} \quad (6)$$

where $\beta \equiv \frac{\xi-1}{2}$ and $S(\nu)$, the flux density, is defined as energy per unit area per unit time per unit frequency interval.

It is worth mentioning here that most radio sources show power law radiofrequency spectra, with a spectral index $\beta = 0.8$ in the optically thin regime, which corresponds to a $\xi \approx 2.6$.

If the low frequency variations are intrinsic, difference spectra between two different times should then have the character of synchrotron spectra since any additional flux is caused by injected synchrotron emitting electrons.

With this in mind, we shall compare observed difference spectra with synchrotron spectra in both the monoenergetic and power-law cases.

We plot the function $F(\frac{\nu}{\nu_c})$ versus frequency, as well as the spectrum of the varying component of flux in the *Lowvar* data set (Mitchell et al. 1994), for the sources 1117+14 and 1611+34, in order to compare the change in the flux density of the sources to that of synchrotron radiation from a monoenergetic electron spectrum.

Such an energy spectrum gives the strongest high frequency cutoff in the radiated spectrum.

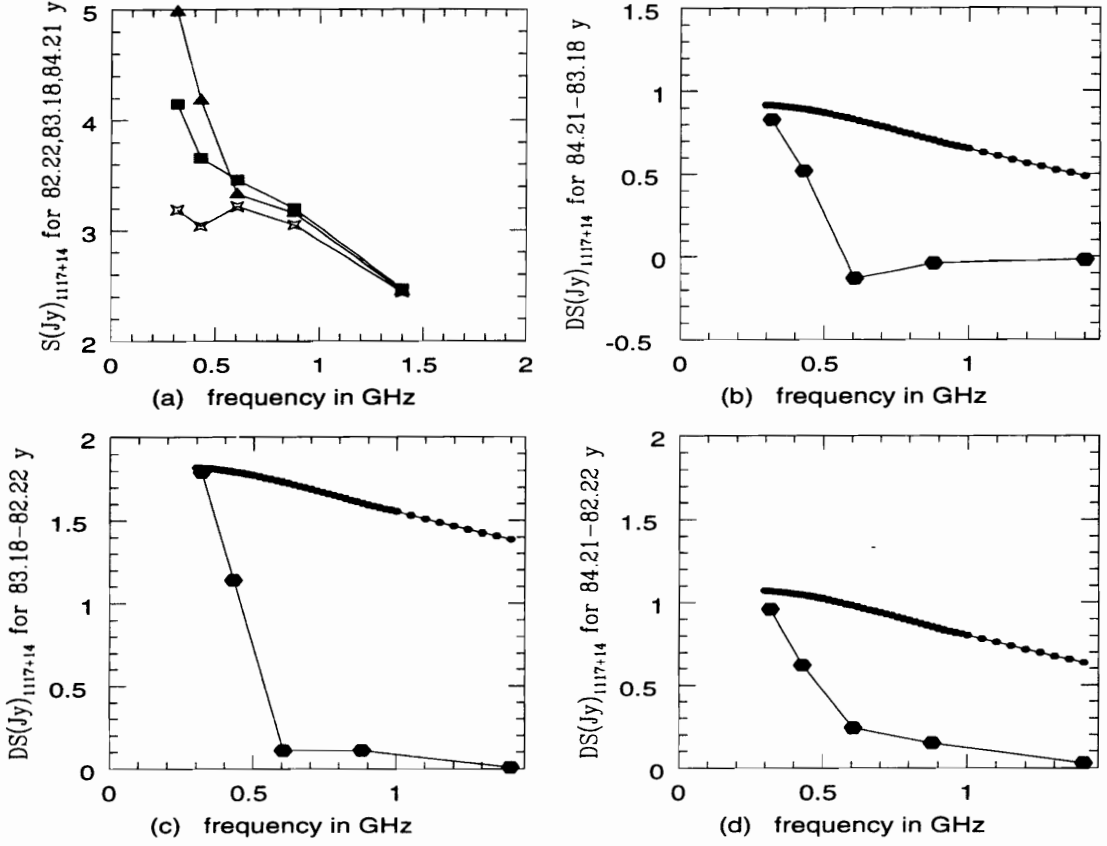


Fig.2(a,b,c,d) Source 1117 + 14; (a) radio spectra for three dates, 82.22y in asterisks, 83.18y in triangles and 84.21y in squares; (b), (c), (d) difference in spectra between dates shown. Synchrotron emission spectrum (small hexagons) for monoenergetic electrons.

When we plot the spectrum of the excess emission attributable to the variations (by using the spectrum on a particular date as a base level; figures 2b,c,d and 3b,c,d) we find that at low frequencies this spectrum is remarkably steep and very different from that expected if the change in flux is due to monoenergetic synchrotron radiation.

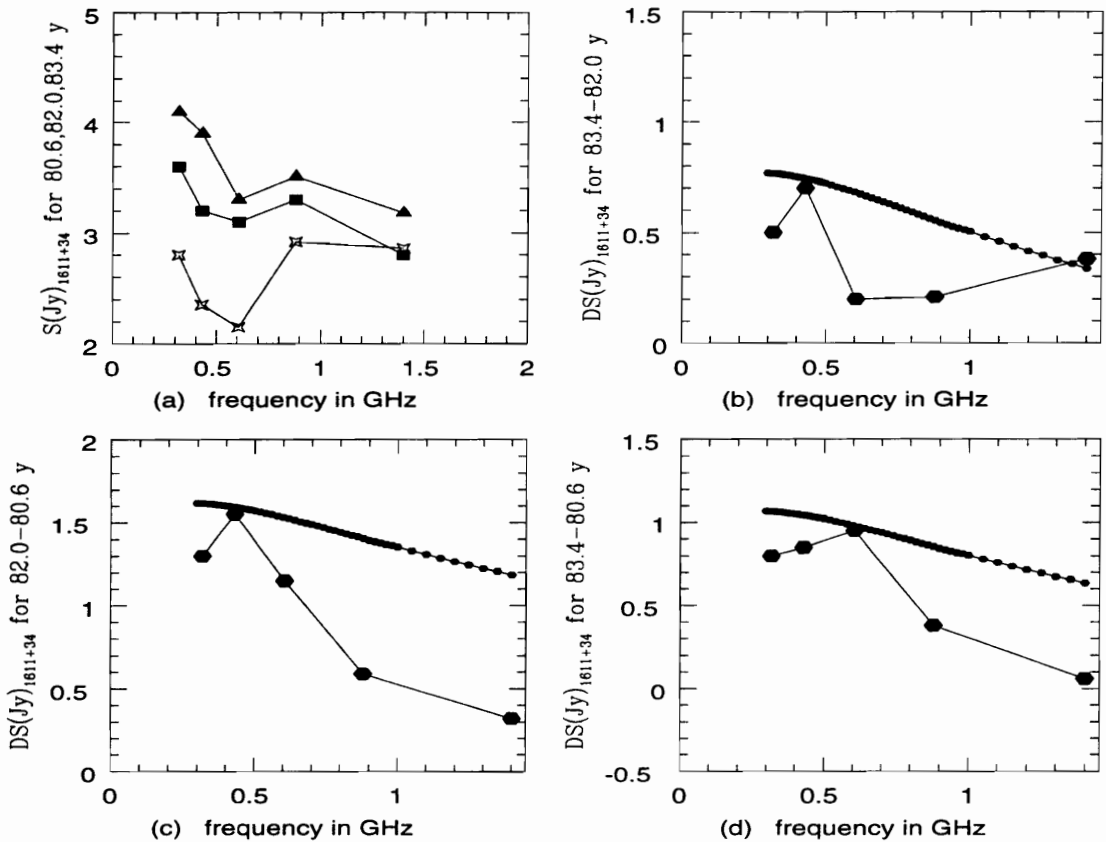


Fig.3(a,b,c,d) Source 1611 + 34; (a) radio spectra for three dates, 80.6y in asterisks, 82.0y in triangles and 83.4y in squares; (b), (c), (d) difference in spectra between dates shown. Synchrotron emission spectrum (small hexagons) for monoenergetic electrons.

The case that the distribution function for the electrons is a power law, as given by equations 5 and 6, is even weaker in explaining our steep spectrum, since we will need a spectral index of $\beta = 5.5$ for the spectrum in figure 2c, that will correspond to $\xi = 12.1$, which is much higher than any value previously observed. On the other hand, in figure 4a,b,c,d, for the source 0235 + 16, we see that the change in flux does not have a particular trend at low frequencies, as expected, since this source is a bona Flux variability at low frequencies in extragalactic sources

fide intrinsic variable (O'Dell et. al. 1988).

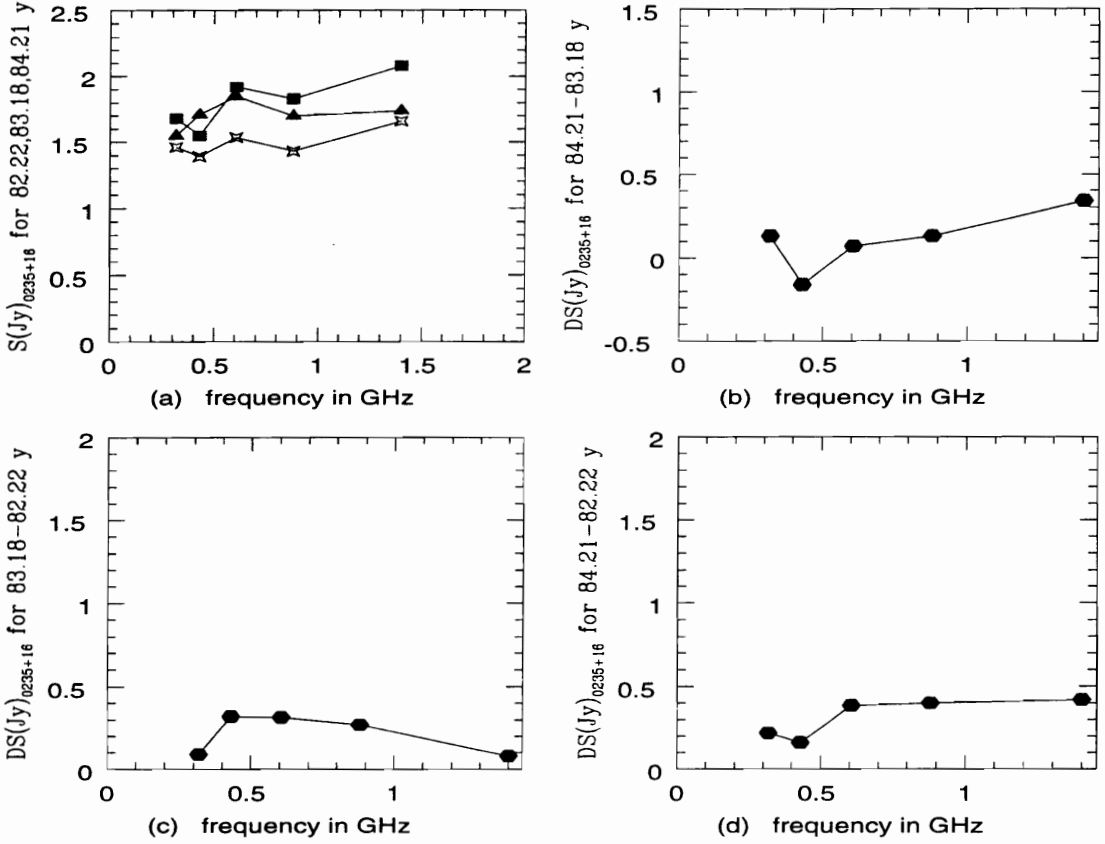


Fig.4(a,b,c,d) Source 0235 + 16; (a) radio spectra for three dates, 82.22y in asterisks, 83.18y in triangles and 84.21y in squares; (b), (c), (d) difference in spectra between dates shown.

In the above treatment we did not take absorption and stimulated emission into consideration. This is because when one takes into account absorption (Pacholczyk, 1970) then $S_\nu \propto \nu^{2.5}$ for the case of a power law distribution of the electrons. For monoenergetic electrons, the flux also increases with frequency. But our spectra have a negative slope, and one could not explain the excess flux with that scenario.

2.2 Absorption by ionized Hydrogen

Flux variability at low frequencies in extragalactic sources

One of the reasons that the observed radio spectrum differs from the radiated spectrum could be due to a cloud of ionized gas located in front of the source. If this is the case, then the observed flux density will fall sharply below a cutoff frequency ν_0 where the optical depth is unity. Marscher (1979) found that a wide variety of observed effects can be explained that way. Free-free absorption by a dense, $n_e \geq 10^5 \text{ cm}^{-3}$, photoionized cloud was said to be the most favorable mechanism by which changes in optical depth can cause the observed flux variations. But this model was shown to be less than convincing by Condon et. al, (1979). Condon (1979) showed that results from different models were in favor of relativistic expansion of incoherent electron-synchrotron components over coherent emission mechanisms, when compared to observational data.

2.3 Expanding source models

O'Dell (1982) summarizes the fundamental assumptions of expanding cloud models: uniformity, particle conservation, adiabaticity of relativistic electrons and frozen-in magnetic field. In the *Lowvar* data set, (Mitchell et al. 1994) there are two main types of sub-GHz variations, delayed and simultaneous. The delayed bursts propagate from high to low frequencies with amplitudes decreasing with decreasing radio frequency. Simultaneous bursts occur at the same time with amplitudes comparable at all low frequencies, or decreasing sharply with increasing frequency. A grave consequence of trying to explain the simultaneous sub-GHz variability in terms of evolution of a transparent synchrotron source is the high brightness temperatures (equation 1), implied by the flux changes and variability timescales. Although the expanding cloud models can account for the delayed bursts qualitatively (Shklovsky 1965, Kellerman 1968), they are not as successful in accounting for the simultaneous ones. A point has been made (O'Dell 1982) that simultaneous spectral variations would originate at different regions of the source at low frequencies, than the ones

at high frequencies. Some models state that the relativistic electrons are reheated, re-isotropized, or injected due to shocks, compression of the magnetic field and other acceleration mechanisms (Marscher, 1982, Aller 1982). Aller (1982) and others have pointed out that the time scales of the variability will require the emitting regions to travel towards the observer at relativistic velocities in order to avoid excess T_b in the rest frame of the emitting region, as given by equation 1. Thus, if we consider re-acceleration of particles by a shock front, then the particles as well as the shock-front should be moving toward the observer relativistically, down a jet for example, which is an extra restriction to the theory.

2.4 Refractive effects

Although wave-propagation effects have been examined for the last two decades, there remain strides to be made in the theory of wave propagation with regard to refractive effects, as well as unsolved questions about the turbulent nature of the interstellar medium, and the spectrum that characterizes it. It is now accepted (Shapirovskaia 1978, Rickett et al. 1984, Rickett 1986) that refraction is at least partly responsible for the variability. Refractive effects in the inhomogeneous interstellar plasma have been visited many times over the years, triggered by observations (Dennison et al. 1981, Mitchell et al. 1994), as well as by the recognition of refractive effects in strong scintillations (Rickett 1986, Rickett et al. 1984, Fanti et al. 1981).

We will be taking a close look at low-frequency variability in compact extragalactic sources at frequencies less than 1.4 GHz and time scales ranging from months to years, testing the refractive scintillation hypothesis over a wide range of length scales with the help of a model. Here we examine the spectral characteristics of low-frequency variables with a numerical simulation, in the geometrical optics limit, and compare them with statistical properties of the data (Mitchell et al., 1994).

The source angular size, θ , spatially filters out fluctuations on scales smaller

than θz , where z is the screen distance. Dennison and Condon (1981), Steinebring and Condon (1990) searched for diffractive scintillations and conclude that the most compact ($\theta \approx 10^{-3}$ arcsec) extragalactic sources do not show diffractive scintillation. Dennison and Condon searched for diffractive scintillations in the spectral domain, by searching for corrugations in the spectra of these objects, with frequency scale of the order of the correlation bandwidth. They concluded that these objects did not show diffractive scintillation above $\approx 0.1\%$ even during strong low-frequency variations. The smallest scale in the observed variations then is $\theta z_c \approx v\tau$. For a cloud moving with $v = 50 \text{ km sec}^{-1}$ and $\tau = 1$ year and $z_c = 500 \text{ pc}$, we have $L = v\tau = 1.5 \times 10^{13}$ cm. The Fresnel scale is $L_F = \sqrt{\lambda z_c}$. At meter wavelengths $L_F \approx 10^{10}$ cm. The spatially filtered variations have a scale much greater than the Fresnel scale, and thus the geometric optics limit is valid and the variations in the flux density come from the focusing or defocusing effect of the irregularities, if the observer is located at a distance comparable to the focal length.

We are considering the ionized component of the interstellar medium and regard the large scale structure of the ionized gas as turbulent fluctuations, with spatial power spectrum characterized by a power law with index α , that can take Kolmogorov ($\alpha = \frac{11}{3}$ in a 3-dimensional medium), and non Kolmogorov values:

$$P_{\delta n}(q) = C_n^2 q^{-\alpha} \quad (7)$$

where C_n^2 gives the strength of the fluctuations and q is the spectral wavenumber. In this work we perform a numerical simulation of the radio waves propagating through phase screens. Let $n(\vec{r})$ be the density of the ISM (interstellar medium) with fluctuation $\delta n(\vec{r})$. The autocorrelation function is defined as

$$A_{\delta n}(\vec{\tau}) = \frac{1}{V} \int_V \delta n(\vec{r}) \delta n(\vec{r} + \vec{\tau}) dV \quad (8)$$

where V is the volume. Denote with $\delta n(\vec{k})$, the fourier transform of $\delta n(\vec{r})$, and $P_{\delta n}(\vec{k})$, the power spectrum of the density. Then

$$P_{\delta n}(\vec{k}) = \frac{1}{(2\pi)^{3/2}} \int A_{\delta n}(\vec{\tau}) e^{-i\vec{k}\cdot\vec{\tau}} d^3\vec{\tau} \quad (9)$$

If we write $e^{-i\vec{k}\cdot\vec{\tau}} = e^{-i\vec{k}\cdot\vec{r}} e^{i\vec{k}\cdot\vec{r}-i\vec{k}\cdot\vec{\tau}}$ then we have:

$$P_{\delta n}(\vec{k}) = \frac{1}{(2\pi)^{\frac{3}{2}}} \frac{1}{V} \int d^3\vec{\tau} \int d^3\vec{r} \delta n(\vec{r}) \delta n(\vec{r} + \vec{\tau}) e^{i\vec{k}\cdot\vec{r}} e^{-i\vec{k}\cdot(\vec{r}+\vec{\tau})} \quad (10)$$

or

$$P_{\delta n}(\vec{k}) = \frac{1}{V} \frac{1}{(2\pi)^{\frac{3}{2}}} \int d^3\vec{r} \delta n(\vec{r}) e^{i\vec{k}\cdot\vec{r}} \int d^3\vec{\tau} \delta n(\vec{r} + \vec{\tau}) e^{-i\vec{k}\cdot(\vec{r}+\vec{\tau})} \quad (11)$$

Where the first integral can be identified with $n^*(k)$ and the second with $n(\vec{k})$. Then the power spectrum is given by

$$P_{\delta n}(\vec{k}) = \frac{(2\pi)^{\frac{3}{2}}}{V} \delta n^*(\vec{k}) \delta n(\vec{k}) \quad (12)$$

with the assumption that the process describing the density fluctuations is stationary, i.e. if one chooses a window large enough, the physical quantities involved can be described by their means.

As the wave is incident on a slab of thickness z_c , the density variations will cause the phase to change. Naturally, we will have to simulate the phase. To do this we have to calculate the phase power spectrum and relate it to the power spectrum of the density fluctuations. The phase minus an additive constant is given by:

$$\Phi(x, y) = -r_e \lambda \int_0^{z_c} n(\vec{r}) dz \quad (13)$$

and its fourier transform:

$$\Phi(\vec{k}) = \frac{1}{2\pi} \int \int \Phi(\vec{r}) e^{-i(k_x x + k_y y)} dx dy \quad (14)$$

Where r_e is the classical electron radius. Let's denote $\vec{k}(k_x, k_y, 0) = \vec{q}$, and $\vec{r}(x, y, 0) = \vec{\rho}$ then

$$\Phi(\vec{q}) = -\frac{r_e \lambda}{2\pi} \int \int dx dy e^{-i\vec{q} \cdot \vec{\rho}} \int_0^{z_c} n(\vec{r}) dz \quad (15)$$

Substituting $n(\vec{r}) = \frac{1}{(2\pi)^{\frac{3}{2}}} \int_V d^3 \vec{k}' e^{i\vec{k}' \cdot \vec{r}} n(\vec{k}')$, we have :

$$\Phi(\vec{q}) = -\frac{r_e \lambda}{(2\pi)^{\frac{5}{2}}} \int \int \int \int \int \int e^{-i\vec{q} \cdot \vec{\rho}} e^{i\vec{k}' \cdot \vec{r}} n(\vec{k}') dx dy dz dk'_x dk'_y dk'_z \quad (16)$$

By writing $i\vec{k}' \cdot \vec{r} = i\vec{q}' \cdot \vec{\rho} + ik'_z z$ we have:

$$\Phi(\vec{q}) = -\frac{r_e \lambda}{(2\pi)^{\frac{5}{2}}} \int \int \int \int \int \int e^{-i(\vec{q}-\vec{q}') \cdot \vec{\rho}} e^{ik'_z z} n(\vec{k}') dk'_x dk'_y dk'_z dx dy dz \quad (17)$$

or

$$\Phi(\vec{q}) = -r_e \lambda \sqrt{2\pi} \int d^3 k' \delta(\vec{q} - \vec{q}') \delta(k'_z) n(\vec{k}') \quad (18)$$

or

$$\Phi(\vec{q}) = -r_e \lambda \sqrt{2\pi} n(\vec{q}) \quad (19)$$

Where $n(\vec{q}) = n(k_x, k_y, 0)$, and z_c is very large compared to the outer scale.

The power spectrum for the phase can now be calculated as

$$\begin{aligned} P_\Phi(\vec{q}) &= \frac{1}{2\pi} \int d^2 \vec{\tau} e^{-i\vec{q} \cdot \vec{\tau}} A_\phi(\vec{\tau}) \\ &= \frac{1}{2\pi} \int d^2 \vec{\tau} e^{-i\vec{q} \cdot \vec{\tau}} \frac{1}{A} \int d^2 \vec{\rho} \Phi(\vec{\rho}) \Phi(\vec{\rho} + \vec{\tau}) \\ &= \frac{1}{2\pi A} \int d^2 \vec{\tau} \int d^2 \vec{\rho} \Phi(\vec{\rho}) \Phi(\vec{\rho} + \vec{\tau}) e^{i\vec{q} \cdot \vec{\rho}} e^{-i\vec{q} \cdot (\vec{\rho} + \vec{\tau})} \end{aligned}$$

$$\begin{aligned}
&= \frac{(2\pi)}{2\pi A} \int d^2\bar{\rho} \Phi(\bar{\rho}) e^{i\bar{q}\cdot\bar{\rho}} \frac{1}{(2\pi)} \int d^2\bar{\tau} \Phi(\bar{\rho} + \bar{\tau}) e^{-i\bar{q}\cdot(\bar{\rho}+\bar{\tau})} \\
&= \frac{(2\pi)}{A} \Phi^*(\bar{q}) \Phi(\bar{q})
\end{aligned} \tag{20}$$

Again with the stationary process assumption, we see that if it is valid for δn it must be also valid for Φ .

Substituting for the phase $\Phi(\bar{q}) = -r_e \lambda (2\pi)^{\frac{1}{2}} n(\bar{q})$ we can get the power spectrum in terms of density:

$$P_{\Phi}(\bar{q}) = V \frac{(2\pi)^2}{A} r_e^2 \lambda^2 n(\bar{q}) n^*(\bar{q}) \tag{21}$$

From equations 7,12 and 21 above we have:

$$P_{\Phi}(\bar{q}) = C_n^2 q^{-\alpha} V \frac{\sqrt{2\pi}}{A} r_e^2 \lambda^2 \tag{22}$$

or

$$P_{\Phi}(\bar{q}) = C_n^2 q^{-\alpha} z_c \sqrt{2\pi} r_e^2 \lambda^2 \tag{23}$$

where z_c is the cloud thickness and $k_z = 0$, assuming that z is far greater than any wavelength of interest, therefore it does not play a role in any wavelength cutoff.

Now we can get a power law for the phase:

$$P_{\Phi}(\bar{q}) = C_{\Phi}^2 q^{-\alpha} \tag{24}$$

valid for $q_1 < q < q_2$, where q_1 is the low spatial frequency-cutoff corresponding to the outer scale (in the order of parsecs), and q_2 is the high spatial frequency-cutoff corresponding to the inner scale.

From equations 23 and 24 above we have:

$$C_{\Phi}^2 = \sqrt{2\pi} (r_e \lambda)^2 z_c C_n^2 \tag{25}$$

which gives the relation between C_Φ and C_n , thus connecting the power spectrum of the phase with that of the density.

2.4.1 Phase and curvature of the wavefront

Since the difference in the pathlength in a ray is proportional to the phase, we have in the geometrical optics limit

$$\bar{\theta} = \frac{\lambda}{2\pi} \bar{\nabla} \Phi \quad (26)$$

where $\bar{\theta}$ is the refraction angle, a two-dimensional vector, and Φ is the phase. The mean squared phase fluctuation is calculated in appendix A, and is given by

$$\langle \Phi^2 \rangle = C_\Phi^2 \left(\frac{q_2^{2-\alpha} - q_1^{2-\alpha}}{2-\alpha} \right), \quad \alpha \neq 2 \quad (27)$$

For $\alpha > 2$ and $q_2 \gg q_1$, we have $\langle \Phi^2 \rangle \approx C_\Phi^2 \frac{q_2^{2-\alpha}}{\alpha-2}$, and $\langle \Phi^2 \rangle^{1/2}$ tends to be outer-scale dominated.

The average curvature of the wavefront is obtained from

$$\left(\frac{1}{r} \right)_{rms} = \frac{\lambda}{2\pi} \langle (\nabla^2 \Phi)^2 \rangle^{1/2} \quad (28)$$

where $\left(\frac{1}{r} \right)_{rms}^{-1}$ is a characteristic radius of curvature, and hence, a characteristic focal length. The quantity $\langle (\nabla^2 \Phi)^2 \rangle$ describes the second derivative of the phase and is given by

$$\langle (\nabla^2 \Phi)^2 \rangle = C_\Phi^2 \left[\frac{q_2^{6-\alpha}}{6-\alpha} - \frac{q_1^{6-\alpha}}{6-\alpha} \right], \quad \alpha \neq 6 \quad (29)$$

see appendix A. For $\alpha < 6$ and $q_2 \gg q_1$, $\langle (\nabla^2 \Phi)^2 \rangle \approx C_\Phi^2 \frac{q_2^{6-\alpha}}{6-\alpha}$, thus, in this regime, the curvature tends to be inner-scale dominated.

2.4.2 Range of the power-law index and image wander

We now consider a statistical average of the first derivative of the phase, which via equation 26 will characterize the refraction angle. It is calculated (see appendix A) to be

$$\langle (\bar{\nabla}\Phi) \cdot (\bar{\nabla}\Phi) \rangle = C_{\Phi}^2 \left[\frac{q_2^{4-\alpha} - q_1^{4-\alpha}}{4-\alpha} \right], \quad \alpha \neq 4 \quad (30)$$

Since this gives an *rms* slope of the wavefront, an *rms* refraction angle can be defined via equation 26. In equation 30, we have two regimes:

$$\alpha > 4 : \langle (\bar{\nabla}\Phi) \cdot (\bar{\nabla}\Phi) \rangle \approx C_{\Phi}^2 \frac{q_1^{4-\alpha}}{\alpha-4} \quad q_2 \gg q_1$$

$$\alpha < 4 : \langle (\bar{\nabla}\Phi) \cdot (\bar{\nabla}\Phi) \rangle \approx C_{\Phi}^2 \frac{q_2^{4-\alpha}}{\alpha-4} \quad q_2 \gg q_1 \quad (31)$$

Thus the refraction angle is inner-scale dominated.

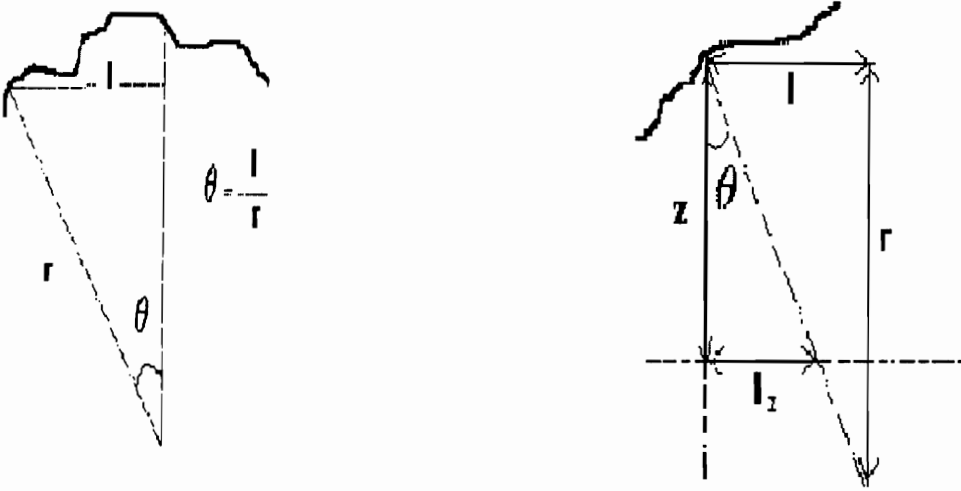


Fig. 5 a. Refraction in ISM.

b. Distance to the observer's plane.

We define a scale l , the transverse scale which generates ray crossings at the

characteristic focal distance see figure 5a,

$$l = \frac{\langle (\bar{\nabla}\Phi) \cdot (\bar{\nabla}\Phi) \rangle^{1/2}}{\langle (\nabla^2\Phi)^2 \rangle^{1/2}} = \frac{\theta_{rms}}{(\frac{1}{r})_{rms}} \quad (32)$$

The characteristic ray translation at the distance of the observer's plane is $l_z = l \frac{z}{r}$ or $l_z = lz(\frac{1}{r})_{rms}$ (figure 5b), where z is the distance from phase screen to observer's plane. Now

$$l = \left[\frac{6 - \alpha}{4 - \alpha} \frac{q_2^{4-\alpha} - q_1^{4-\alpha}}{q_2^{6-\alpha} - q_1^{6-\alpha}} \right]^{1/2}, \text{ for } \alpha \neq 4, \alpha \neq 6 \quad (33)$$

For $\alpha < 4$ and $q_2 \gg q_1$, we have

$$l \approx \left[\frac{6 - \alpha}{4 - \alpha} \frac{q_2^{4-\alpha}}{q_2^{6-\alpha}} \right]^{1/2} = \left(\frac{6 - \alpha}{4 - \alpha} \right)^{1/2} \frac{1}{q_2} \quad (34)$$

Let $q_2 \equiv \frac{2\pi}{a_{inner}}$ and $q_1 \equiv \frac{2\pi}{a_{outer}}$, where a_{inner} and a_{outer} are the inner and outer scales. If we define $M = q_2/q_1$, then, for $4 < \alpha < 6$ and $q_2 \gg q_1$ we have $l \approx \frac{1}{2\pi} \left(\frac{6-\alpha}{\alpha-4} \right)^{1/2} M^{\frac{\alpha-4}{2}} a_{inner}$. One has to mention here that near $\alpha \approx 6$ and $\alpha \approx 4$ these expressions are inaccurate, and the more general expression (equation 33) must be used.

For $\alpha < 4$, ray crossing occurs primarily near the inner scale (as does focusing of the ray bundles). Image wander is small. Gwinn et al. (1988), report from their observations of refractive wander in masers, that their results rule out spectra for density fluctuations in the interstellar medium with $\alpha > 4.0$, placing a constraint to indices less than or about the Kolmogorov value of 3.67 for two particular lines of sight. If $\alpha > 4.0$ then one would expect image wander since we will have increased ray crossings for the range of α between 4 and 6, but the above authors did not report any. There are other observations (Cordes et al. 1985, Armstrong et al. 1981), associated with scattering of pulsar radiation, in an effort to determine the power-law index, which indicate that $\alpha \approx 3.6$. Stinebring and Condon (1990) observed Flux variability at low frequencies in extragalactic sources

25 pulsars at 310, 416 and 750 MHz, and measured their flux density variations and their timescales. Since pulsars are point sources, and intrinsically stable continuum sources, they are valuable for studies of refractive interstellar scintillation. They have estimated the power-law slope for 11 time series (by looking at the power-law slope of the structure functions for lags much less than the saturation time scale, that contains information about the density inhomogeneity spectrum of the interstellar medium. They found α to have a mean of 3.9 but with a considerable spread ranging from 3.1 to 4.6. This spread may be attributable to limited sampling. They concluded that since pulsars are intrinsically stable point sources, one can use their observed long-term variability to estimate effects of refractive interstellar scintillation on extended extragalactic sources, such as quasars. Their observations show that refractive interstellar scintillation is caused by scattering in regions with angular size $\theta \leq 10^{-3}$ arcsec. and hence extragalactic sources with about the same angular size will undergo refractive interstellar scintillation as well, but the time scales will be months instead of weeks, since the extragalactic sources are extended and do not have the large proper motions characteristic of pulsars.

Chapter 3

Numerical simulation

Initially, we simulate the phase screen in the following manner: As the wave is incident on a slab of thickness z , the density variations will cause the phase to change. The phase is given (see Appendix A) by

$$\Phi(r) \propto -r_e \lambda \int_0^{z_c} n(\vec{r}) dz \quad (35)$$

and the refraction angle is going to be determined by the gradient of the phase screen as

$$\bar{\theta} = \frac{1}{k} \bar{\nabla} \Phi \quad (36)$$

Note that since we have deflection in any direction we write θ as a two-dimensional vector.

We start by generating a white screen (see section 3.1) in $x - y$ space in order to build the slab of variable phase. Next, we Fourier-transform the screen to q -space. At this point we impose upon the screen a power-law form by multiplying by $q^{-\alpha/2}$. The phase power-spectrum is given by

$$P(\bar{q}) = C_\phi^2 q^{-\alpha} \quad (37)$$

Next, we use an inverse Fourier transform to get back to $x - y$ space, at which point we are ready to propagate the wave through the first phase screen.

For the completion of the model we need to determine the parameters of the problem and incorporate them in the simulation in order to be able to compare the computer generated data to the actual observational data (Mitchell et al., 1994). To do this, we have to calculate the mean and the root mean square of the actual computer generated grid, and find the power-spectrum for the simulated data, and

consequently after imposing the desired power-law, relate the results to equation 24 of chapter 2.

3.1 Prewhitening

To simulate the refractive scintillation hypothesis for the observed variations in the data, we produce $2 - D$ random screens (see figure 4) with flat power-spectrum as we show in the next section, in other words white screens (Williamson et. al. 1990). We use the random number generator ran1 (Press et al. 1988), to produce two dimensional random number arrays ranging from 0 to 1 . This array $\mathcal{R}(x, y)$ can be thought of as an array with zero mean, $R(x, y)$ offset by a constant, c .

$$\mathcal{R}(x, y) = c + R(x, y) \quad (38)$$

In our case, $c = 1/2$.

3.2 Construction of the phase screens

To follow the construction of the numerical simulation we need to define the *rms* of the phase in terms of the power-spectrum:

$$\begin{aligned} \langle \phi^2 \rangle &= \lim_{A \rightarrow \infty} \frac{1}{A} \int \int \phi^2(\bar{\rho}) dx dy \\ &= \frac{1}{2\pi} \int \int d^2 k P_\phi(\bar{q}) \end{aligned} \quad (39)$$

Let's start with $R(\bar{\rho})$, the array of random numbers from $-1/2$ to $1/2$, and find the mean squared:

$$\langle R^2 \rangle = \int_{-1/2}^{1/2} w^2 dw = \frac{1}{12} \quad (40)$$

where w is the random variable . The *rms* is:

$$R_{rms} = \langle R^2 \rangle^{\frac{1}{2}} = \frac{1}{\sqrt{12}}$$

Now we find the phase and the power spectrum of the computer generated data:

$$\phi(\bar{q}) = R(q) (q)^{-\alpha/2} \quad (41)$$

and

$$P_\phi(\bar{q}) = \frac{2\pi}{A} R^*(q) R(q) (q)^{-\alpha} \quad (42)$$

First we calculate $R^*(q)R(q)$:

$$R^*(\bar{q})R(\bar{q}) = \frac{1}{(2\pi)^2} \int \int dx dy e^{-i\bar{q}\cdot\bar{\rho}} R(\bar{\rho}) \int \int dx' dy' e^{i\bar{q}\cdot\bar{\rho}'} R(\bar{\rho}') \quad (43)$$

Converting this integral to a sum, we have

$$\begin{aligned} |\mathcal{R}(k)|^2 &= \frac{1}{(2\pi)^2} \sum_{x,y} \Delta x \Delta y (c + R(x,y)) \sum_{x',y'} \Delta x' \Delta y' (c + R(x',y')) e^{i\bar{k}\cdot(\bar{r}' - \bar{r})} \\ &= \frac{1}{(2\pi)^2} \sum_{x,y} \Delta x \Delta y \sum_{x',y'} \Delta x' \Delta y' [c^2 + (R(x,y) + R(x',y')) + R(x,y)R(x',y')] e^{i\bar{k}\cdot(\bar{r}' - \bar{r})} \end{aligned}$$

Note that we represent the random array as $\mathcal{R}(k) = c + R(x, y)$, as discussed earlier. We now calculate the expectation of $|R^2|$. Since we have no correlation at any non-zero scale for the set of random numbers, the only contribution comes from $(x, y) = (x', y')$. Thus,

$$\begin{aligned} \langle |\mathcal{R}(k)|^2 \rangle &= \frac{c^2}{(2\pi)^2} L^4 \delta_{k,0} + 0 + \frac{1}{(2\pi)^2} \sum_{x,y} \Delta x \Delta y R(x,y) R(x,y) \Delta x \Delta y \\ &= \frac{c^2}{(2\pi)^2} L^4 \delta_{k,0} + \frac{L^2}{(2\pi)^2} \langle R^2 \rangle \end{aligned} \quad (44)$$

where the first term is equivalent to the direct current term(DC) and acts only at $k = 0$ in the fourier domain, but we set it to zero in the code. The second term is zero because the variations have zero mean, and the last term is as calculated in the previous page $\langle R^2 \rangle = 1/12$. Then

$$\begin{aligned}
 |\mathcal{R}(k)|^2 &= \frac{L^2}{(2\pi)^2} \langle R(\bar{r})^2 \rangle \Delta x \Delta y \\
 &= \frac{L^2}{(2\pi)^2} \langle R(\bar{r})^2 \rangle (1)(1) \\
 &= \frac{N^2}{(2\pi)^2} \frac{1}{12}
 \end{aligned} \tag{45}$$

where the spacing between the grid points $\Delta x, \Delta y$ has been taken to be unity, and the length L is the number of grid points N .

At this point we should find the power spectrum for the grid, which is given by:

$$P_R = \frac{2\pi}{L^2} \frac{L^2}{(2\pi)^2} \frac{1}{12} = \frac{1}{12} \frac{1}{2\pi} \tag{46}$$

As we can see, the power spectrum is flat at this point, the screen is shown in figure 6, as expected since we started out with a white screen (prewhitening).

white screen (64x64)

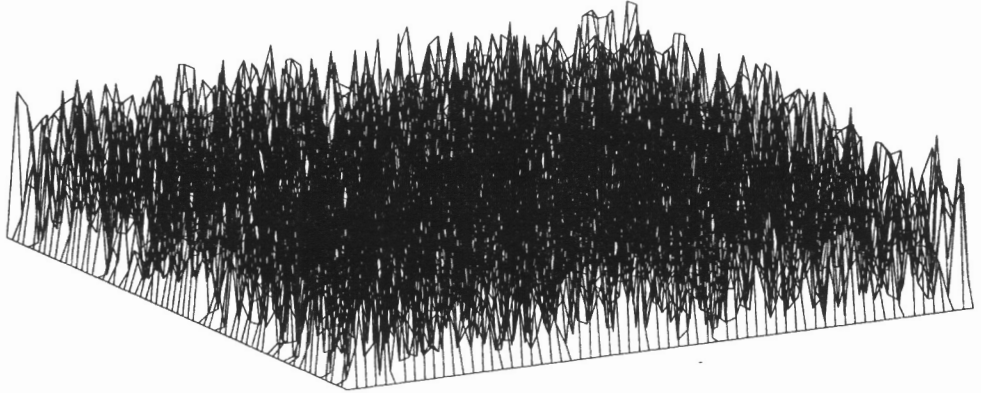


Fig. 6 The random 2-d screen

We now impose a power law, and the power spectrum becomes (Fig. 7):

$$P_{\phi}(k) = \frac{1}{2\pi} \frac{1}{12} (q)^{-\alpha} \quad (47)$$

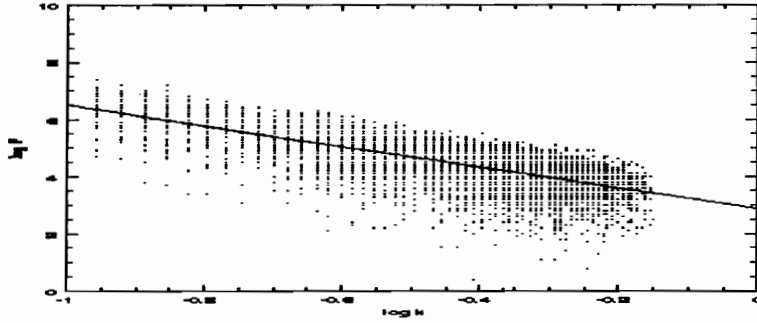


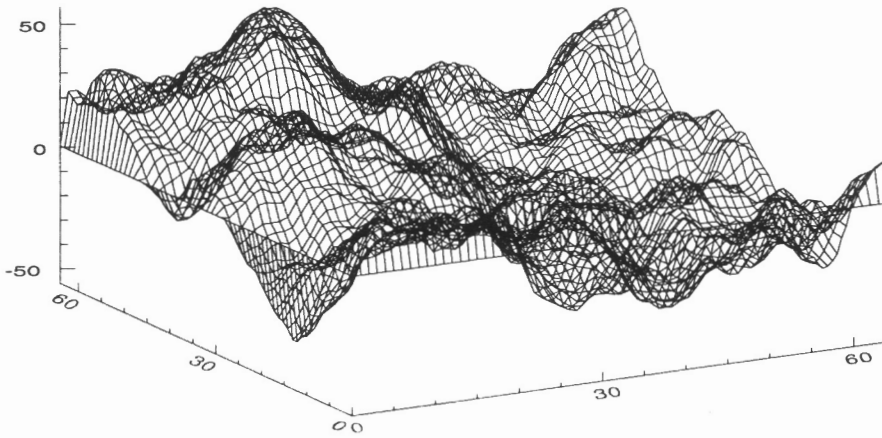
Fig. 7 Log P versus log k for the phase screen

The rms of the phase can now be calculated by substituting for the power-spectrum and integrating (see appendix A):

$$\begin{aligned}
 \langle \phi^2 \rangle &= \frac{1}{A} \int \int (\Phi(\rho))^2 d^2 \rho \\
 &= \frac{C_\phi^2}{2 - \alpha} [q^{2-\alpha}]_{q_1}^{q_2} \\
 &\approx \frac{C_\phi^2}{\alpha - 2} q_1^{2-\alpha}
 \end{aligned} \tag{48}$$

for $\alpha > 2$ and $q_2 \gg q_1$, q_1 being the large scale cutoff (small q , large wavenumber, that is the outer scale), and q_2 the small scale cutoff (small wavenumber, that is the inner scale).

The phase screen (64x64), ($\alpha = 3.6$)



The phase screen ($\alpha = 3.6$) contours at levels -40, -35... 0.5... 75

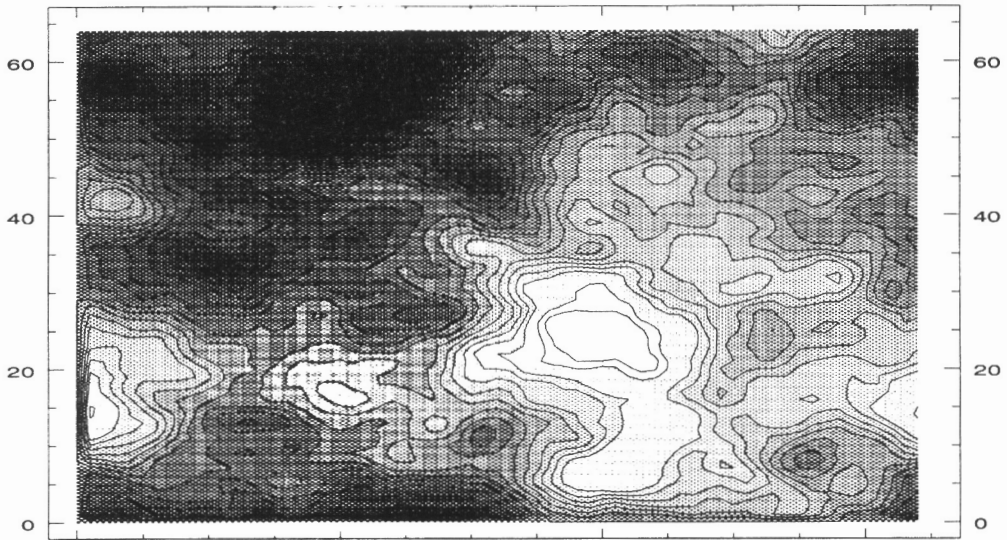


Fig.8 Simulated phase and contours, as a function of $x - y$ coordinates. Phase units arbitrary. x - y units in pixel elements.

Figures 8 and 9 show some of the resulting phase screens, and contours at different levels, for α 3.6 and 2.1.

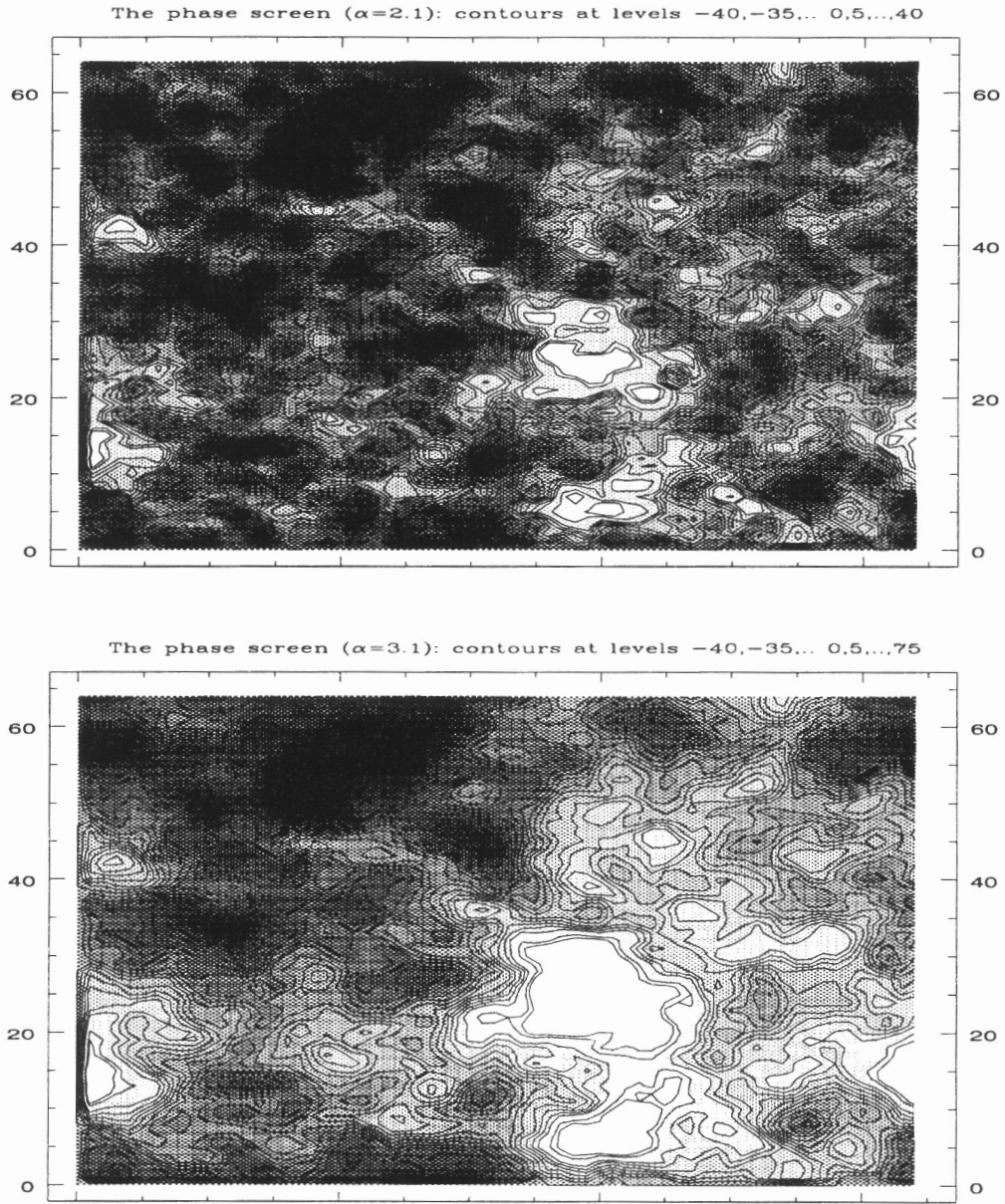


Fig. 9 Contours at levels $1*(-40, \dots, 70, 75)$. Phase units arbitrary. x-y units in pixel elements.

3.3 The inner and outer scales

Numerical simulation

As noted above, the electron density irregularities are characterized by the spatial power spectrum given by equation 47, which holds for wavenumbers $q_1 \ll q \ll q_2$. When the exponent α equals 11/3 we have the Kolmogorov spectrum.

Since the wavenumbers of interest to scintillation are between the outer and inner scales, and the power-spectrum should fall rapidly to zero for $q > q_2$, where q_2 is the high frequency cutoff corresponding to the inner scale, we modify the power spectrum (Rickett 1990) as:

$$P_\phi(q) = C_\phi^2 q^{-\alpha} e^{-q^2/q_2^2} \quad (49)$$

The wavenumber cutoffs for our phase screen as imposed by computation, are :

$$q_1 = \frac{2\pi n}{N}, \text{ at } n = 1, \quad q_1 = \frac{2\pi}{N}$$

and

$$q_2 = \frac{2\pi n}{N}, \text{ at } n = N/2, \quad q_2 = \pi$$

These values come about from the grid spacing, the grid size, and the manner by which the phase screen is Fourier transformed.

We use the FFT, (Press 1988), for a grid size $N \times N$, or $2N \times N$ since we are using complex variables. We keep the grid spacing equal to unity. In addition, the FFT employs the following definition for the Fourier transform:

$$\begin{aligned} \Phi(f) &= \int \Phi(r) e^{-i2\pi f r} dr \\ &= \int \Phi(r) e^{-iqr} dr \end{aligned}$$

where $q = 2\pi f$. In our algorithm, that will translate for the outer and inner scales as $q_1 = 1/N$, and $q_2 = 0.5$. The inner scale imposed by the grid size is also the Nyquist critical frequency for the FFT as applied to our grid.

Chapter 4

Wave propagation through the phase screen

At this stage we are ready to propagate plane waves through the phase screens as prepared above. We shall denote the screen with unprimed symbols and the observer's plane with primed symbols. We will consider a small bundle of rays going through the screen at the position \bar{r}_o ; the boundary of the bundle being at \bar{r} . Where the bundle intersects the screen, the bundle has a circular cross-section. The radius of the bundle is given by the magnitude of $\bar{\beta} = \bar{r} - \bar{r}_o$, at that point. The observer's plane is located at some distance z from the screen. As the ray bundle emerges from the screen at an angle θ , refractive effects have already taken place, and as a result, the bundle of rays is deformed. The cross-section is not necessarily circular, but is in general elliptical. This comes about because of astigmatism in the ray bundle due to the random variations of the refractive index in the screen.

Let \bar{r}'_o be the position that the ray bundle intersects the observer's plane, with boundary at \bar{r}' . Then:

$$\bar{r}'_o = \bar{r}_o + z\bar{\theta}(\bar{r}_o) \quad (50)$$

$$\bar{r}' - \bar{r}'_o = \bar{r} - \bar{r}_o + z(\bar{\theta}(\bar{r}) - \bar{\theta}(\bar{r}_o)) \quad (51)$$

Now $\bar{\beta}'$ is given by:

$$\bar{\beta}' = \bar{\beta} + z(\bar{\theta}(\bar{r}) - \bar{\theta}(\bar{r}_o)) \quad (52)$$

$$\beta'_x = \beta_x + z(\bar{\beta} \cdot \bar{\nabla}\theta_x) \quad (53)$$

to first order. Therefore:

$$\beta'_x = \beta_x + z \left[\beta_x \left(\frac{\partial \theta_x}{\partial x} \right)_{\bar{r}_o} + \beta_y \left(\frac{\partial \theta_x}{\partial y} \right)_{\bar{r}_o} \right] \quad (54)$$

$$\beta'_y = \beta_y + z \left[\beta_x \left(\frac{\partial \theta_y}{\partial x} \right)_{\bar{r}_o} + \beta_y \left(\frac{\partial \theta_y}{\partial y} \right)_{\bar{r}_o} \right] \quad (55)$$

Taking into account that $\bar{\theta} = \frac{\lambda}{2\pi} \bar{\nabla} \bar{\phi}$ i.e $\theta_x = \frac{\lambda}{2\pi} \frac{\partial \phi}{\partial x}$ and $\theta_y = \frac{\lambda}{2\pi} \frac{\partial \phi}{\partial y}$ we have:

$$\beta'_x = \beta_x + \frac{\lambda z}{2\pi} \left[\beta_x \left(\frac{\partial^2 \phi}{\partial x^2} \right)_{\bar{r}_o} + \beta_y \left(\frac{\partial^2 \phi}{\partial x \partial y} \right)_{\bar{r}_o} \right] \quad (56)$$

$$\beta'_y = \beta_y + \frac{\lambda z}{2\pi} \left[\beta_x \left(\frac{\partial^2 \phi}{\partial x \partial y} \right)_{\bar{r}_o} + \beta_y \left(\frac{\partial^2 \phi}{\partial y^2} \right)_{\bar{r}_o} \right] \quad (57)$$

$$\beta'_x = \beta_x + \beta_x \Phi_{xx} + \beta_y \Phi_{xy} \quad (58)$$

and

$$\beta'_y = \beta_y + \beta_x \Phi_{xy} + \beta_y \Phi_{yy} \quad (59)$$

where $\Phi_{xx} = \frac{\lambda z}{2\pi} \frac{\partial^2 \phi}{\partial x^2}$ and so on. We define $l_F^2 = \frac{\lambda z}{2\pi}$.

In other words, we have the following transformation for $\bar{\beta}$ in matrix notation:

$$\begin{pmatrix} \beta'_x \\ \beta'_y \end{pmatrix} = \begin{pmatrix} \Phi_{xx} + 1 & \Phi_{xy} \\ \Phi_{xy} & \Phi_{yy} + 1 \end{pmatrix} \begin{pmatrix} \beta_x \\ \beta_y \end{pmatrix}$$

Figure 10 gives the positions of the rays at the observer's plane, represented by an x, the point on the grid representing the position of the rays on the phase screen.

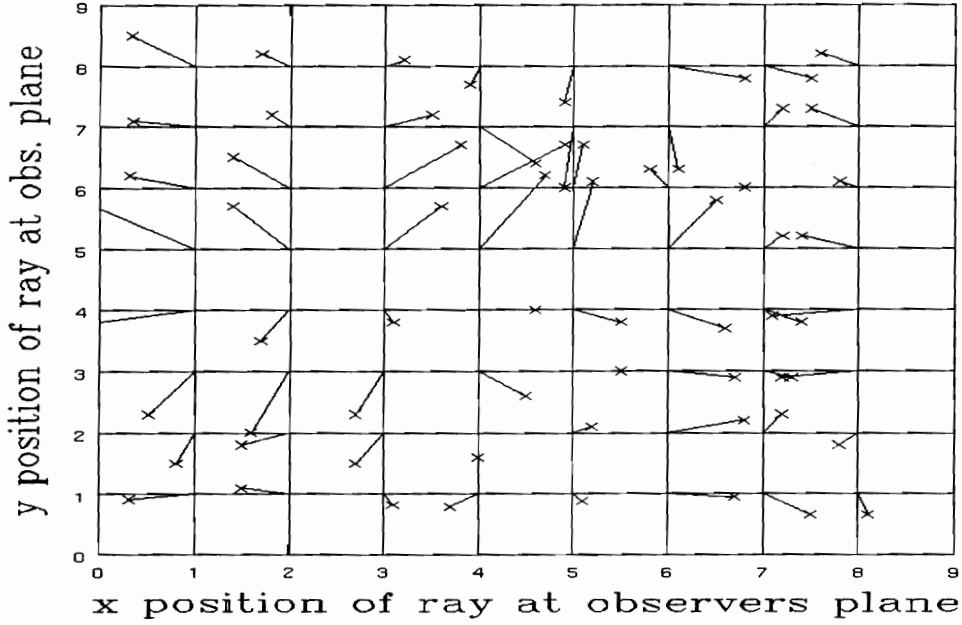


Fig.10 Position of rays on the observer's plane, relative to original locations in the phase screen

Let λ_1 and λ_2 be the two eigenvalues of the transformation matrix, which are functions of position on the screen. We have to look at the intensity along the ray bundle. To do this we have to consider the cross section of the ray bundle at the screen, where the cross-section is circular, and at the observers plane, where the cross-section is now elliptical. The intensity along the ray bundle is defined as the ratio of the two cross sections:

$$I = \frac{\pi r^2}{\pi a b} \tag{60}$$

where a and b are the semimajor and semiminor axes of the elliptical cross section. In a coordinate system (x, y) in which the transformation matrix is diagonal, then $\beta'_x = \lambda_1 \beta_x$ and $\beta'_y = \lambda_2 \beta_y$, and β'_x and β'_y are aligned with the principal axes of the ellipse. The intensity is then given by

$$I = \frac{1}{|\lambda_1 \lambda_2|} \quad (61)$$

and the eigenvalues are given by:

$$\lambda = \frac{1}{2} [(l_F^2 \phi_{xx} + l_F^2 \phi_{yy} + 2) \pm ((l_F^2 \phi_{xx} + l_F^2 \phi_{yy} + 2)^2 - 4((l_F^2 \phi_{xx} + 1)(l_F^2 \phi_{yy} + 1) - (l_F^4 \phi_{xy}^2))]^{\frac{1}{2}}$$

4.1 Focal lengths

To get an estimate of the characteristic focal length, that is the distance from the screen that the ray bundles tend to focus, we follow two methods: first, by using equation 61, and the calculated eigenvalues and their means; and second, by calculating an estimator of the radius corresponding to the curvature in the phase screen. As mentioned in the previous paragraph, $\Phi_{xx} = \frac{\lambda z}{2\pi} \phi_{xx}$, where ϕ is the actual simulated phase, and z is the distance from the screen (interstellar cloud) to the observer. The quantity $\sqrt{\frac{\lambda z}{2\pi}} = l_F$ is a measure of the distance from the screen to the observer. For these calculations, we arbitrarily scale $\lambda = 1$ at the observation frequency of 318 MHz.

We determine the focal lengths first by calculating the eigenvalues and finding the value of l_F which minimizes $\langle |\lambda_2| \rangle$. It should be noted that λ_2 corresponds to the minor axis of the elliptical ray bundle and λ_1 to the major axis. These calculations were performed on screens with $\alpha = 2.1, 3.1$, and 3.6 , as described in section 2.4.1. As we see from figure 11, λ_2 shows a minimum at $l_F = 0.45$ in the 3.6 case. This behavior determines a characteristic focal distance of the irregularities in the phase screen.

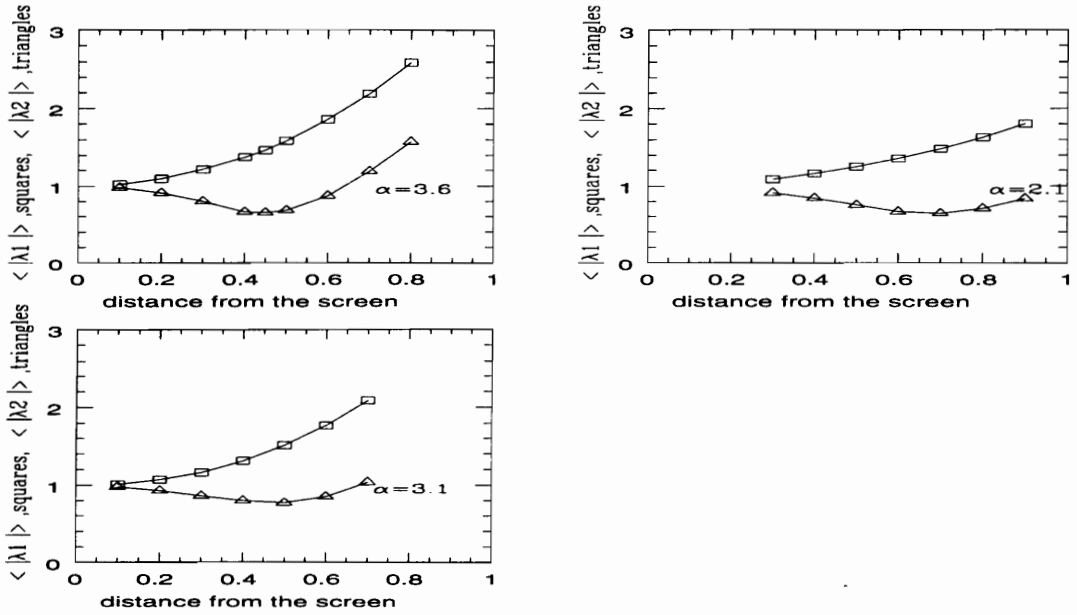


Fig. 11 Focal lengths

The alternative method of estimating the curvature was described in section 2.4.1, that is, the curvature is estimated from $\langle (\nabla^2 \phi)^2 \rangle$ using equation 27. These two methods yield comparable, but not exactly equal, results.

4.2 The interpolation algorithms

In order to calculate intensities at the observer's plane, there is a need for a surface interpolation algorithm, so as to assign values on the grid points at that plane. We used two different methods. For the weak focusing case, in which z is significantly smaller than the characteristic focal length, we calculate the curvature of the phase screen at all grid points, as well as the first derivatives, and using the transformation matrix, we can calculate the intensity at the points with coordinates (x', y') using equation 61. These points are not located on grid points any more, due to the deflection of the rays. (See figure 10, for example.) In other words, since we have followed the light rays all the way to the observer's plane, the arbitrary (not on grid points) positions of the rays are known, as well as the intensity assigned to each

ray. We can then use an existing interpolation algorithm (Renka, 1984) and calculate the intensities on the grid points.

In strong focusing, one has to account for ray crossings, and calculate the intensity not by equation 61, but by accumulating the rays landing in an area, and assigning intensity values to the corresponding area as high (focusing), or low (defocusing) respectively. Thus, we have to treat the strong focusing case, where we have a lot of ray crossing events, differently. For this case, we have treated the problem topologically, in order to account for caustics and folds in the phase screen, since for intensity calculations one is interested in adding the values of the data points that are in the same unit area, not averaging them. We calculate the intensity by looking at a grid square in the phase screen, and follow the rays that emerge from each corner (grid point) of the square to the observer's plane. (See figure 10, for example.) The arrival points of the rays are noted as crosses in figure 10. The four arrival points now form a quadragon on the observer's plane, and the intensity assigned to the quadragon will be given by

$$I = \frac{1}{\frac{1}{2}|((\vec{a} \times \vec{b}) + (\vec{c} \times \vec{d}))|} \quad (62)$$

where the area of the grid square in the phase screen is 1. The quadragon is defined by the clockwise sequence of vectors $\vec{a}, \vec{b}, \vec{c}, \vec{d}$ which bound the quadragon in the observer's plane. Its area is

$$A = \frac{1}{2}|((\vec{a} \times \vec{b}) + (\vec{c} \times \vec{d}))| \quad (63)$$

Deformations of the quadragon can be seen in the top right of figure 10, and are caused by ray crossings and folds of the manifold. A schematic representation can be seen in the fortran subprogram 'cell-interpol.f' and the subroutines therein in Appendix B.6. The result of the surface interpolation algorithm can be seen in figure 12 at all frequencies, and it is obvious that the refraction hypothesis behaves very well, as we can see the intensity diminishing with increasing frequency.

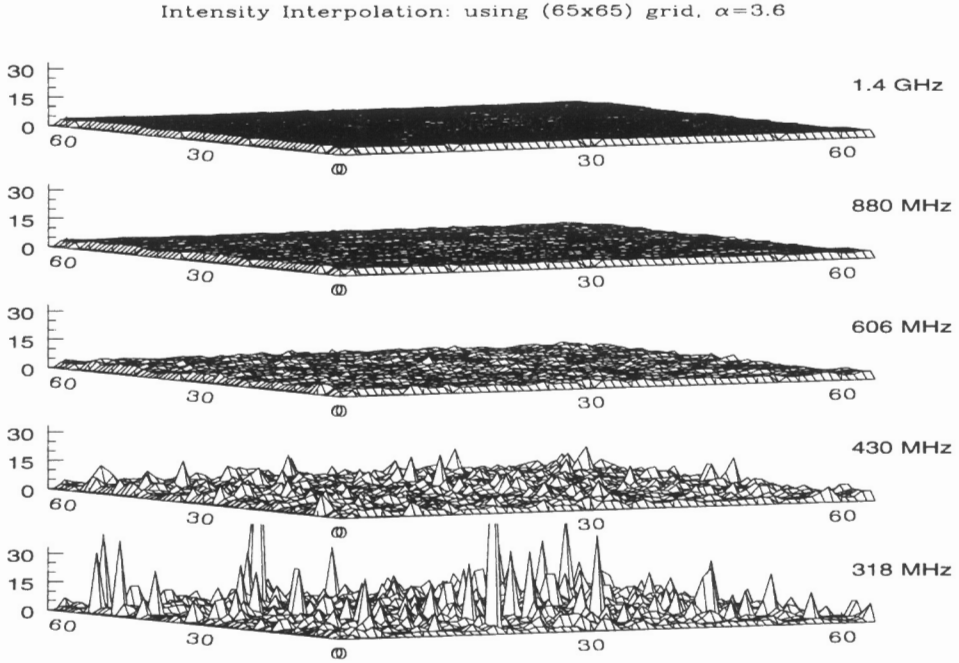


Fig. 12 Intensity at the observer's plane at 318,430, 606,880,1400 MHz, arbitrary units.

Since in the strong focusing case we are accounting for ray crossings that come about from the various deformations (folds) of the phase manifold, we must recover intensities otherwise lost because too many rays may have accumulated on a particular spot on the observer's plane, a spot that is smaller in area than our grid unit square. Thus, there is need for zooming in, by imposing a much finer grid, and recovering lost information. After imposing this grid, we then convolve the image with a Gaussian, of relevant FWHM (full-width-at-half-maximum), and get images of the normalized intensity at the observer's plane. We then calculate trajectories, in simulating Earth's projected motion, and take light curves. We can see one of the images, figure 13, produced using a Kolmogorov power law, ($\alpha = 11/3$), with an eightfold zooming, and at frequency 318 MHz. There are many hot spots, from ray focusing, as well as areas of defocusing. Light curve from this image, as well as of

the corresponding images at frequencies 430,606,880 and 1400 MHz can be seen in figures 14 – 18.

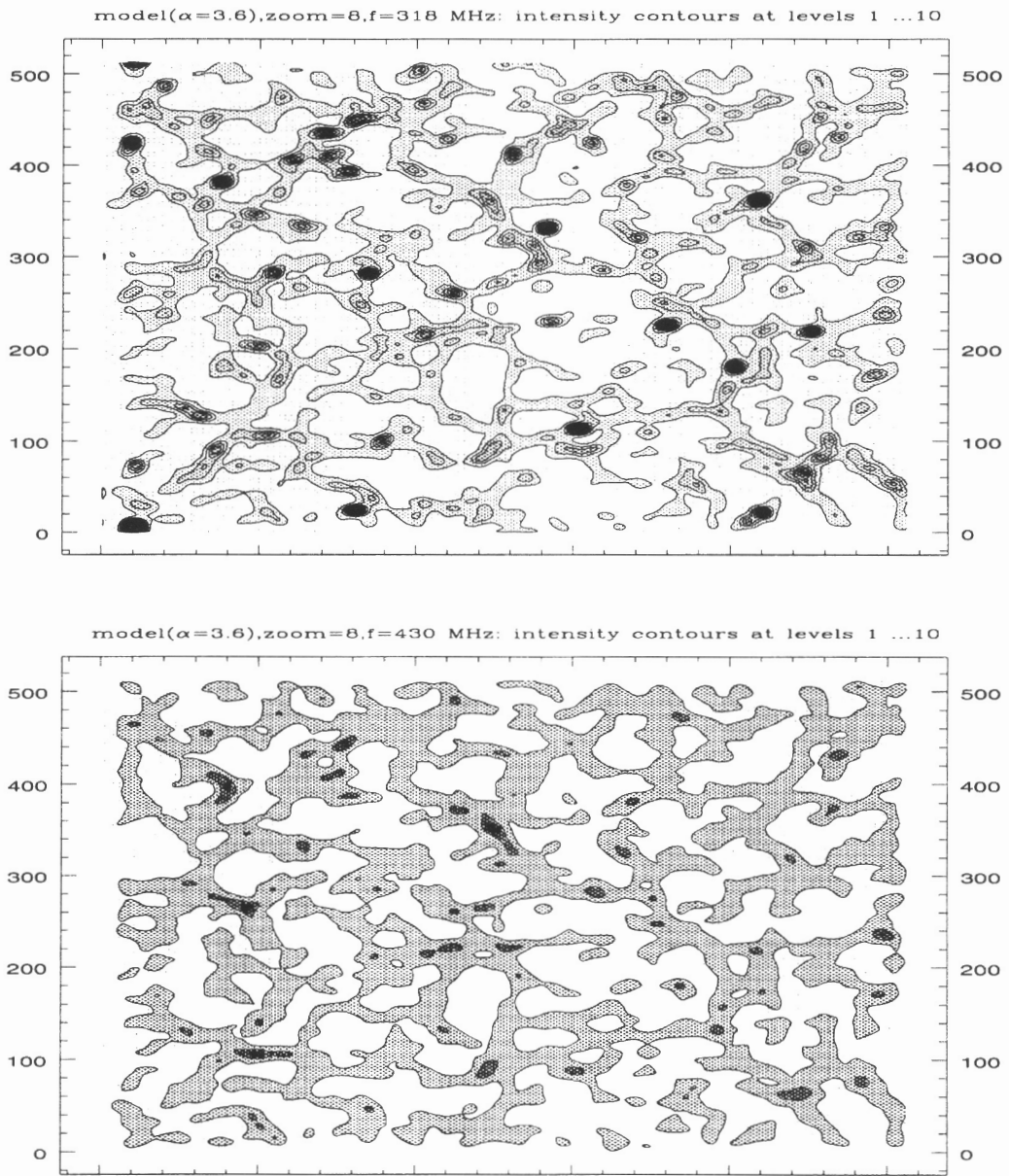


Fig. 13 Images of intensity at the observer's plane at 318 and 430 MHz

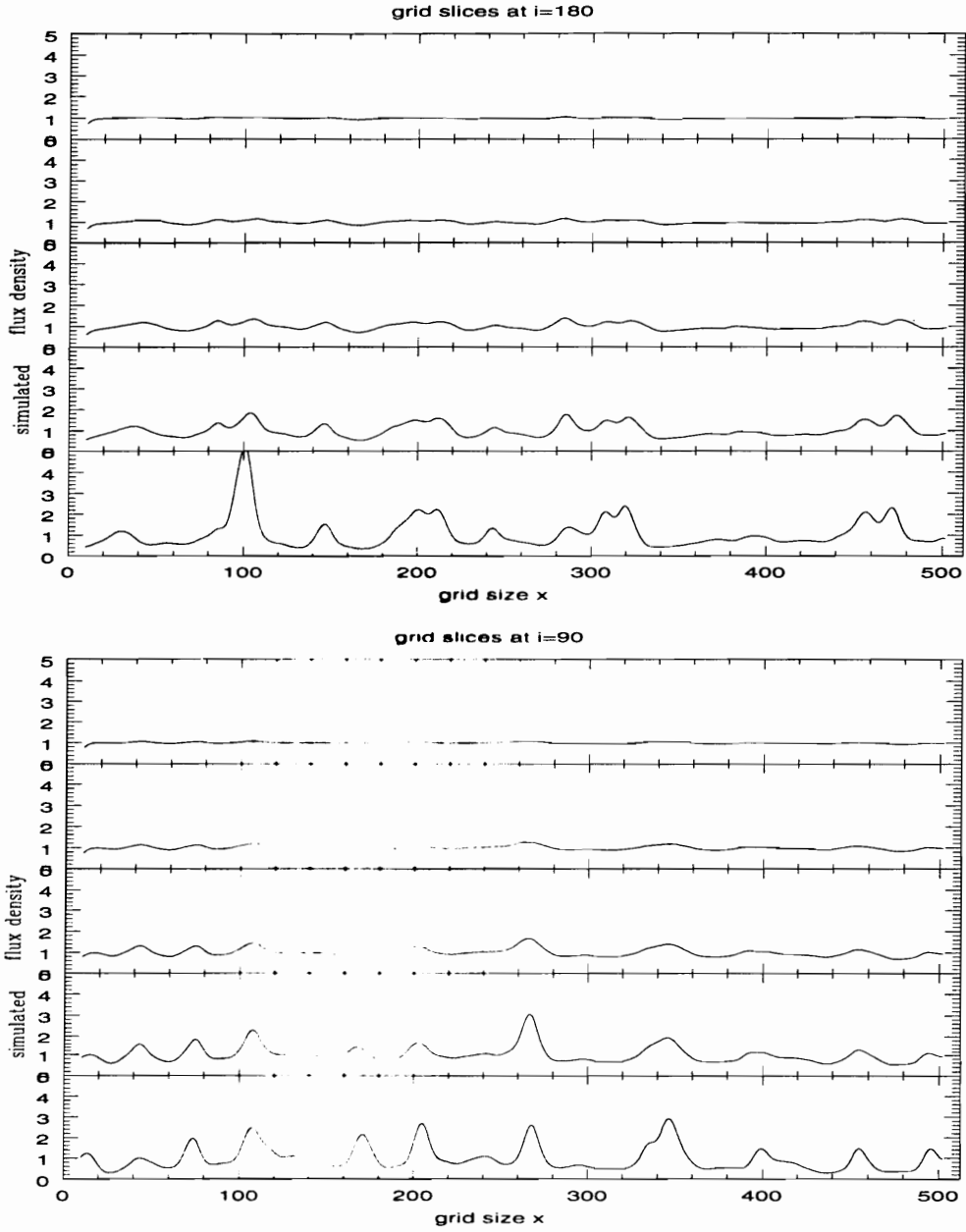


Fig. 14 Simulated light curves at the observer's plane at all 5 frequencies

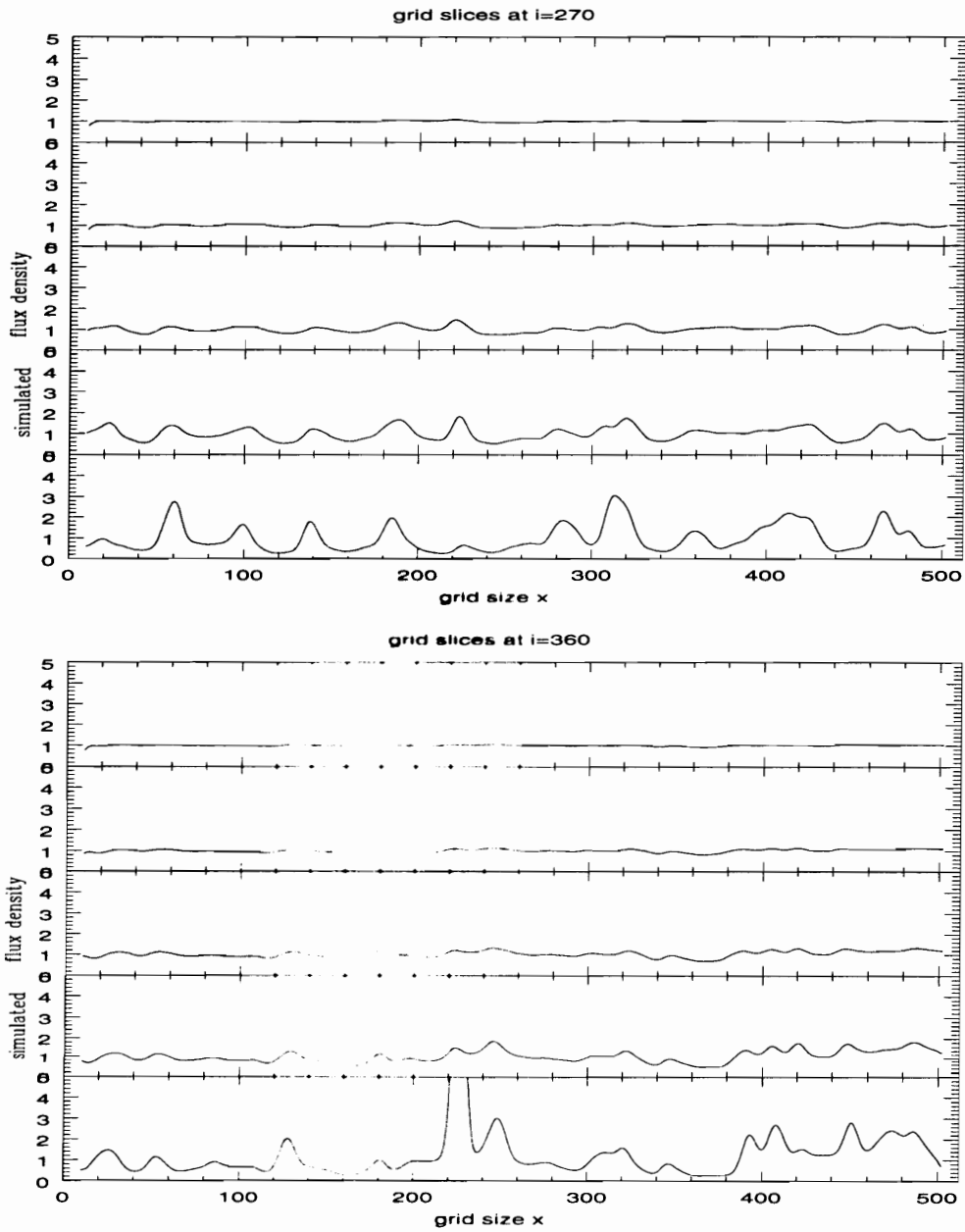


Fig. 15 Simulated light curves at the observer's plane at all 5 frequencies

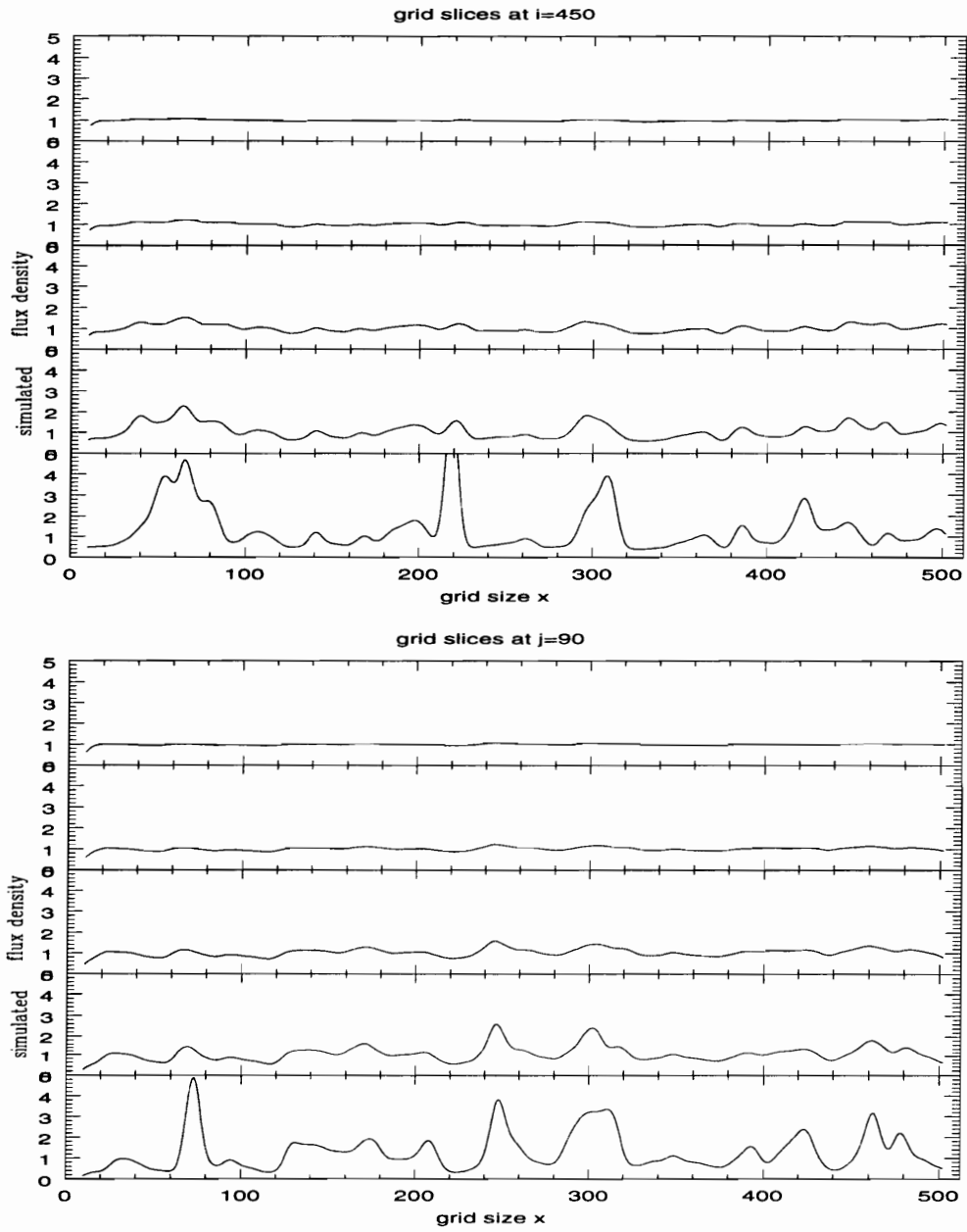


Fig. 16 Simulated light curves at the observer's plane at all 5 frequencies

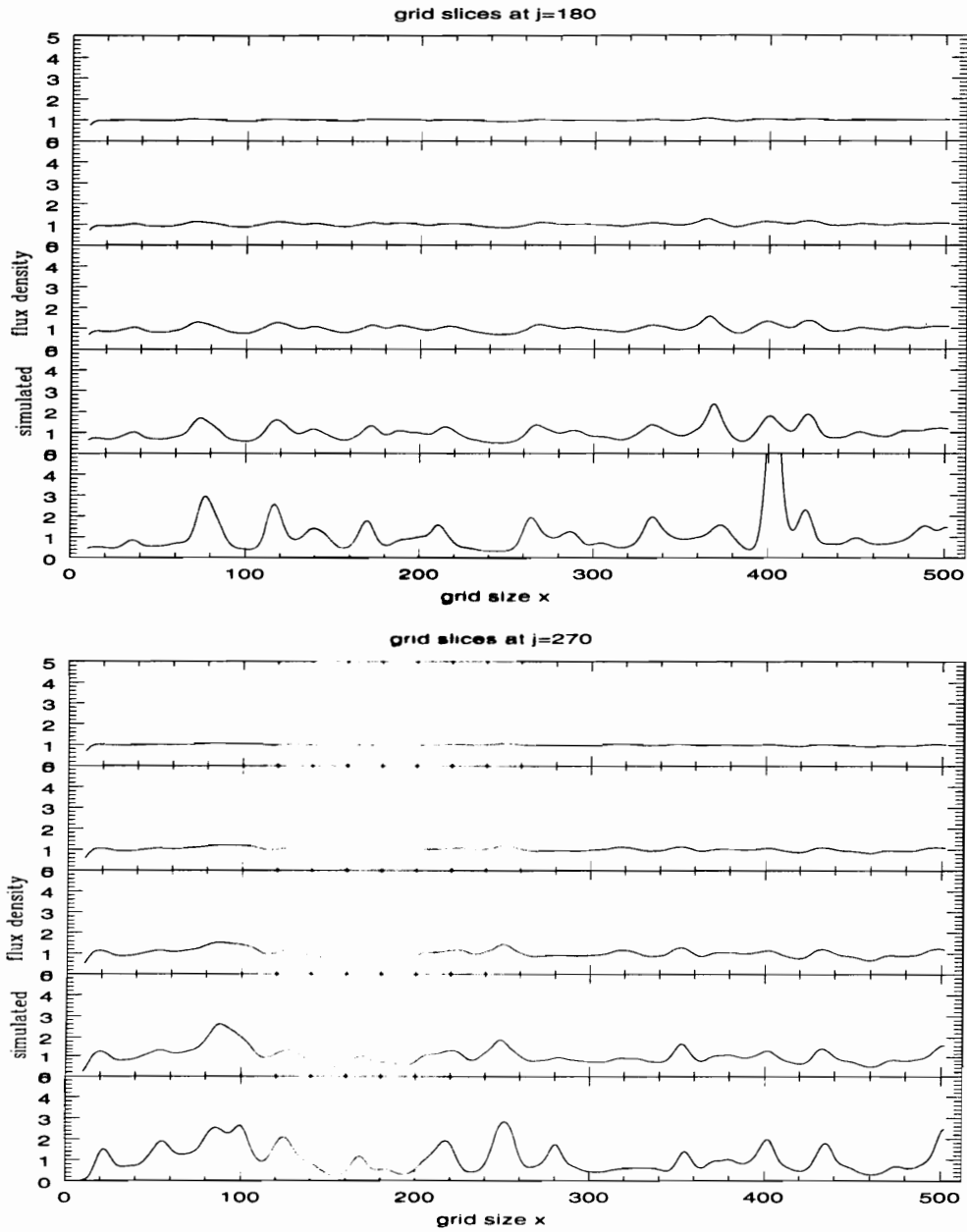


Fig. 17 Simulated light curves at the observer's plane at all 5 frequencies

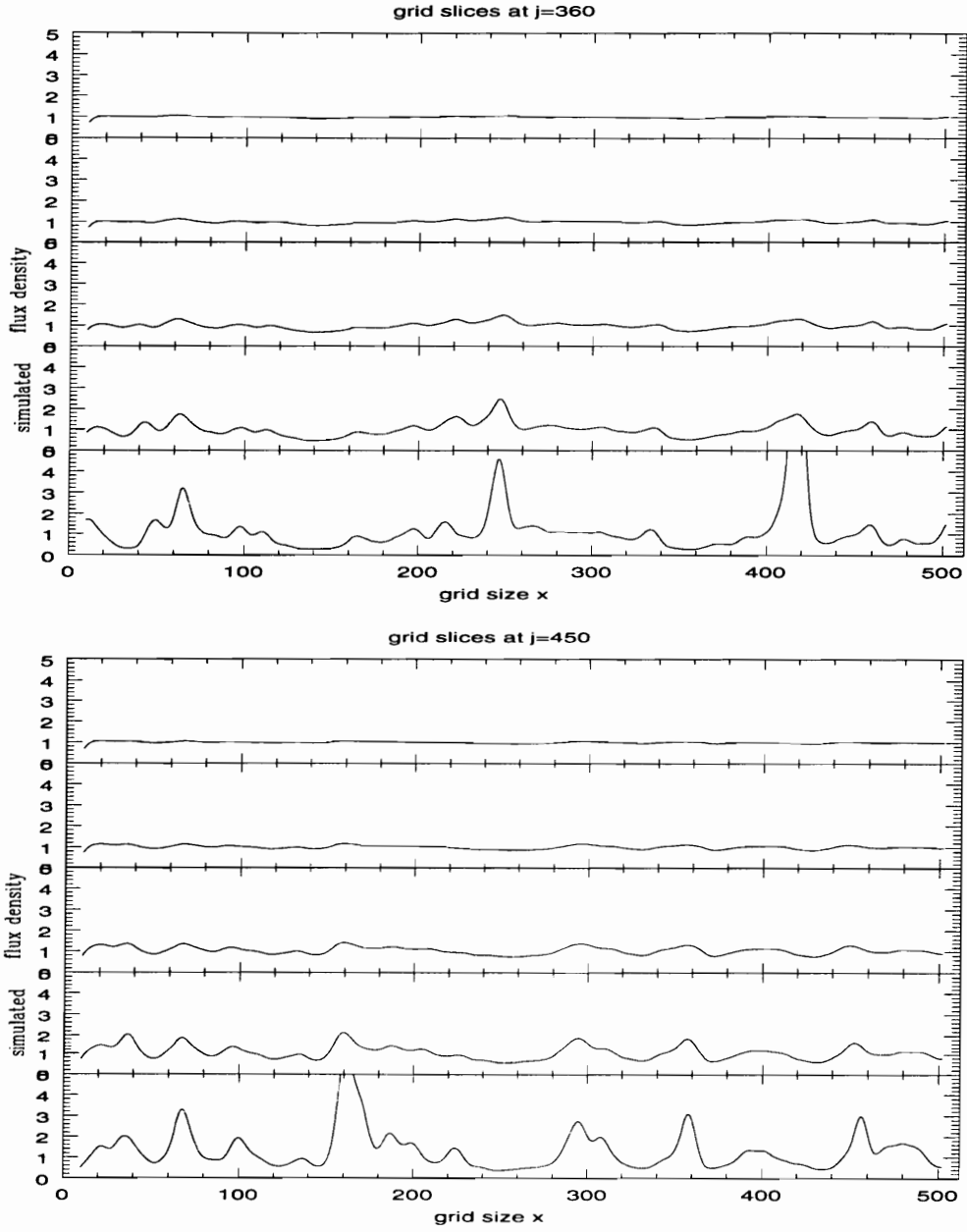


Fig. 18 Simulated light curves at the observer's plane at all 5 frequencies

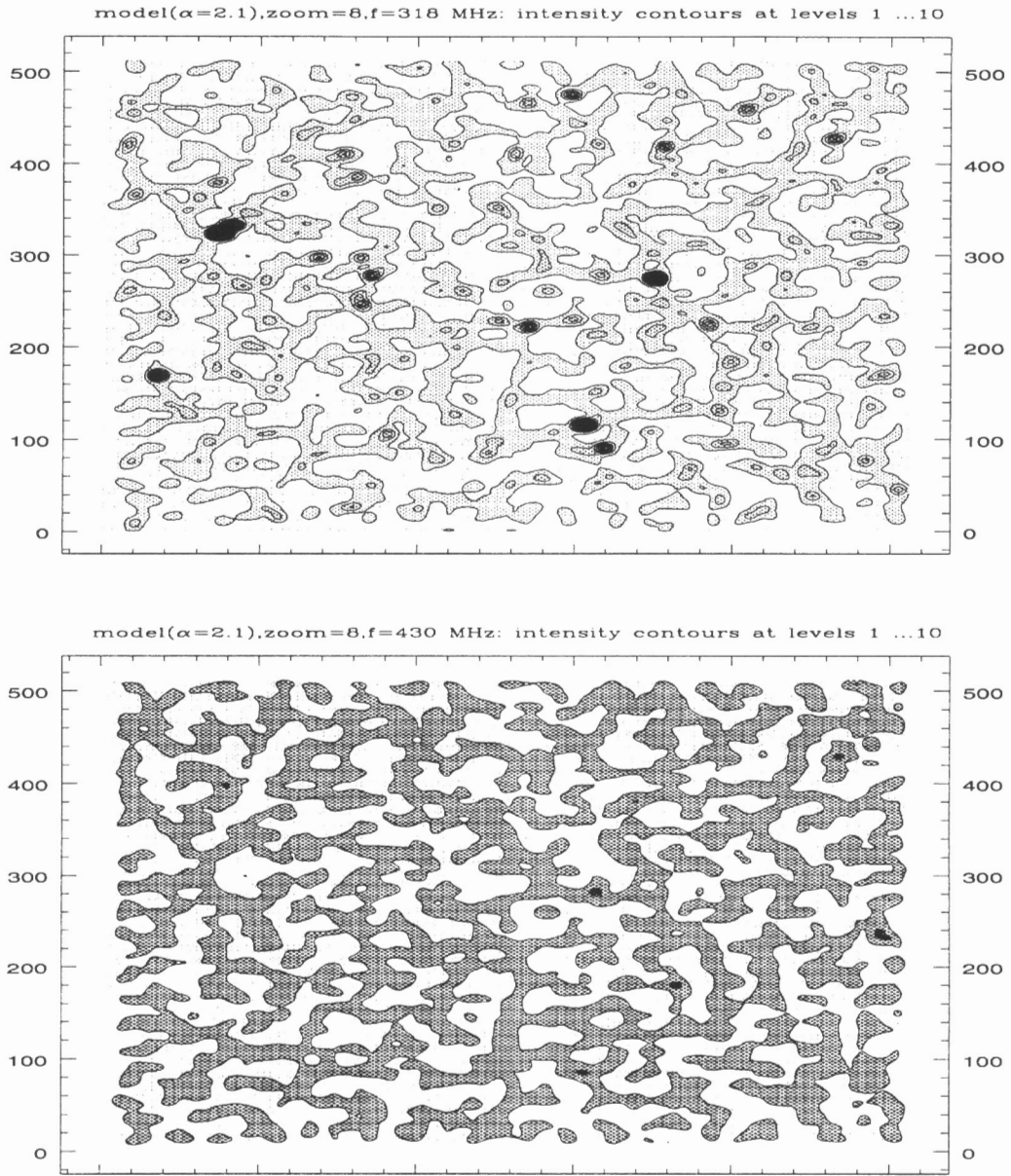


Fig. 19 Images of intensity at the observer's plane at 318 and 430 MHz

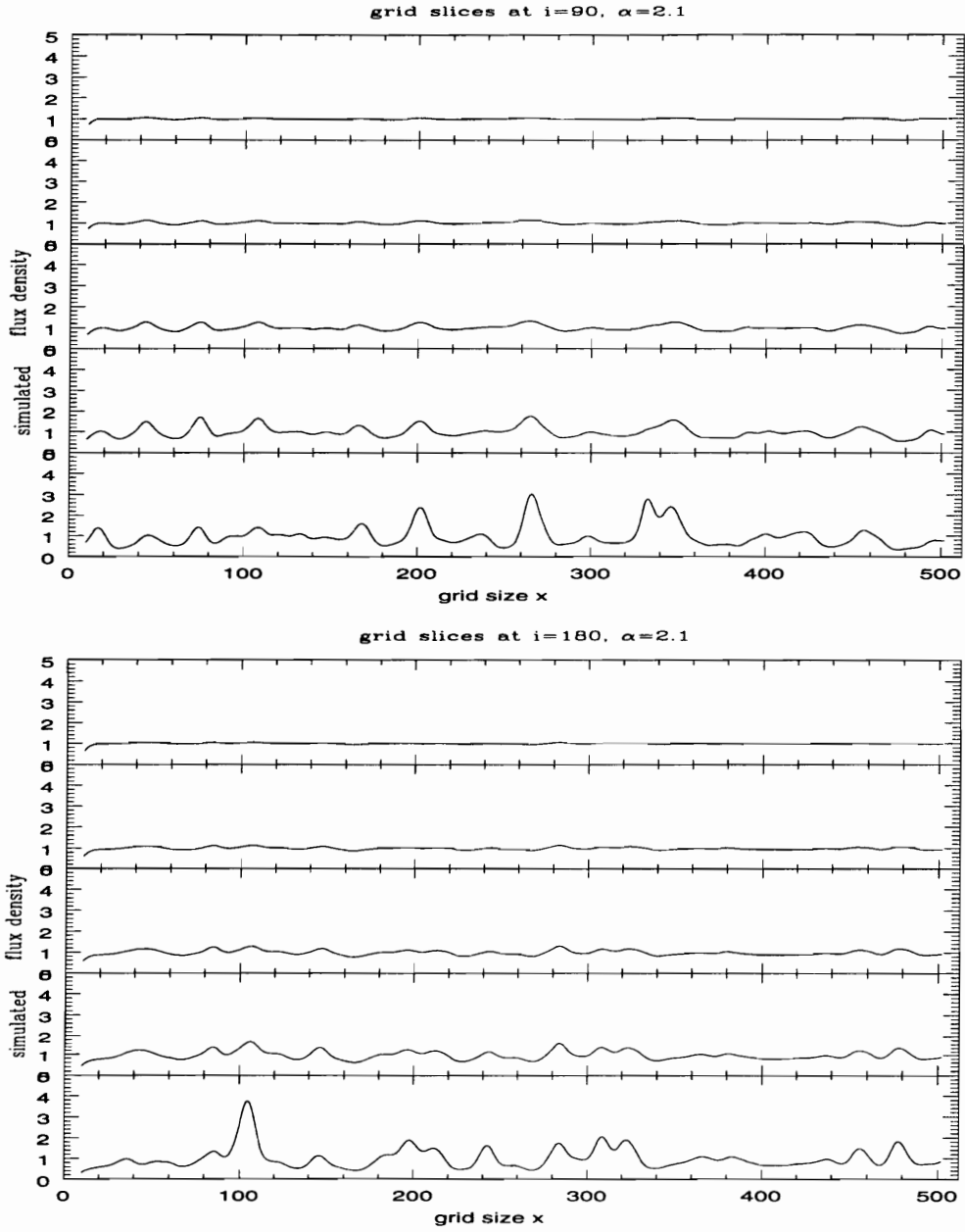


Fig. 20 Simulated light curves at the observer's plane at all 5 frequencies

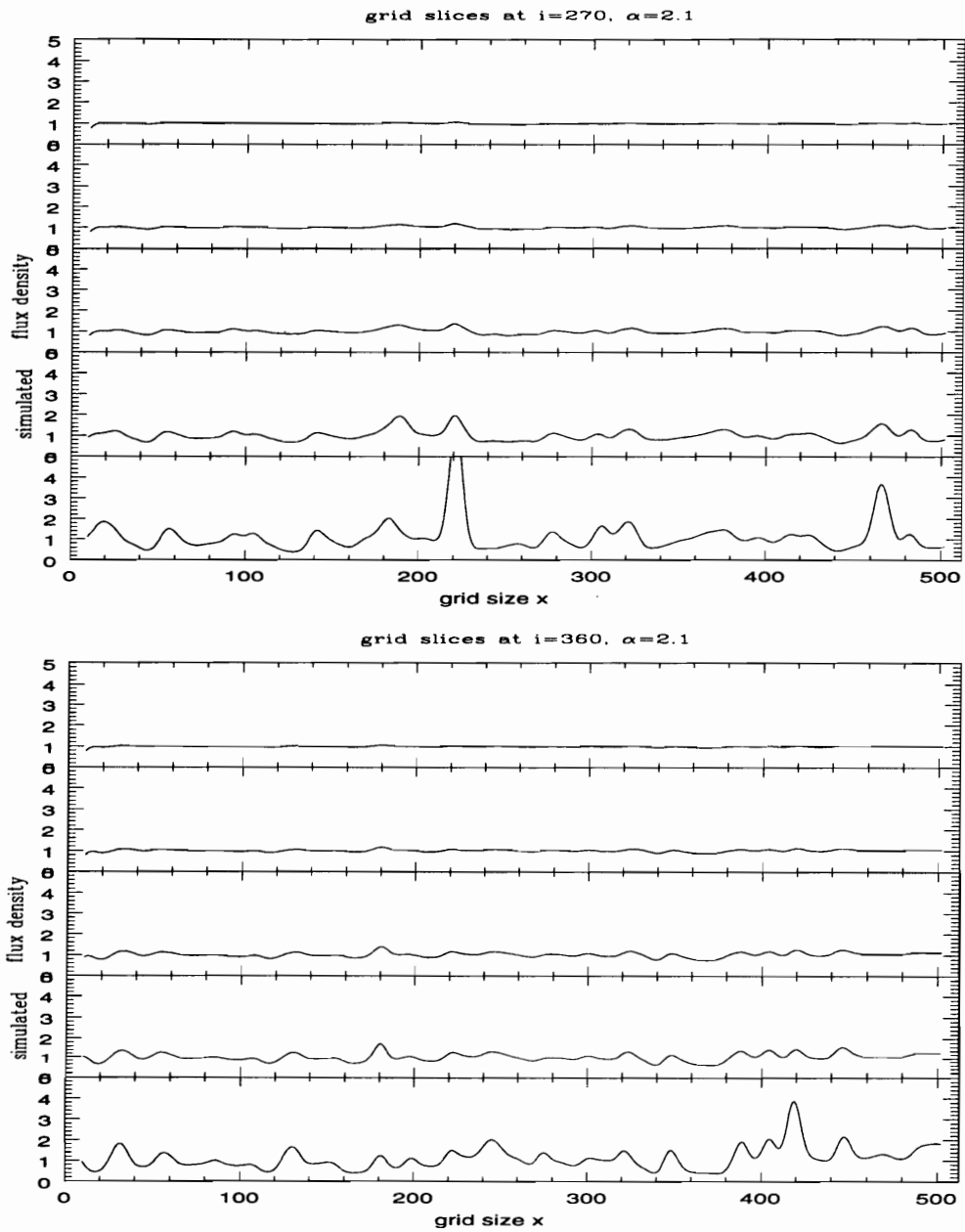


Fig. 21 Simulated light curves at the observer's plane at all 5 frequencies

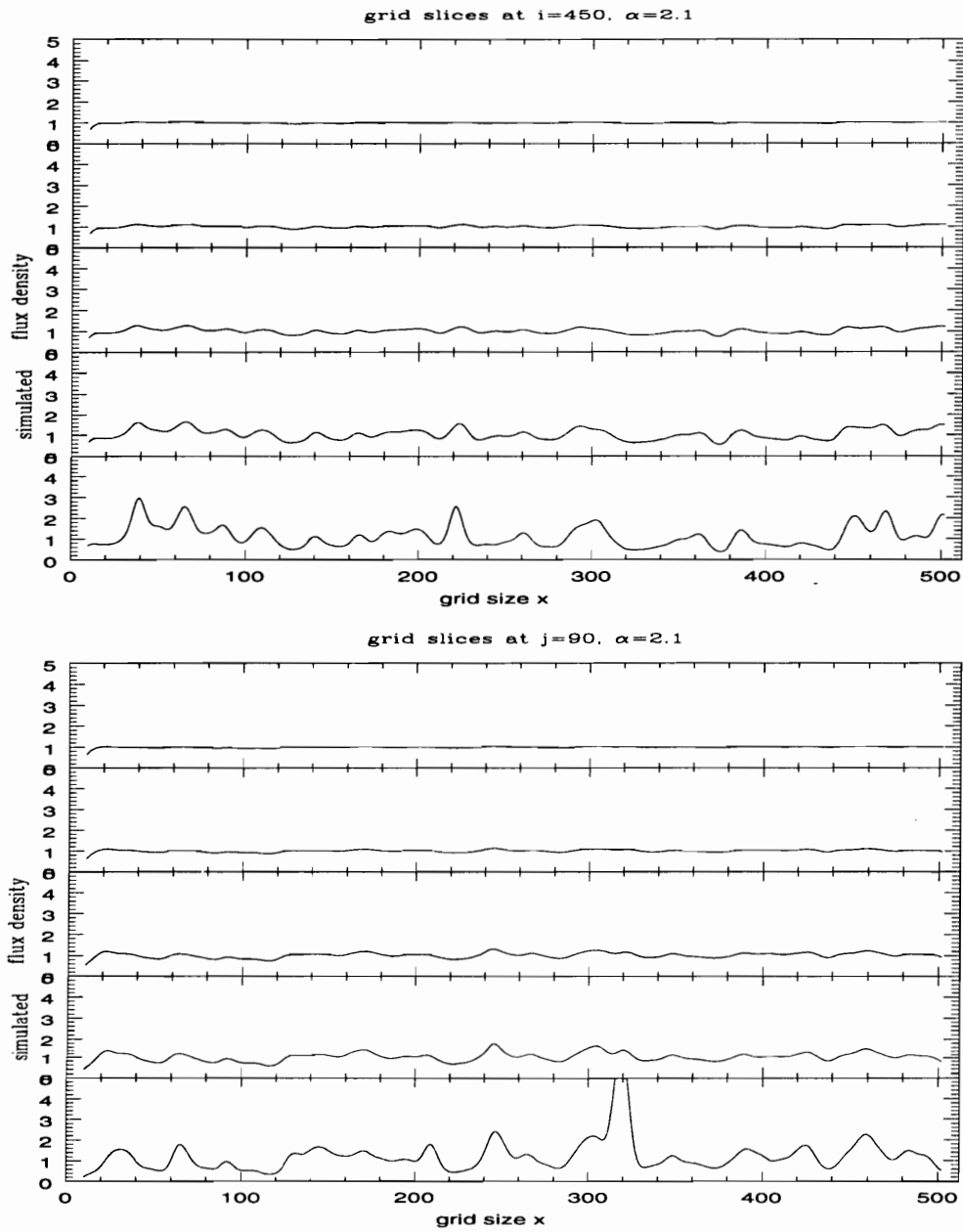


Fig. 22 Simulated light curves at the observer's plane at all 5 frequencies

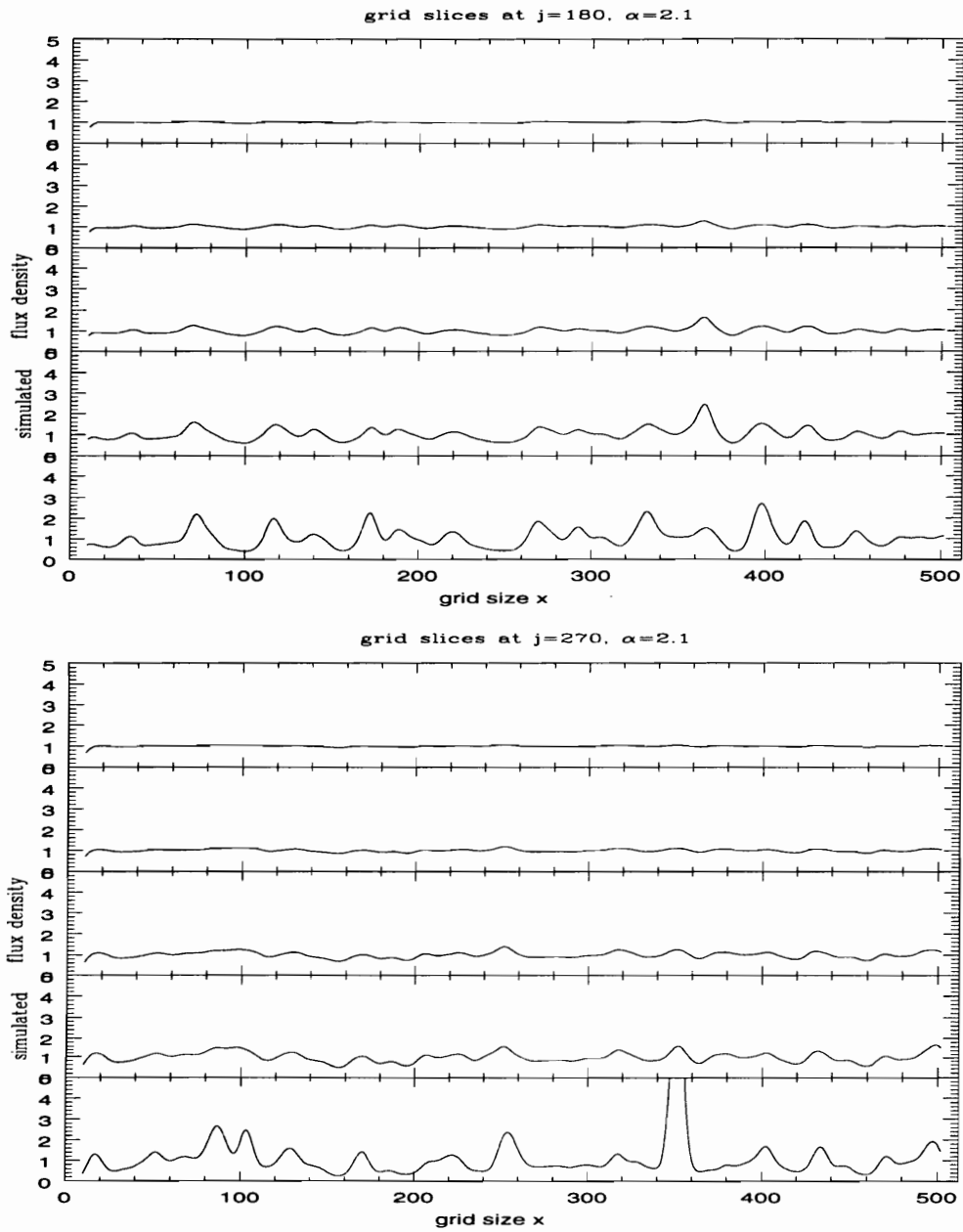


Fig. 23 Simulated light curves at the observer's plane at all 5 frequencies

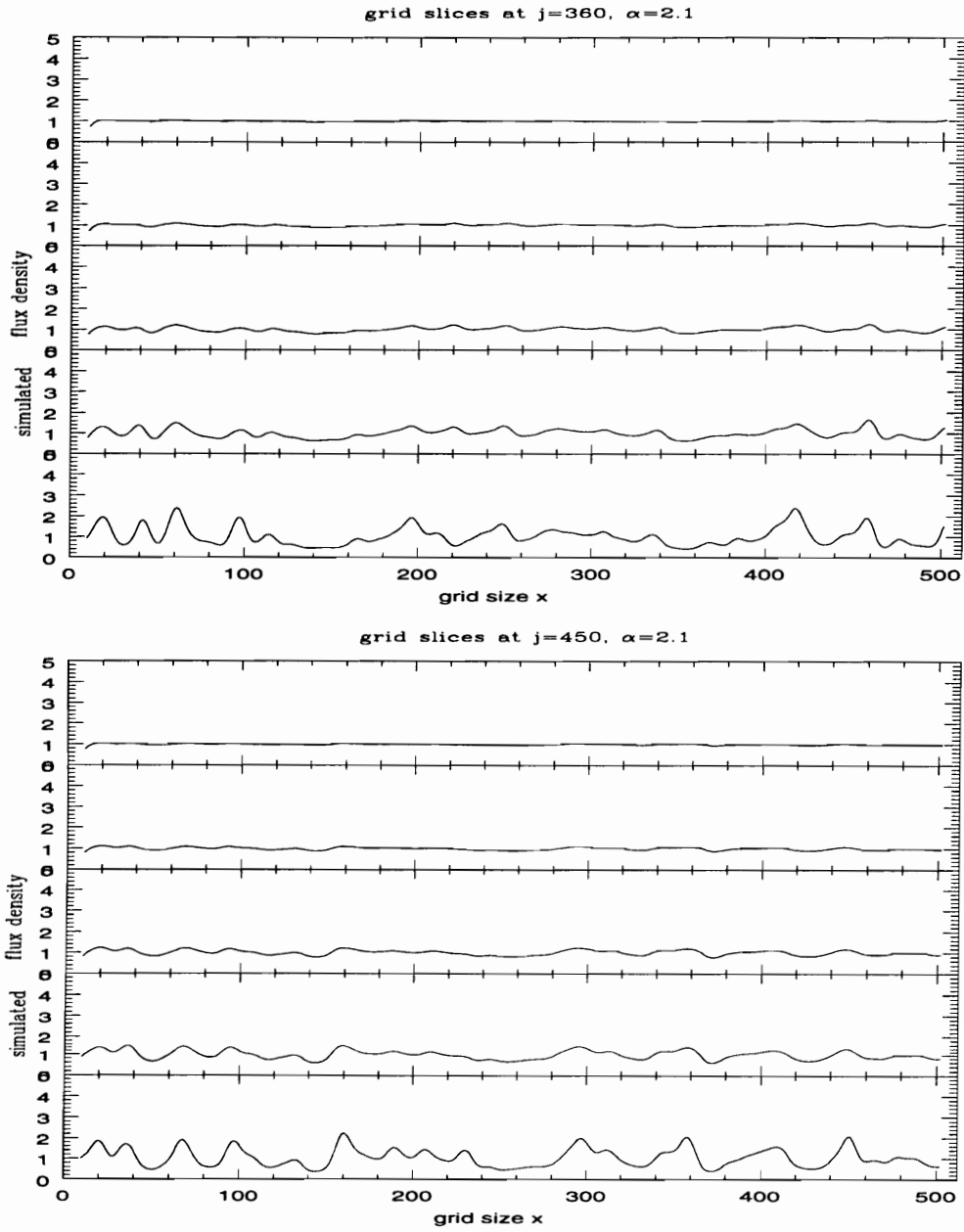


Fig. 24 Simulated light curves at the observer's plane at all 5 frequencies

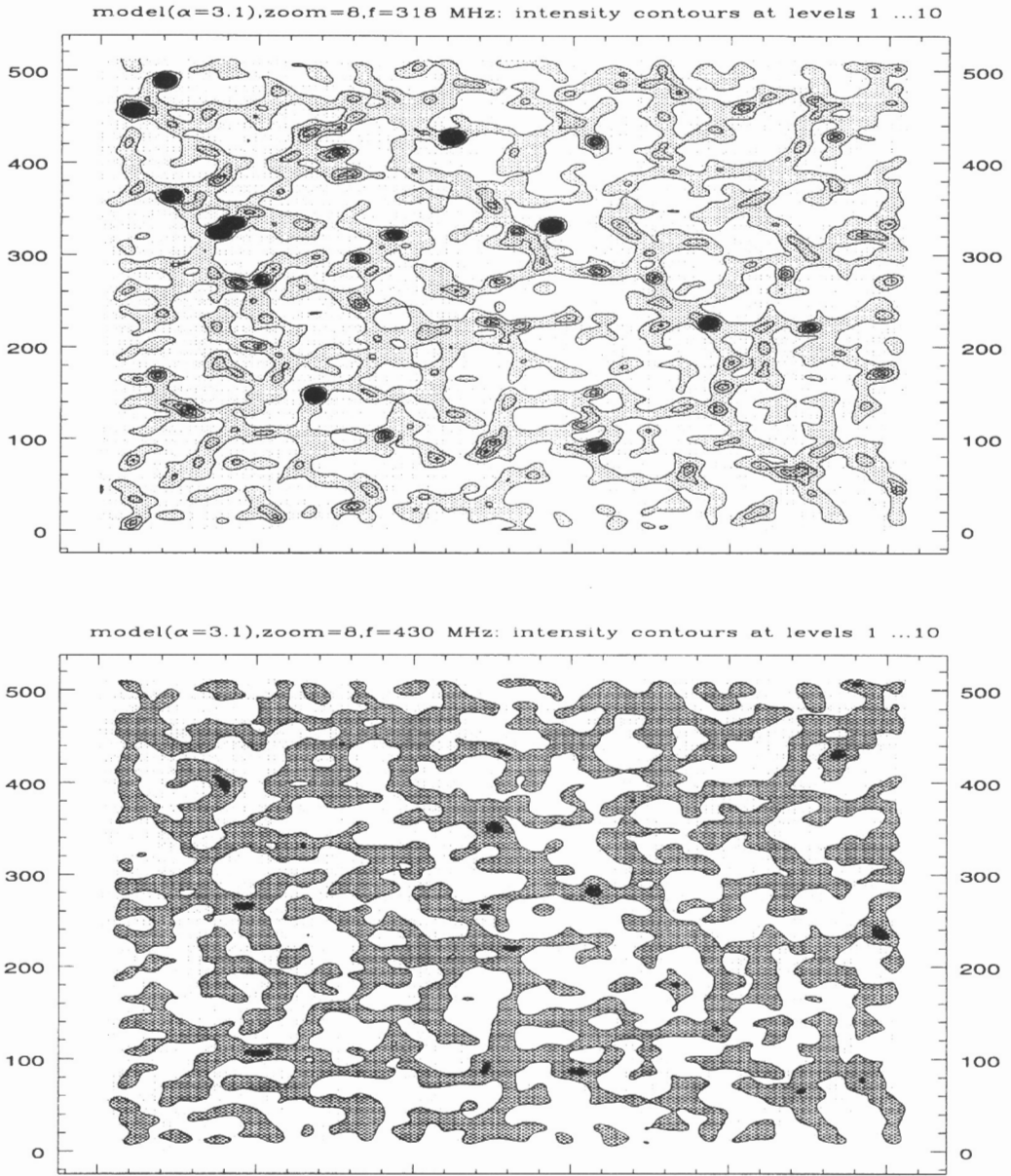


Fig. 25 Images of intensity at the observer's plane at 318 and 430 MHz

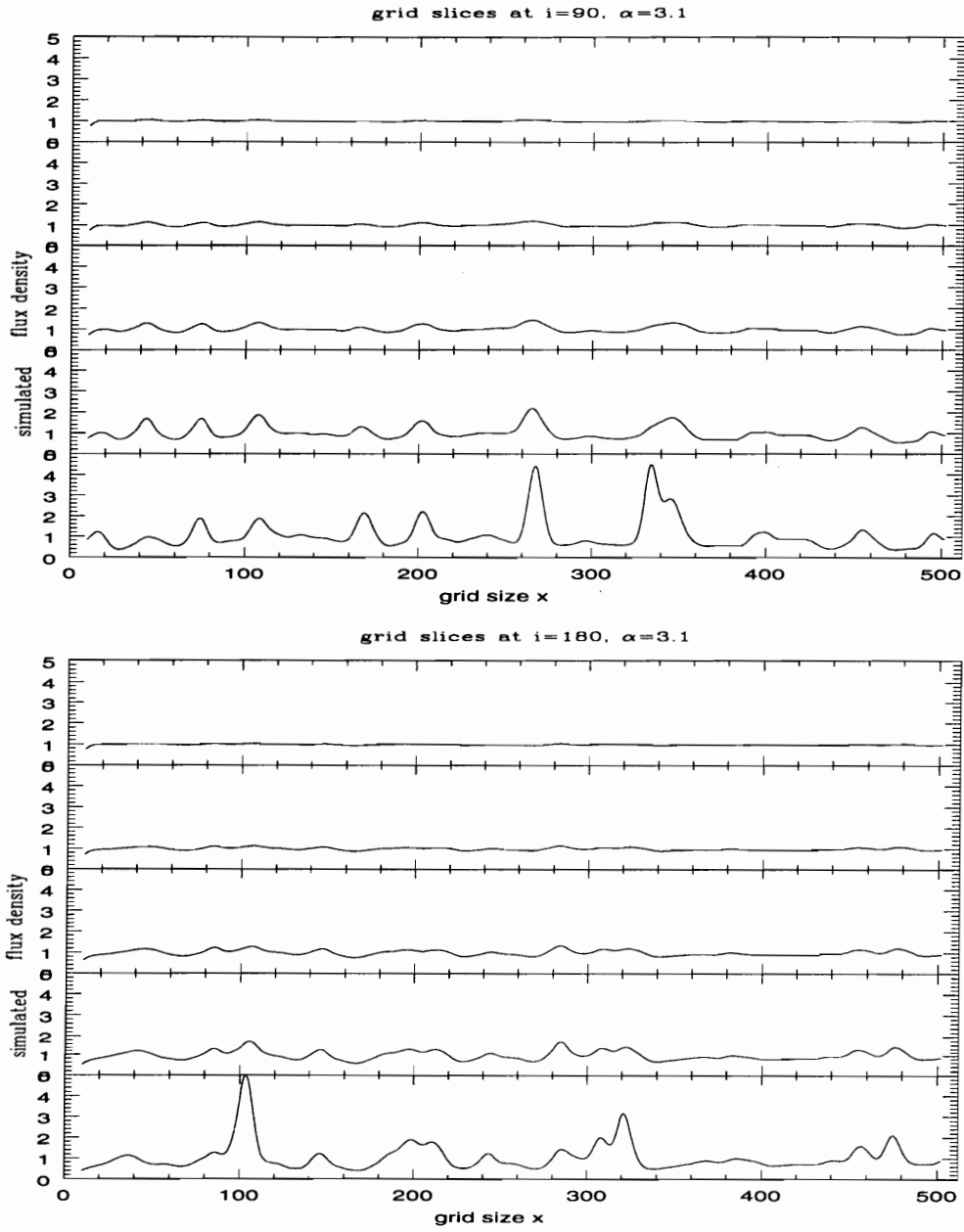


Fig. 26 Simulated light curves at the observer's plane at all 5 frequencies

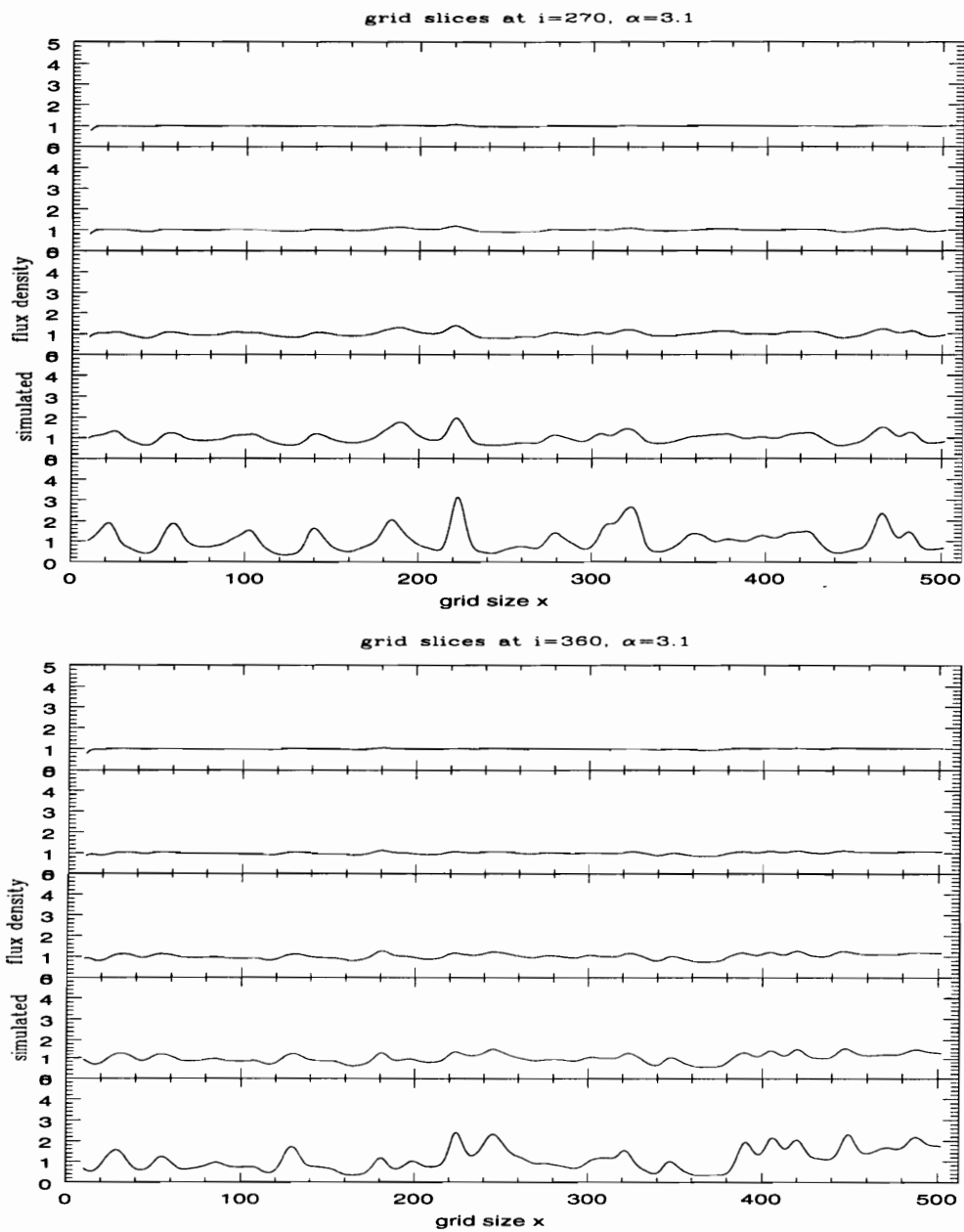


Fig. 27 Simulated light curves at the observer's plane at all 5 frequencies

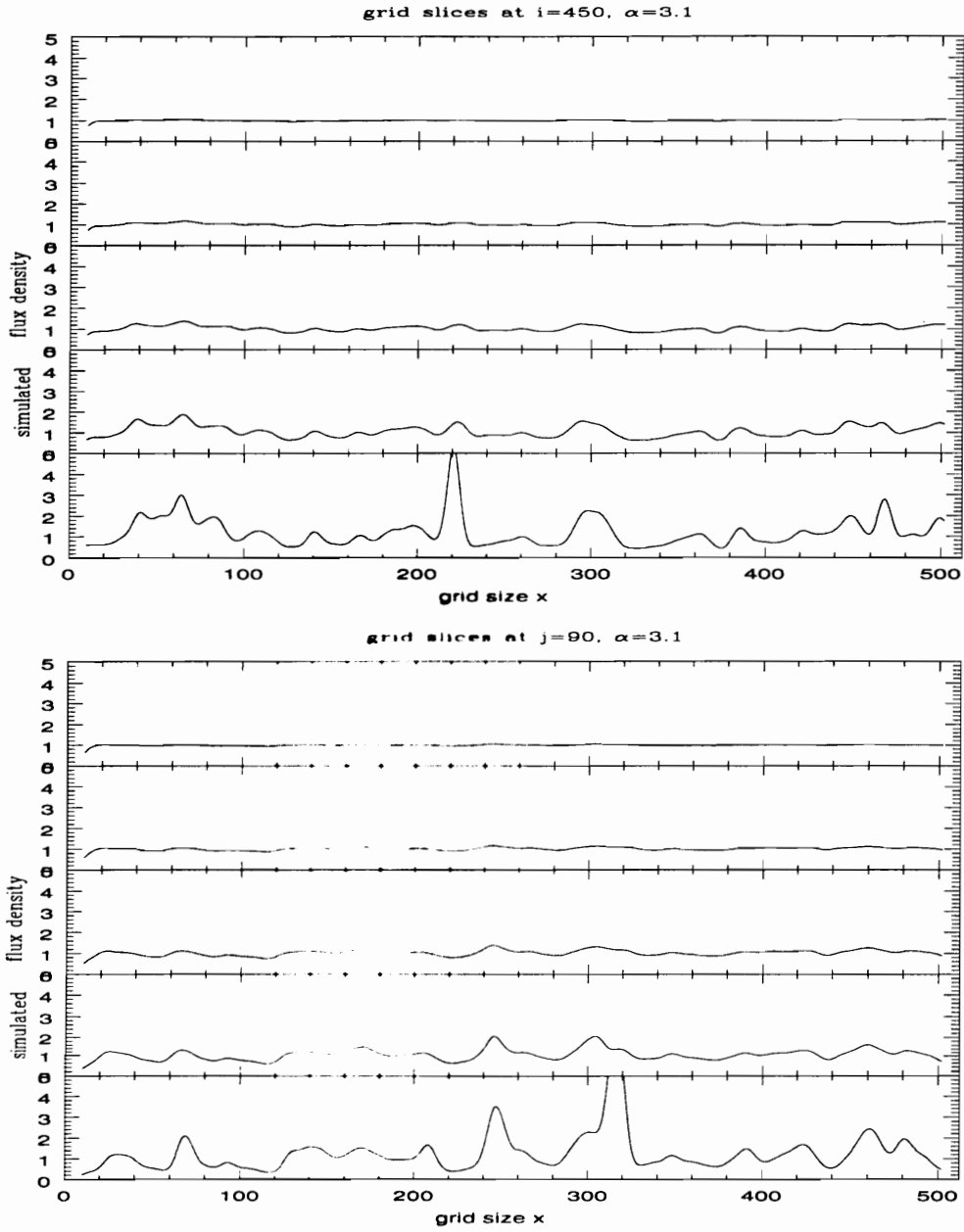


Fig. 28 Simulated light curves at the observer's plane at all 5 frequencies

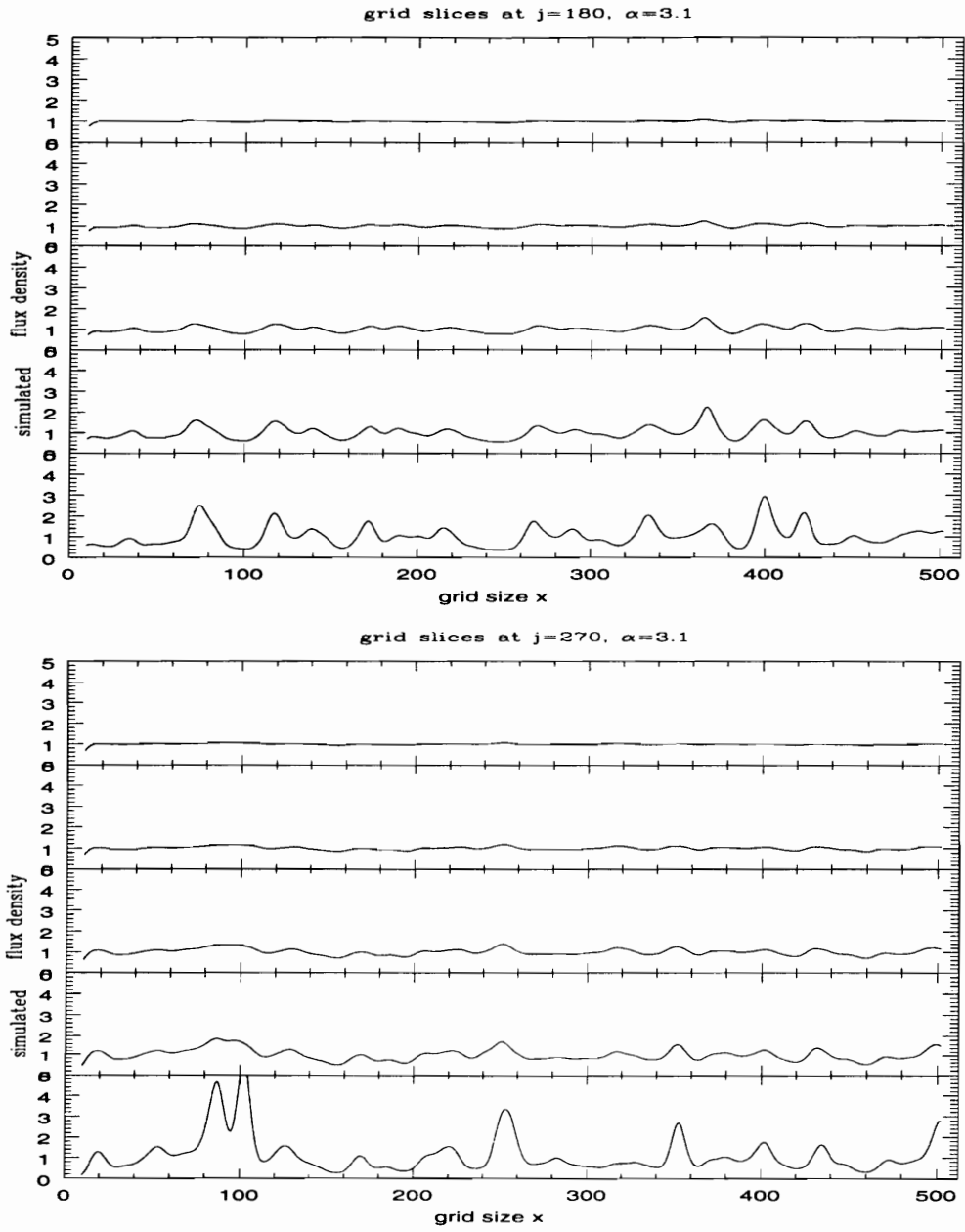


Fig. 29 Simulated light curves at the observer's plane at all 5 frequencies

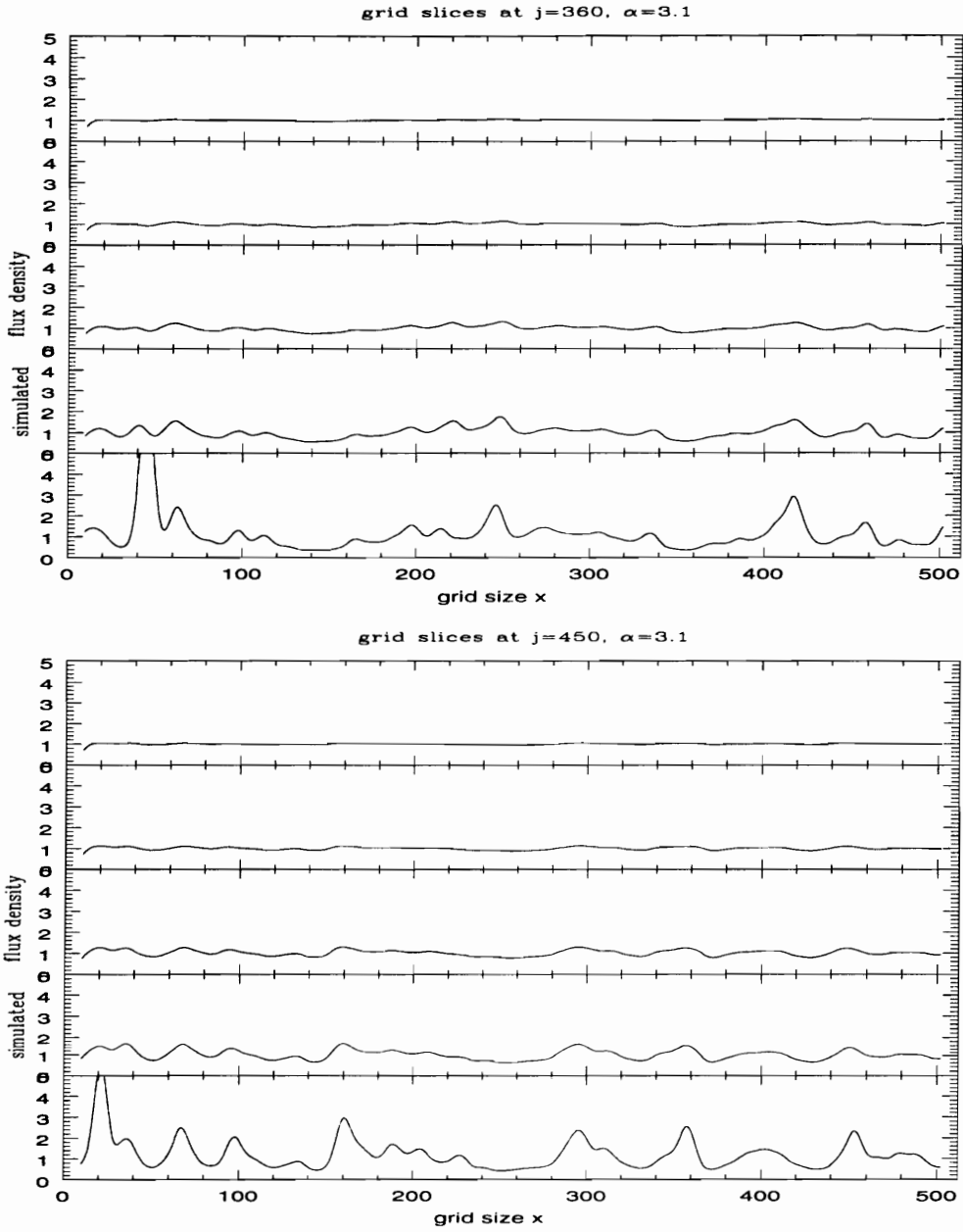


Fig. 30 Simulated light curves at the observer's plane at all 5 frequencies

Chapter 5

Comparison with data

We compare the simulated light curves, (see figures 14, 24, for example) with light curves from observations (Mitchell et. al., 1994), by calculating the auto-structure functions at each frequency, and the cross-structure functions between 430 MHz and the remaining four frequencies. The structure function of $f(x)$ is defined as

$$D_f(\tau_l) = \langle (f(x + \tau_l) - f(x))^2 \rangle \quad (64)$$

where τ_l is the lag. The structure function is normalized to unity. When D_f plotted versus τ_l levels out we are sure that we have enough data to characterize the problem, and we are accounting for all of the variance in the function.

We then take about 10 non-correlated trajectories in each image (figures 13 and 25 are examples of images) and we average the frequency equivalent auto-structure functions and cross-structure functions. (See figures 31, 32 and 33 for the model.) We repeat the procedure for each source from the observations, figures 34 – 50. To calculate the data structure functions though, we had to overcome a number of problems, most importantly that the data points were not equally spaced due to changes in the observation schedules, as well as gaps in the data, due to technical problems. Those problems were solved with the 'read-data-spline.f' and 'structure-function.f' programs (Appendix B). The program 'read-data-spline.f', reads data from the *Lowvar* data set (Michell et al. 1994) identifies missing points, does a cubic spline interpolation, and then selects data points at regular intervals. The subprogram 'structure-function.f' then calculates the structure functions seen in figures 34 – 50.

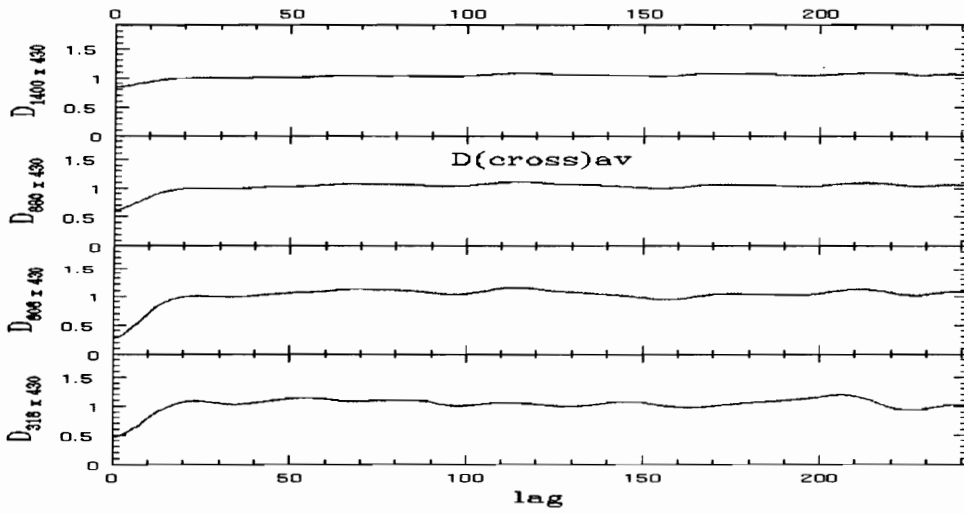
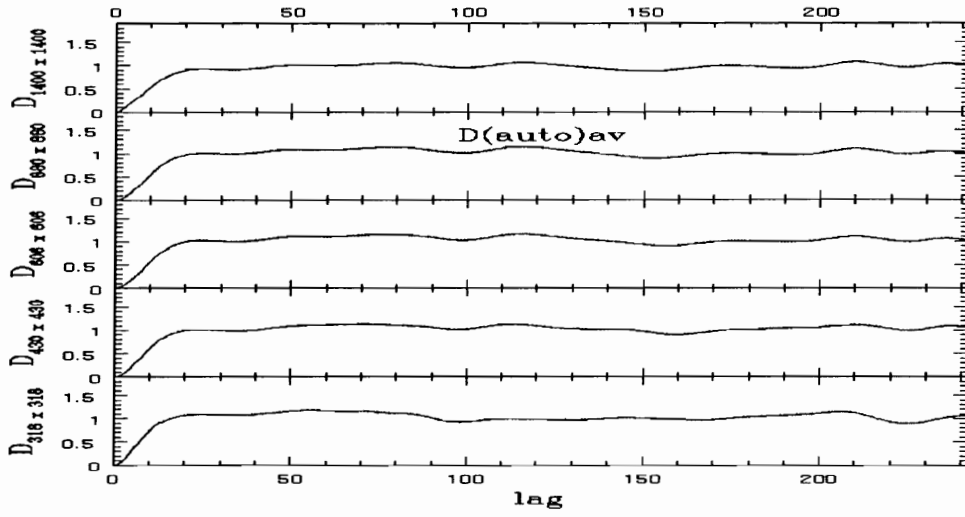


Fig. 31 Auto and cross structure-functions, from the model for $\alpha = 3.6$

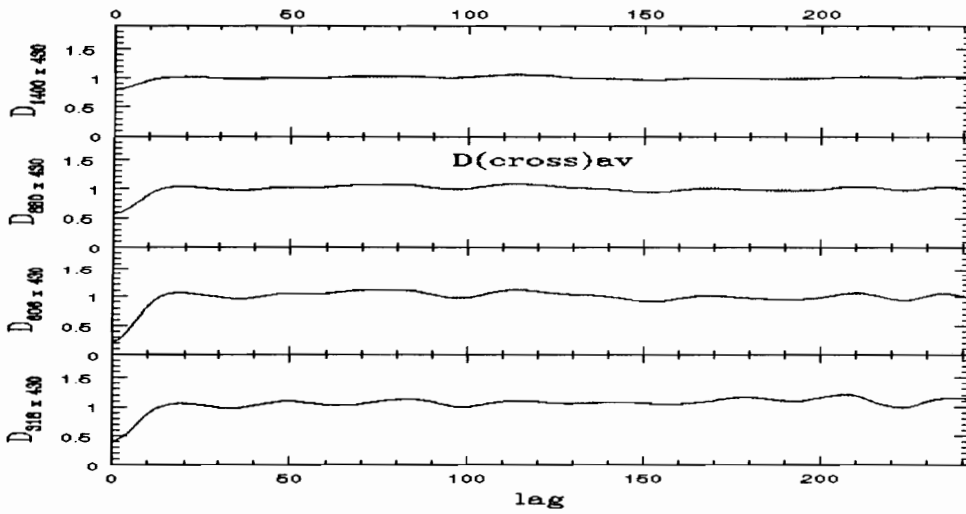
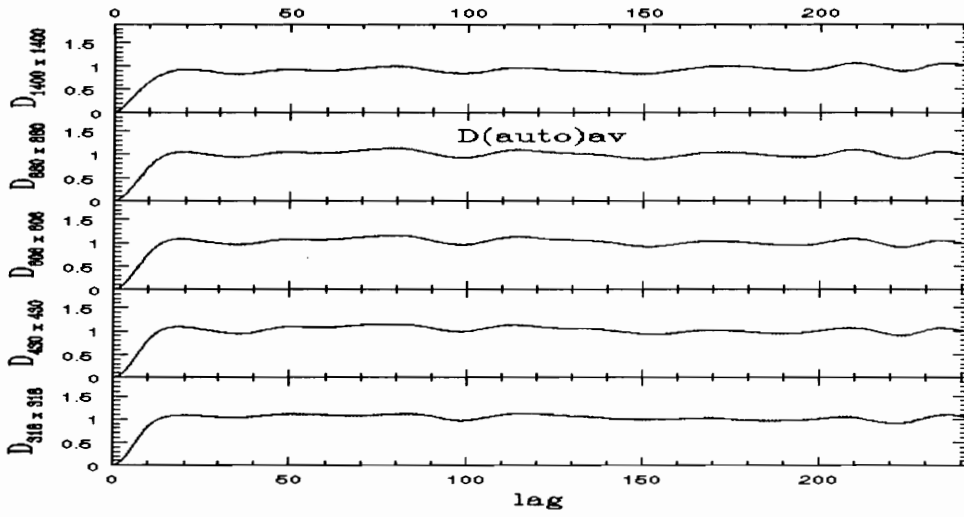


Fig. 32 Auto and cross structure-functions, from the model for $\alpha = 2.1$

Comparison with data

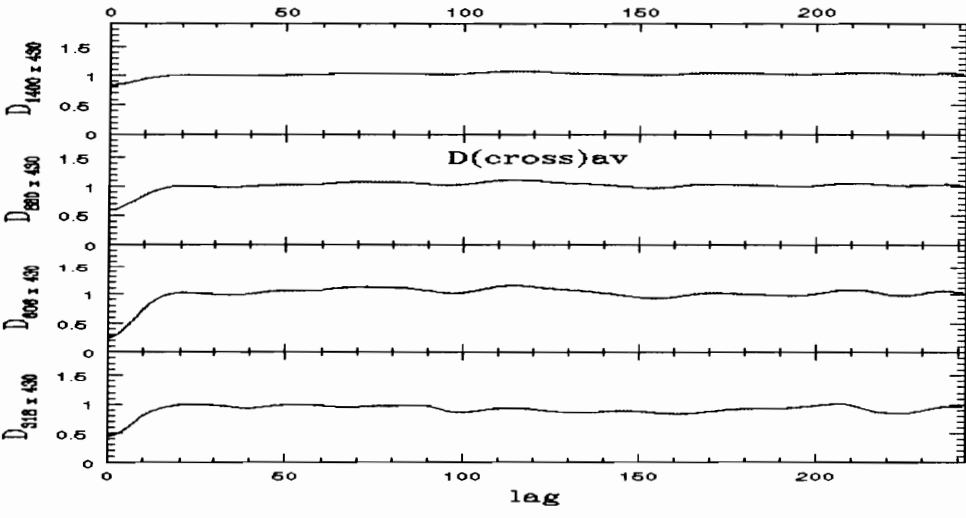
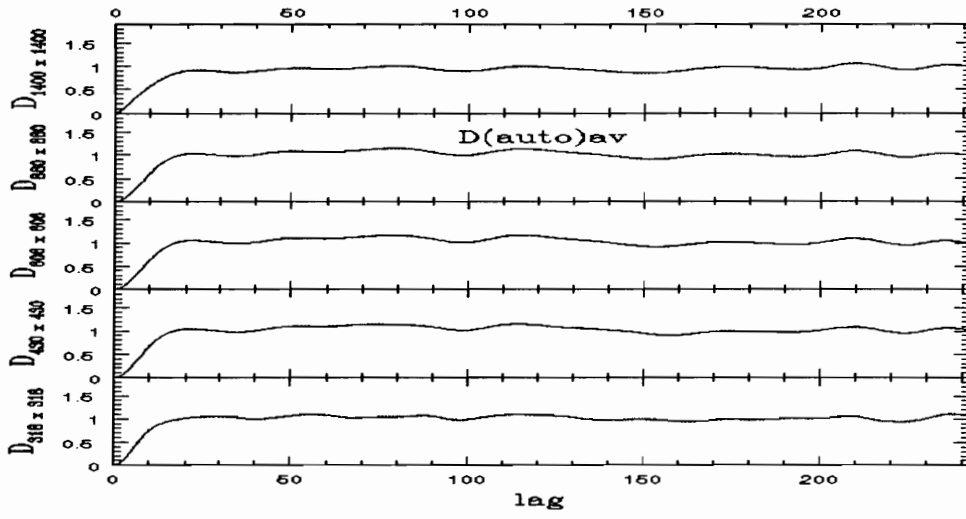


Fig. 33 Auto and cross structure-functions, from the model for $\alpha = 3.1$

Comparison with data

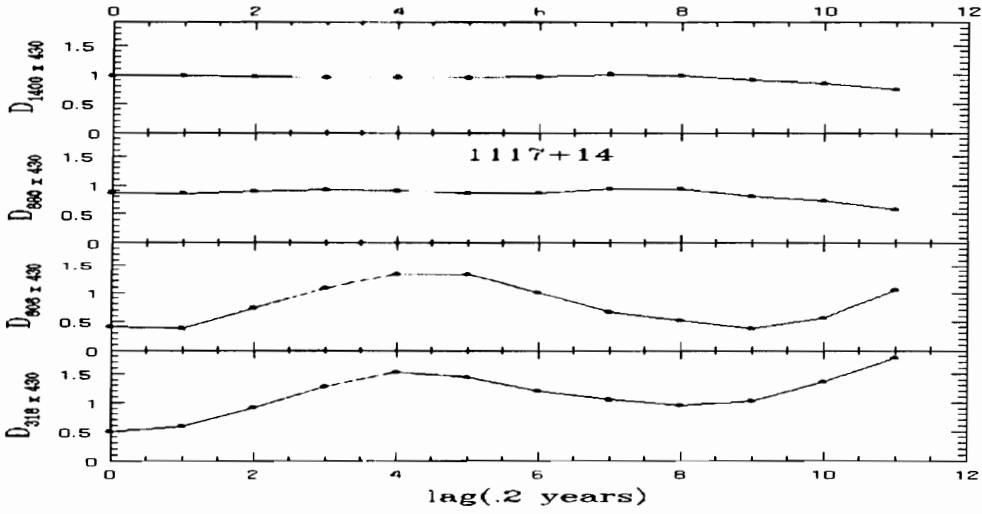
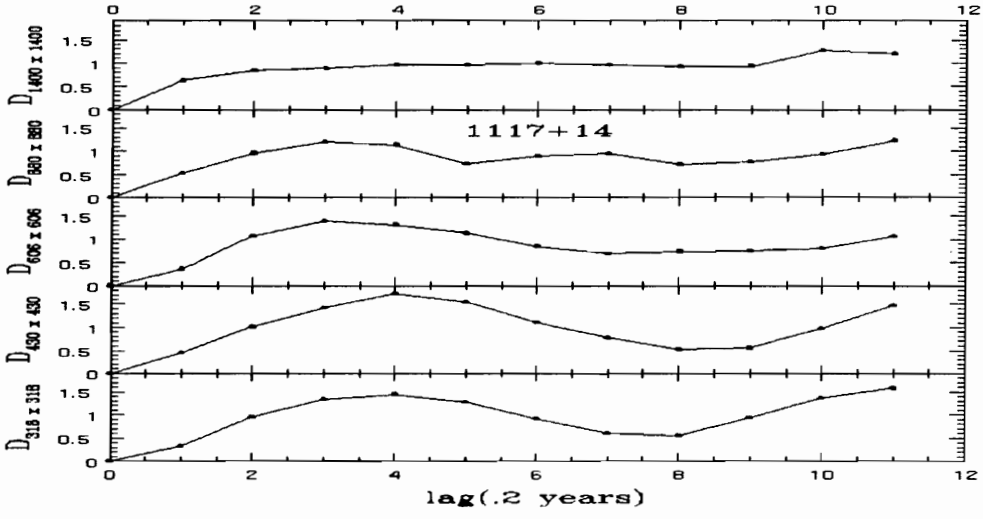


Fig. 34 Auto and cross structure-functions for the source 1117+14

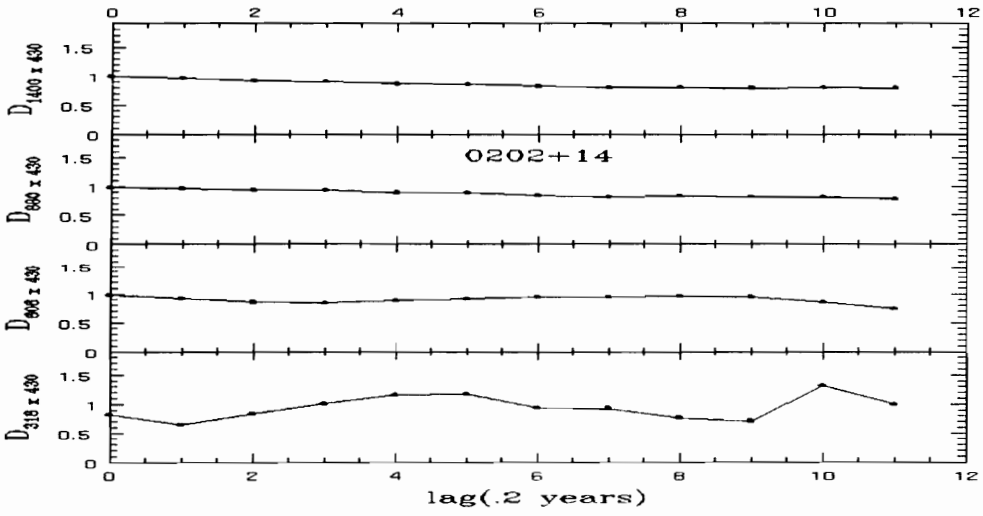
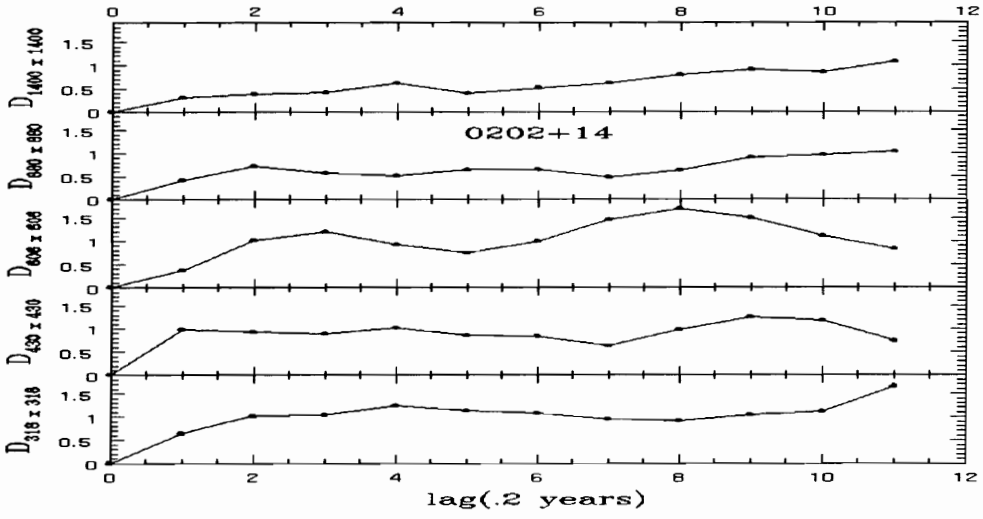


Fig. 35 Auto and cross structure-functions for the source 0202+14

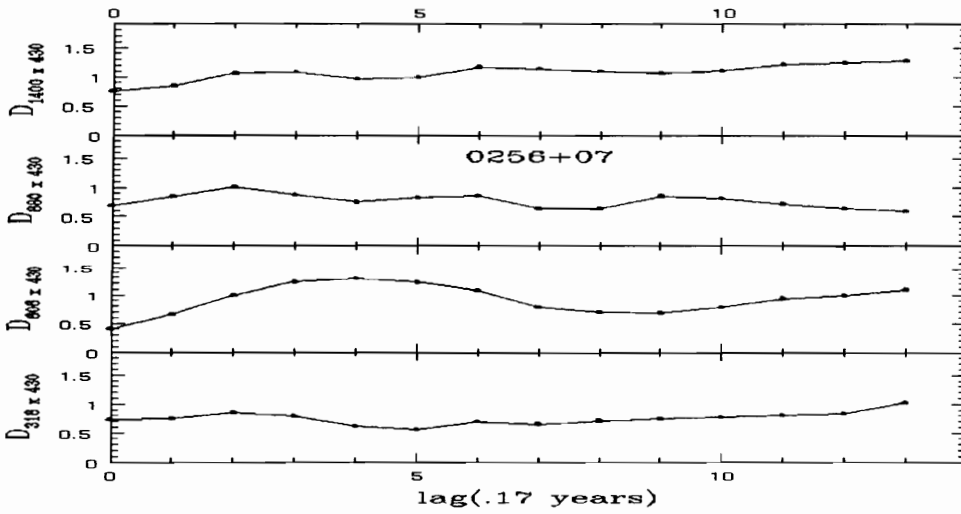
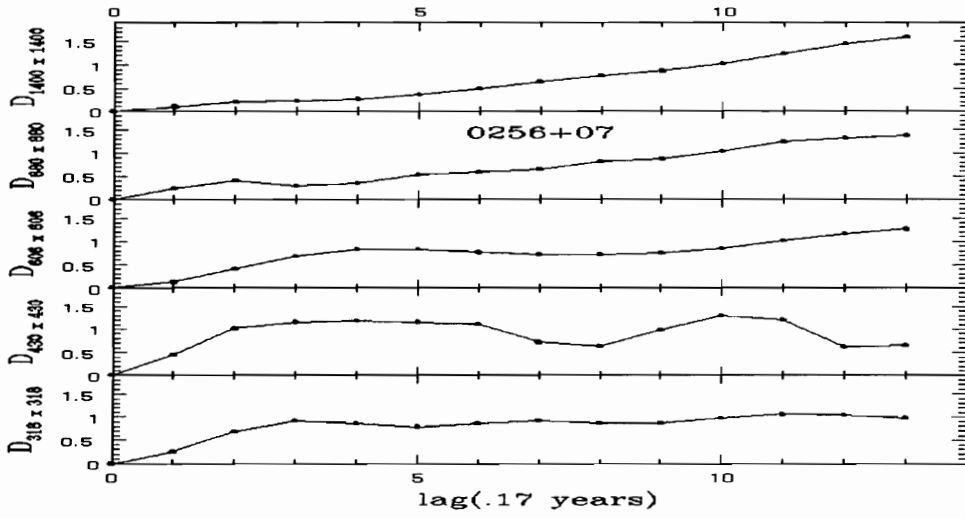


Fig. 36 Auto and cross structure-functions for the source 0256+07

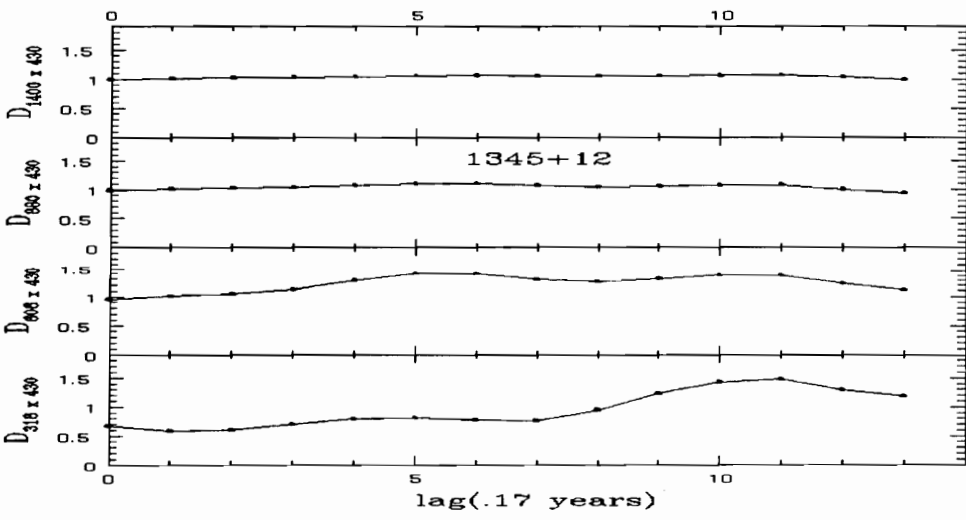
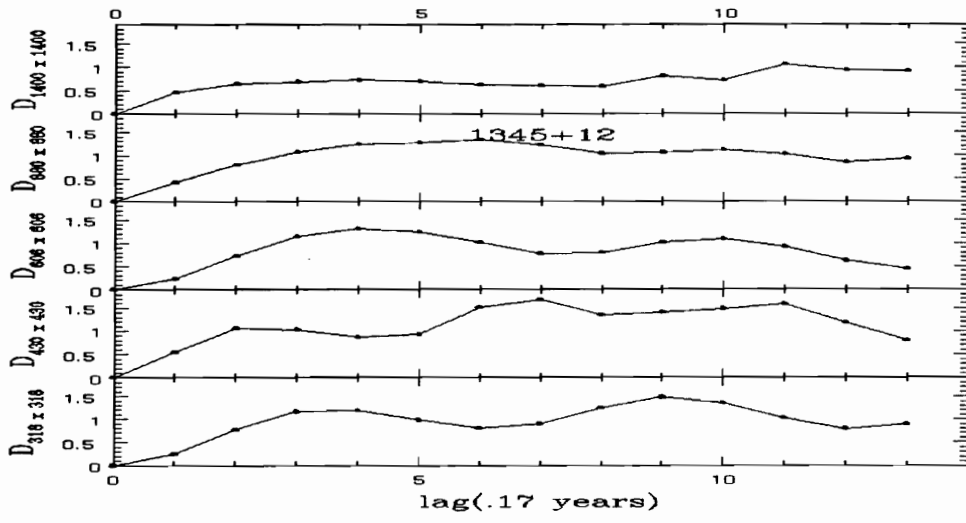


Fig. 37 Auto and cross structure-functions for the source 1345+12

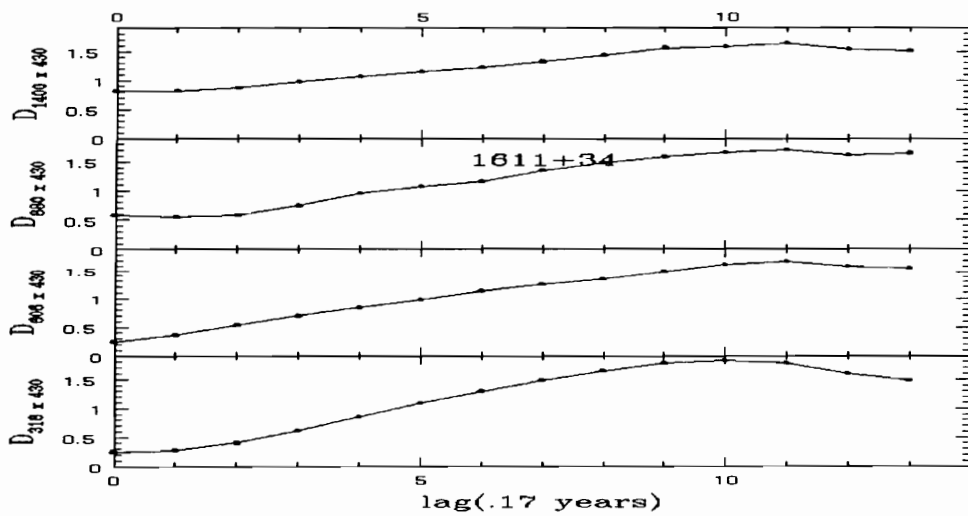
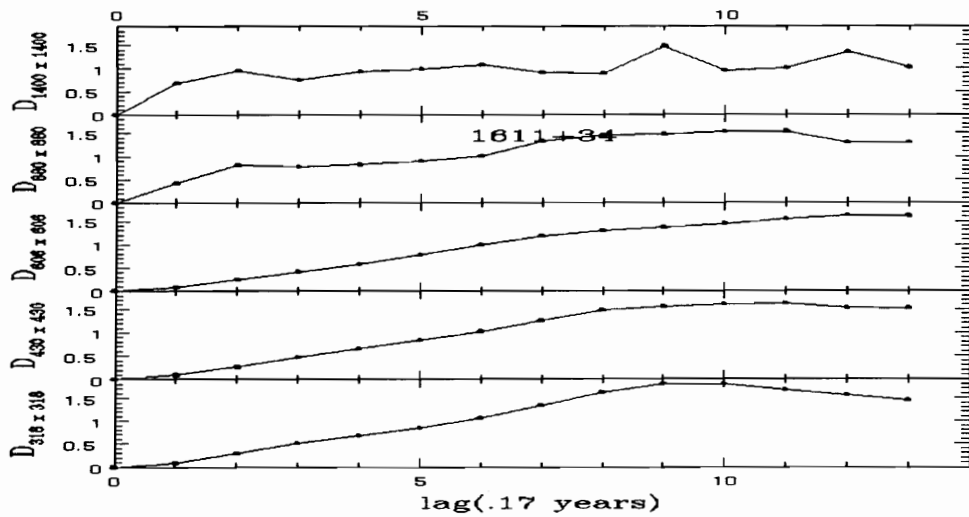


Fig. 38 Auto and cross structure-functions for the source 1611+34

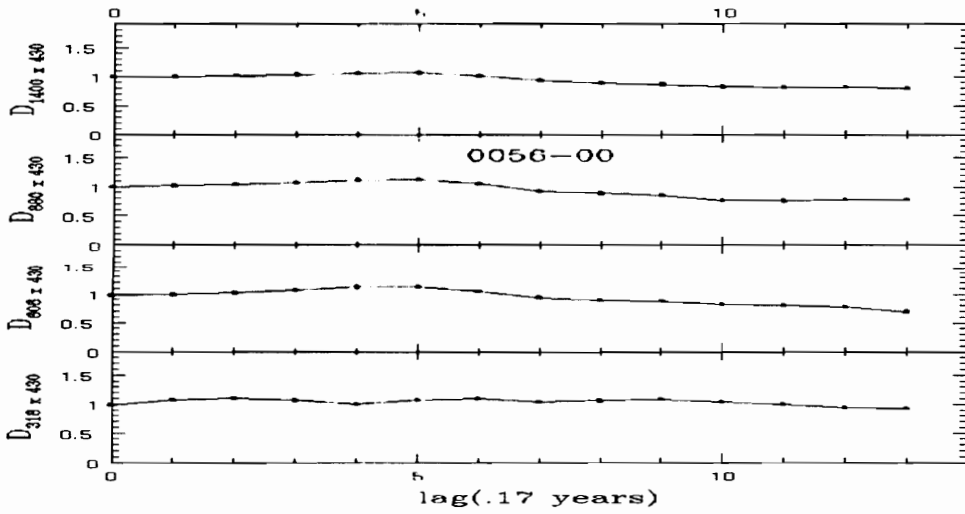
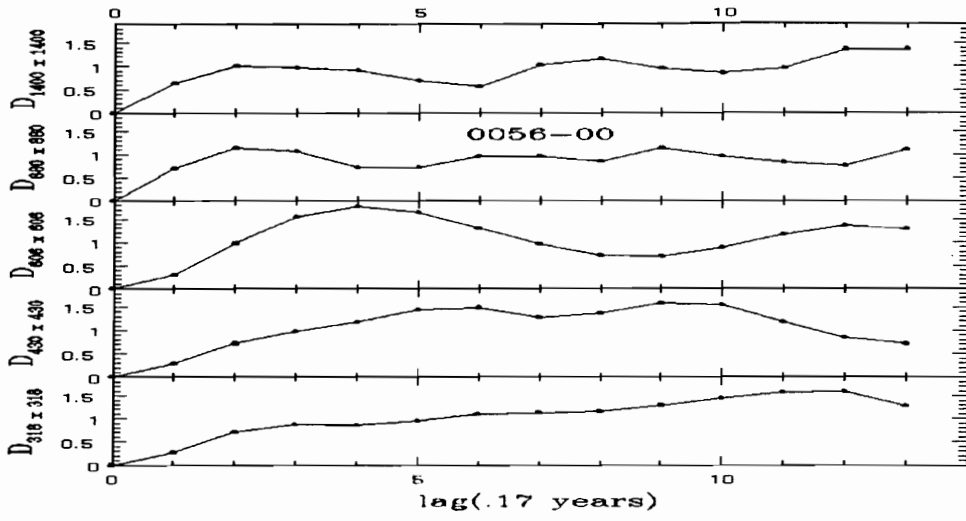


Fig. 39 Auto and cross structure-functions for the source 0056+00

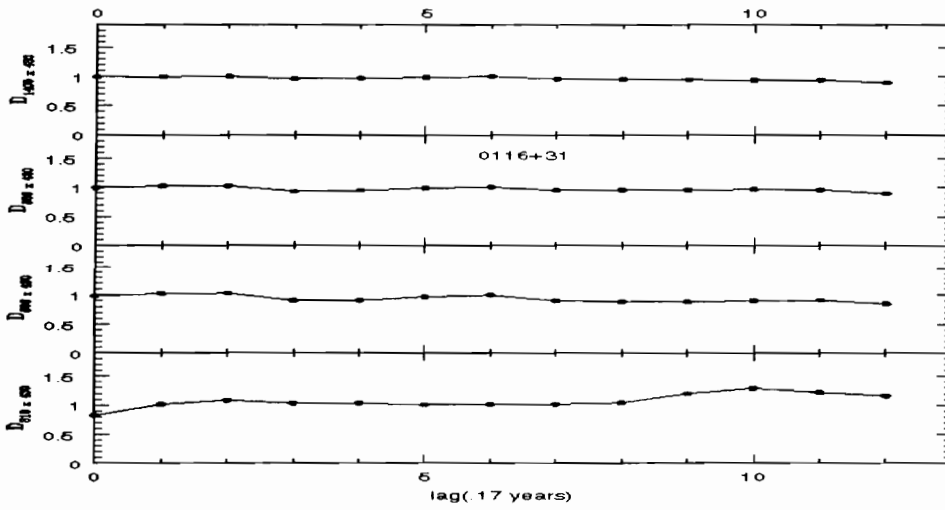
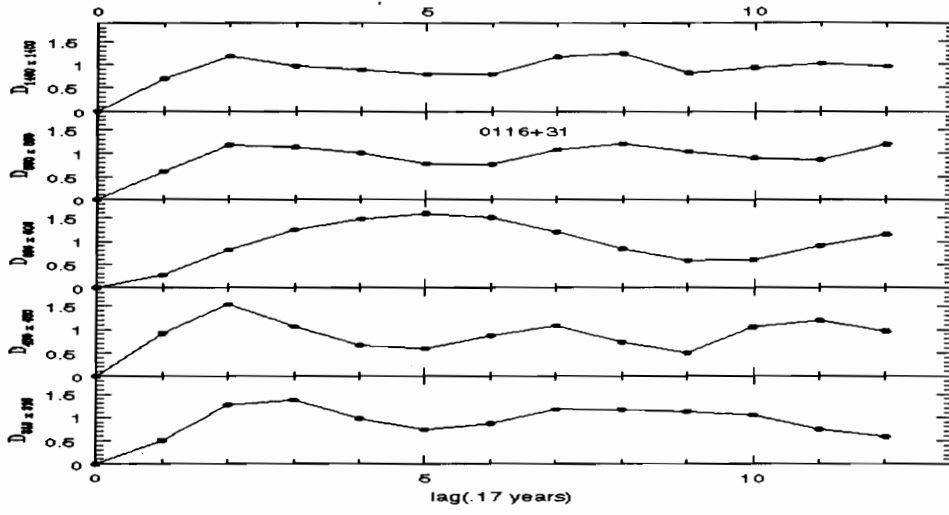


Fig. 40 Auto and cross structure-functions for the source 0116+31

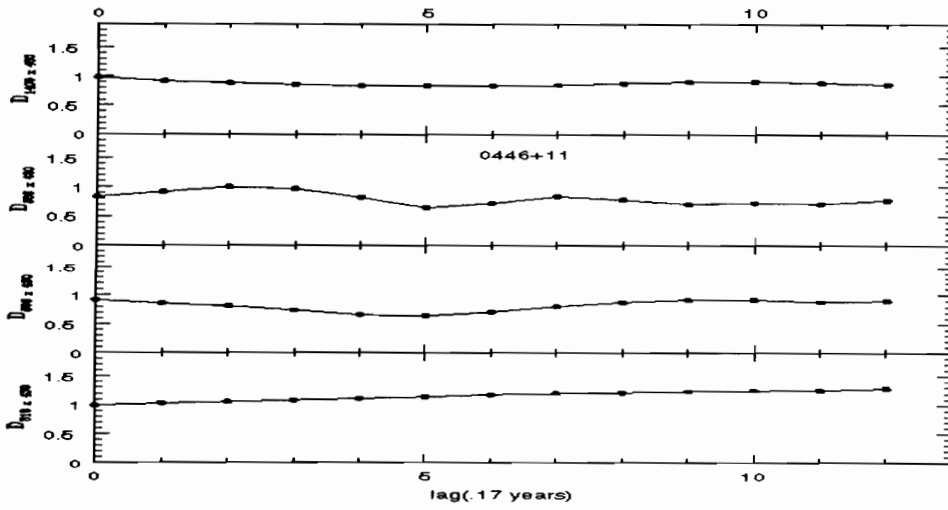
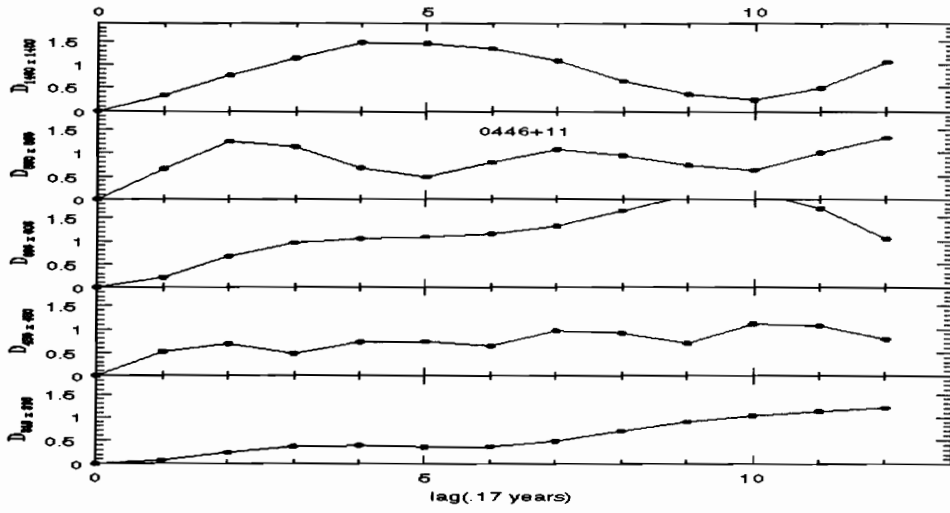


Fig. 41 Auto and cross structure-functions for the source 0446+11

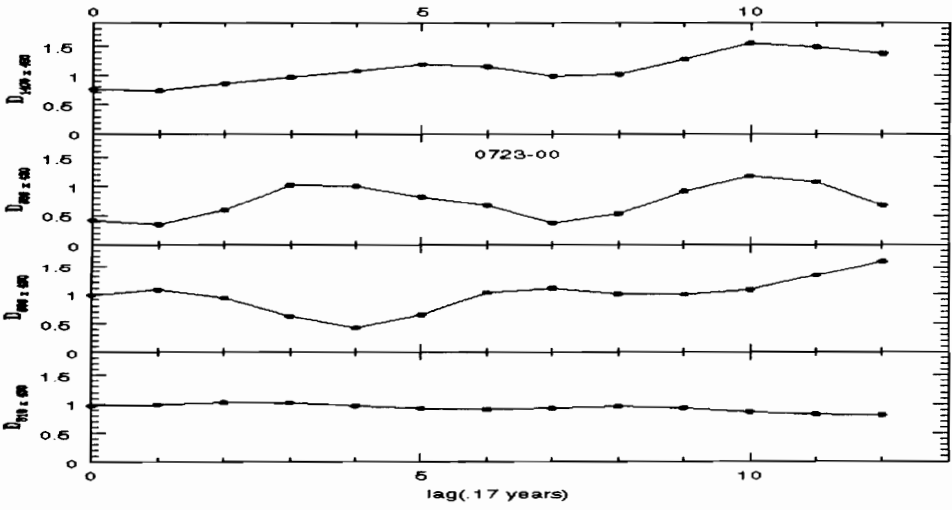
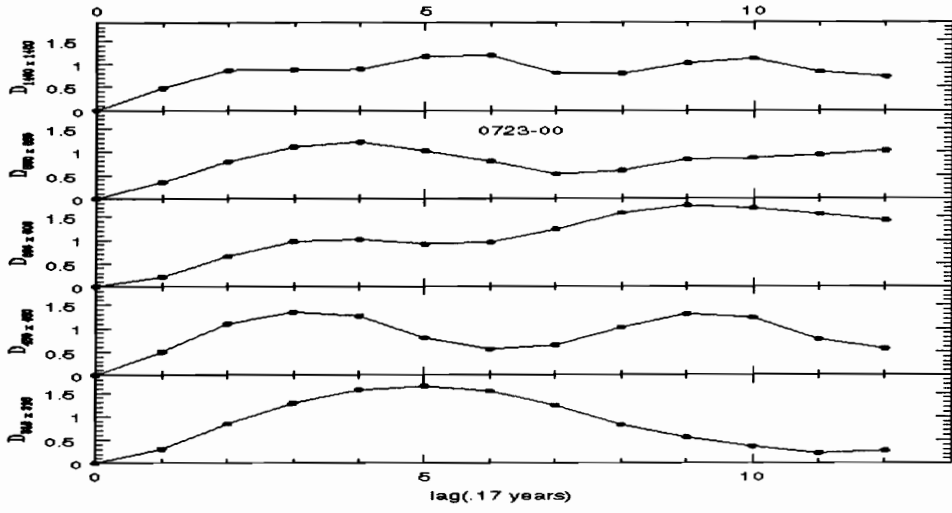


Fig. 42 Auto and cross structure-functions for the source 0723-00

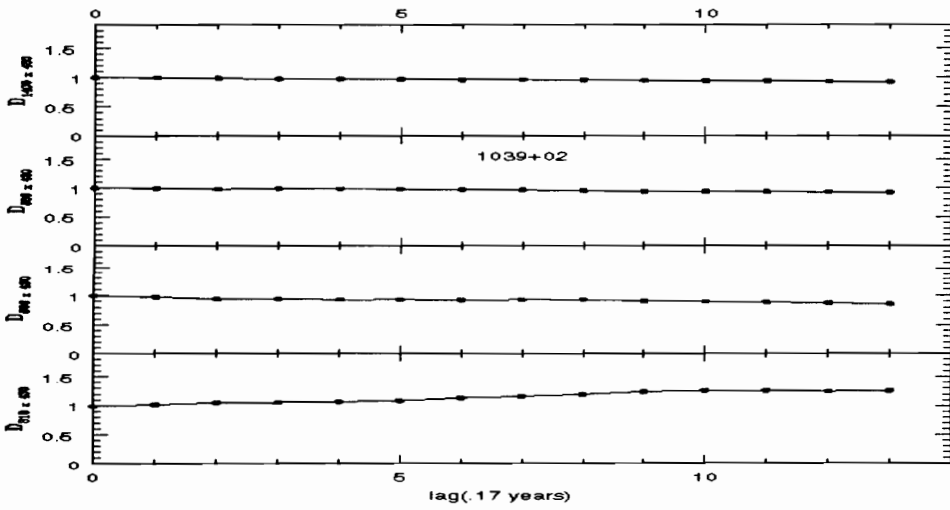
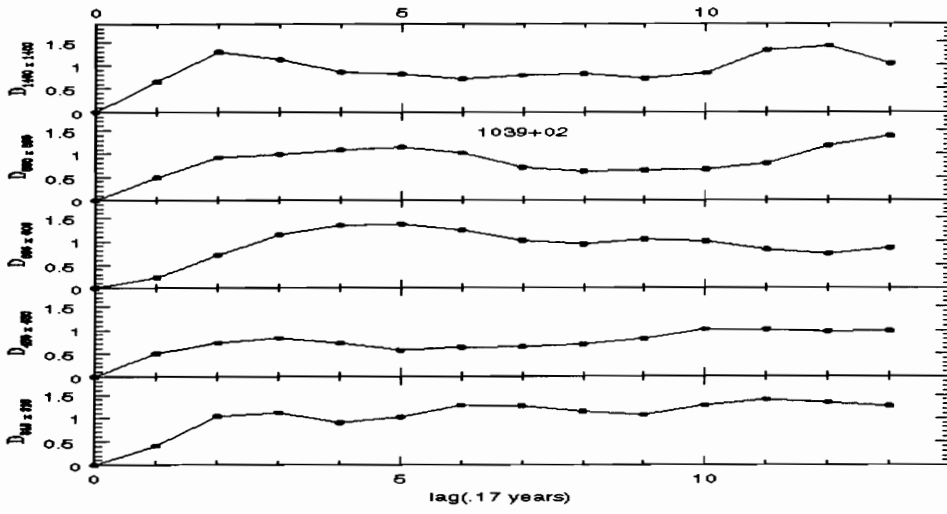


Fig. 43 Auto and cross structure-functions for the source 1039+02

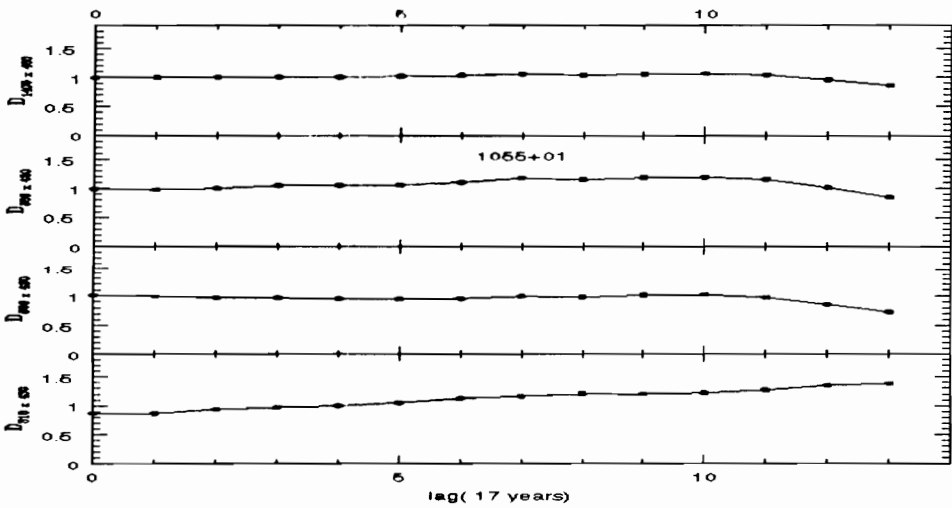
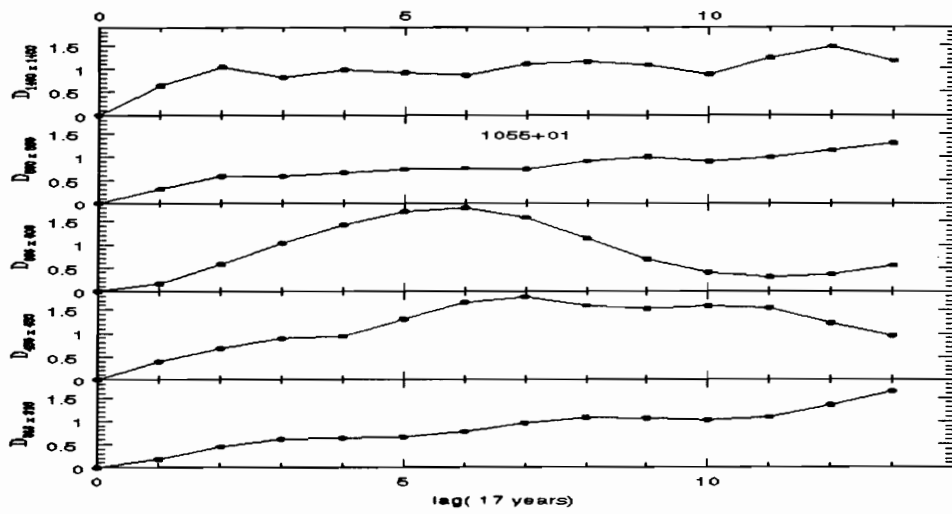


Fig. 44 Auto and cross structure-functions for the source 1055+01

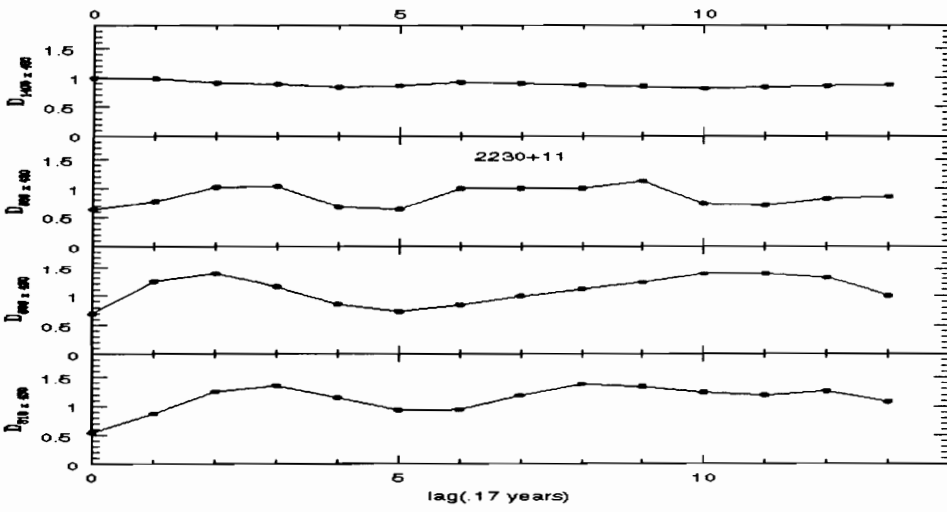
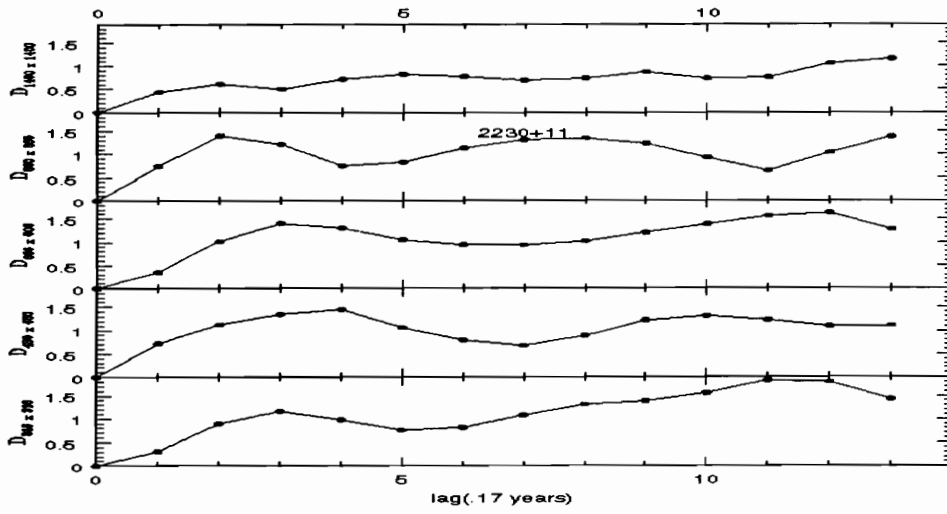


Fig. 45 Auto and cross structure-functions for the source 2230+11

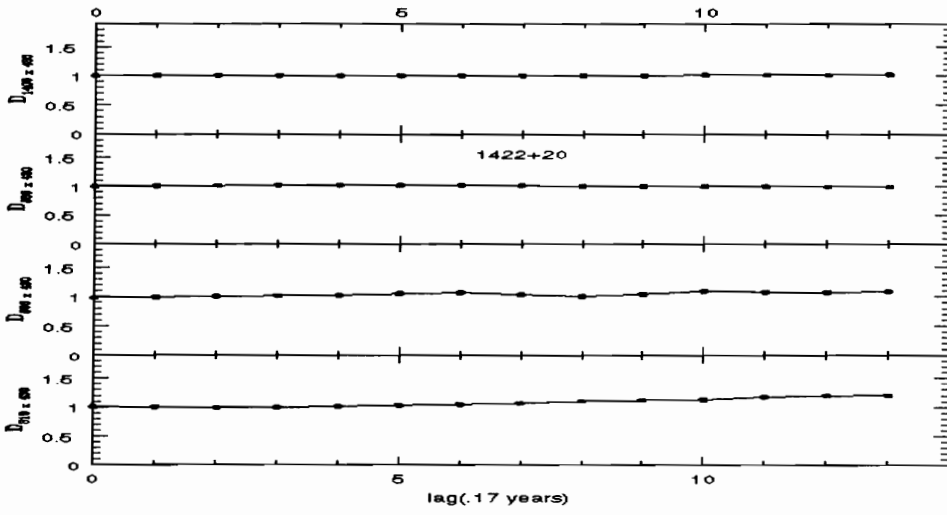


Fig. 46 Cross structure-functions for the source 1422+20

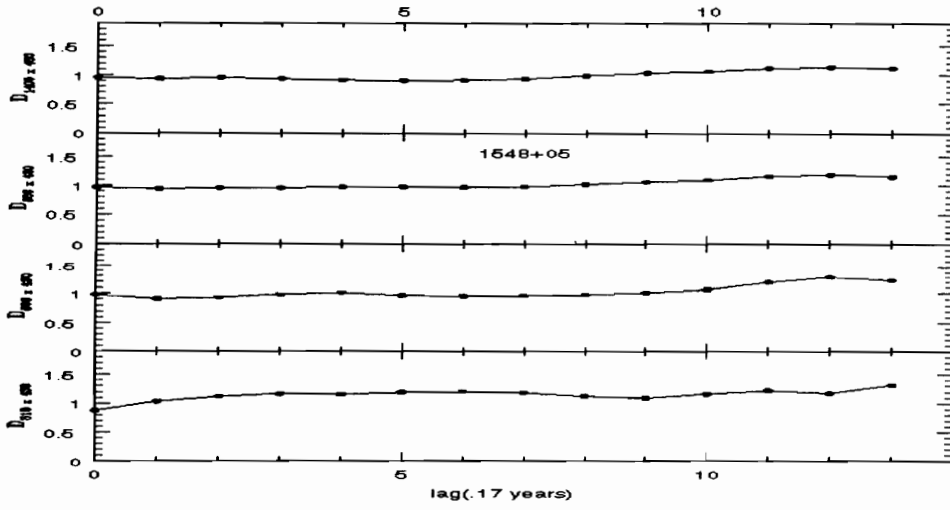
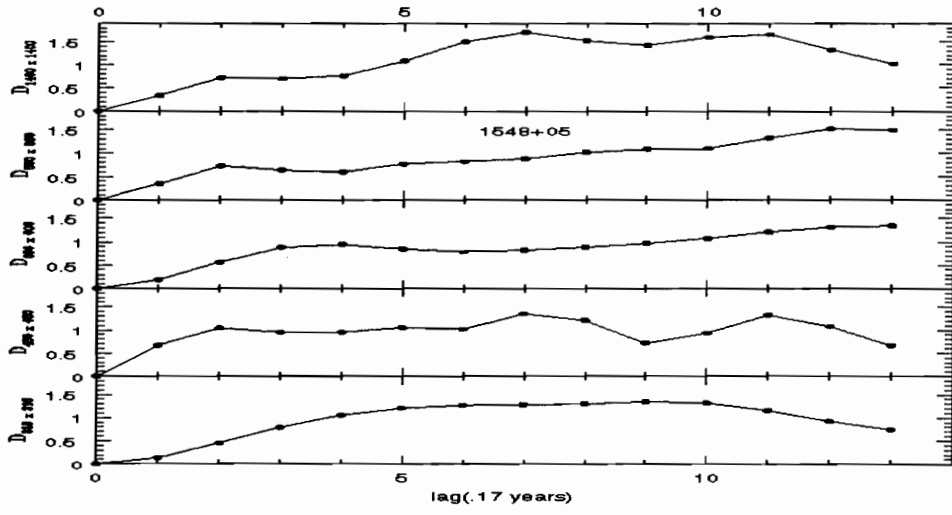


Fig. 47 Auto and cross structure-functions for the source 1548+05

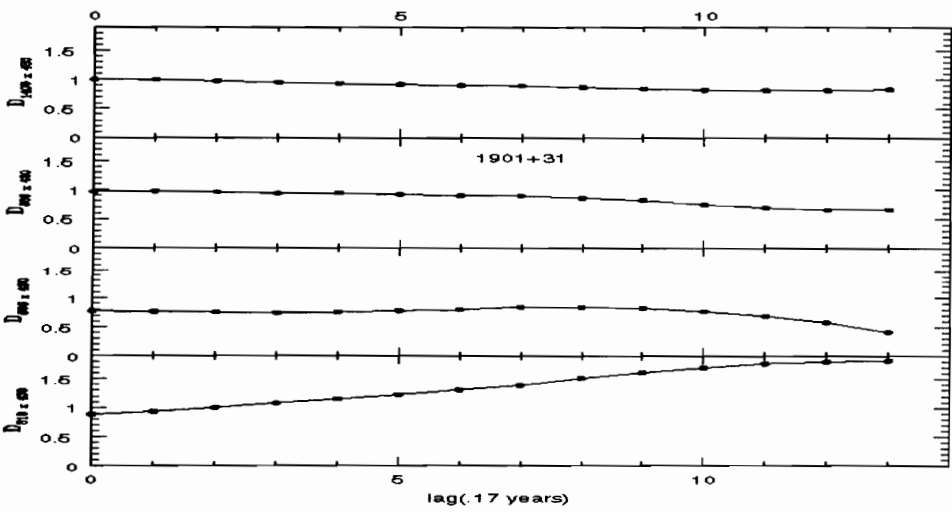
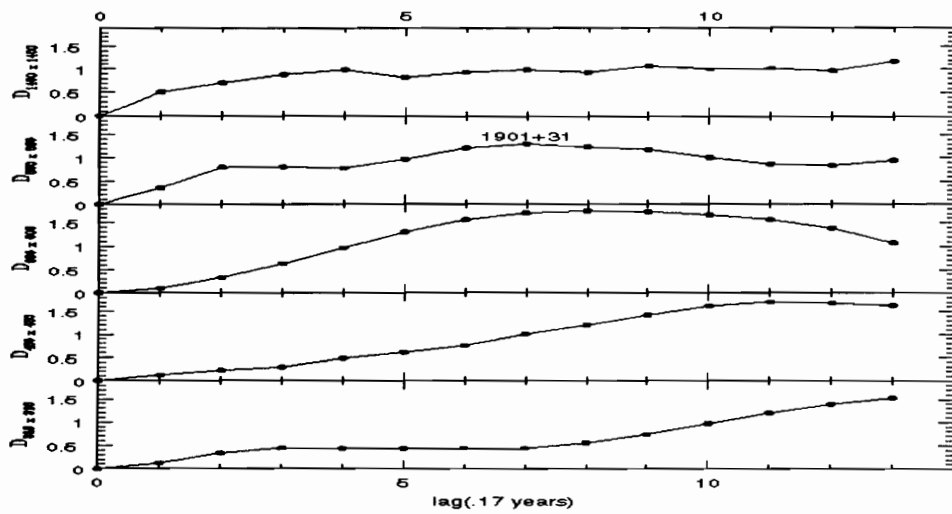


Fig. 48 Auto and cross structure-functions for the source 1901+31

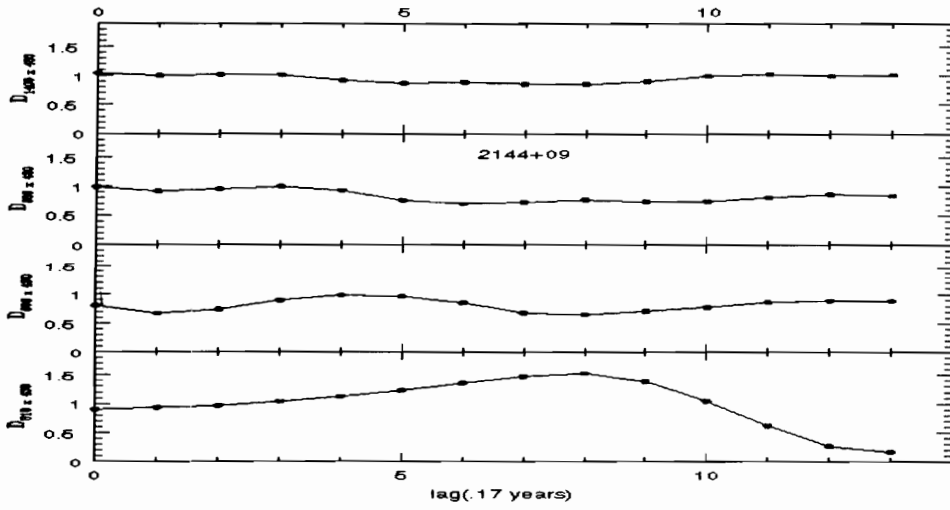
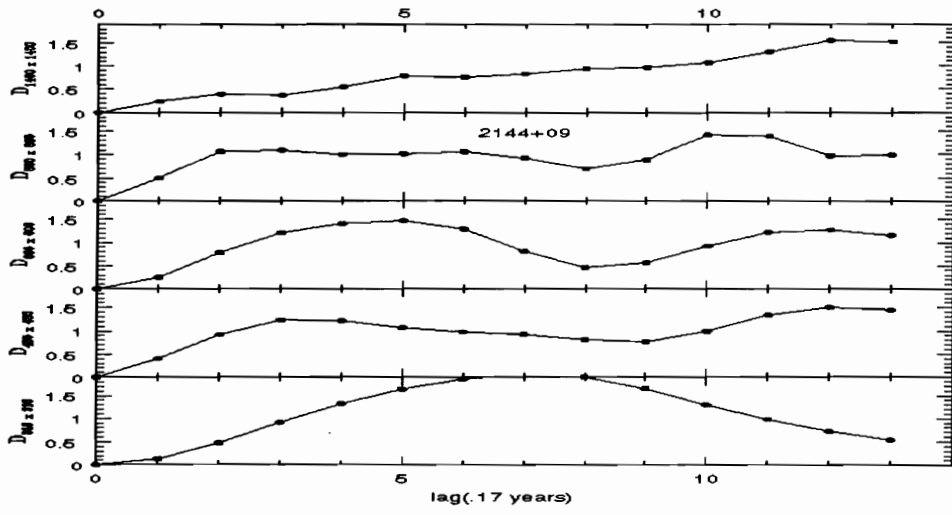


Fig. 49 Auto and cross structure-functions for the source 2144+09

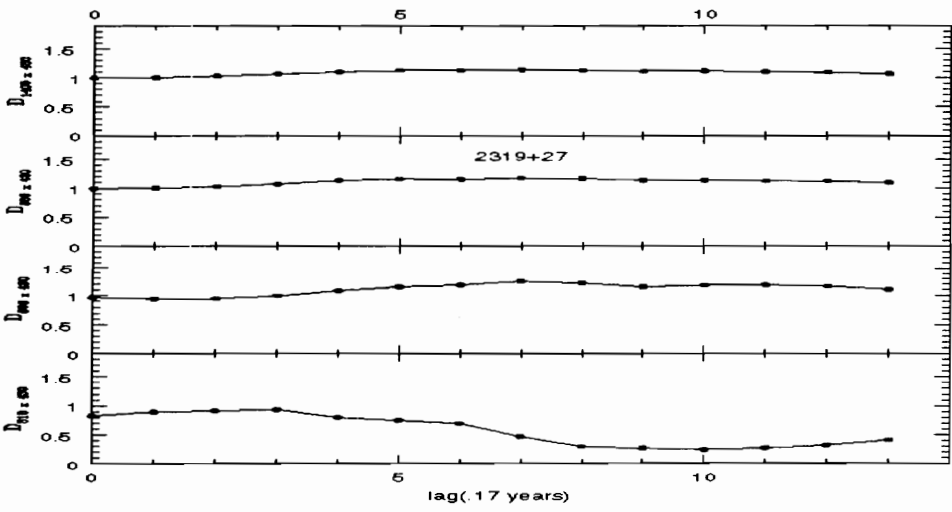
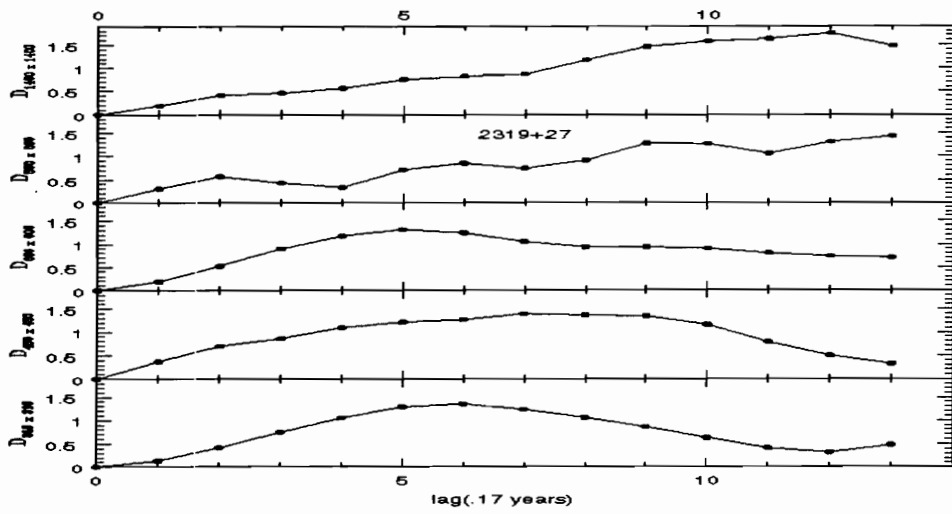


Fig. 50 Auto and cross structure-functions for the source 2319+27

We then estimate the characteristic timescales of the variability by taking one half of the time needed for the structure function to level off. (For the observations, see table 1; for the model, see tables 2 and 3.) We also estimate the minima of the cross-structure functions for the data (see table 4) and their lag, as well as for the model. (See table 3.) The minima for the model occur at zero lag, and the unit of lag for the model corresponds to one grid spacing.

Table 1. Root mean square variabilities and characteristic scales τ of observed sources.

<i>Source</i>	m_{318}	m_{430}	m_{606}	m_{880}	m_{1400}	τ_{318}	τ_{430}	τ_{606}	τ_{880}	τ_{1400}
0056 + 00	2.9	6.8	3.5	2.1	2.2	1.5	2.2	1.25	0.9	1.0
0116 + 31	2.9	2.9	3.0	1.9	2.0	0.9	0.5	1.4	0.8	0.8
0202 + 14	5.0	2.3	3.8	3.1	3.7	1.5	0.5	1.5	1.0	2.0
0256 + 07	14.	12.	14.	17.	17.	1.5	1.0	2.0	6.0	6.0
0446 + 11	4.5	5.3	7.3	8.4	12.	6.0	1.0	2.0	0.8	1.0
0723 - 00	3.3	7.1	2.9	4.8	5.1	1.6	0.9	1.5	1.1	1.0
1039 + 02	2.7	2.9	1.7	1.5	3.0	1.5	1.6	1.5	1.5	0.8
1055 + 01	4.5	2.7	2.6	3.3	3.7	6.0	2.5	1.7	3.0	1.0
1117 + 14	16.	11.	8.1	3.5	2.8	1.2	1.2	1.1	1.4	2.0
1345 + 12	3.4	2.6	3.4	1.5	1.3	2.0	2.7	1.5	1.6	1.9
1422 + 20	3.3	1.7	3.0	2.4	1.9	1.5	0.9	0.9	1.0	2.0
1548 + 05	7.0	3.3	3.5	4.5	4.4	2.5	2.5	2.0	3.1	2.9
1611 + 34	14.	19.	13.	6.6	4.4	4.0	4.5	5.5	4.0	3.0
1901 + 31	3.9	5.5	7.4	6.8	3.2	6.0	4.5	3.0	3.0	2.0
2144 + 09	6.0	8.4	5.2	7.1	9.8	2.0	2.0	1.5	1.5	6.0
2230 + 11	7.2	4.6	5.9	1.7	1.5	1.5	1.0	1.2	0.8	2.0
2319 + 27	5.5	4.6	5.8	5.3	4.7	2.0	1.6	1.5	6.0	6.0

a. where m are the % root mean square variabilities from Mitchell et al. (1994).

b. The timescales τ were estimated from the auto structure functions for each frequency, and $\tau \times .17$ =timescale in years.

Table 2. Root mean square variabilities from the model.

α	m ₃₁₈	m ₄₃₀	m ₆₀₆	m ₈₈₀	m ₁₄₀₀
2.1	69.0	24.0	12.0	5.9	2.8
3.1	72.0	27.0	13.0	6.5	2.8
3.6	72.0	35.0	16.0	7.0	3.4

a. where m are the % root mean square variabilities

The root mean square variabilities are defined as σ/\bar{x} , where:

$$\sigma = \left[\frac{\sum (x_i - \bar{x})^2}{N} \right]^{\frac{1}{2}} 100$$

This is the same way the percent variabilities are calculated in the data. The variability in the model is calculated over the whole intensity image, in order to get an indicative value for each α .

Table 3. Characteristic scales τ and minima of cross structure functions from the model

α	τ_{318}	τ_{430}	τ_{606}	τ_{880}	τ_{1400}	$D_{318 \times 430}$	$D_{606 \times 430}$	$D_{880 \times 430}$	$D_{1400 \times 430}$
3.6	7.5	9.0	9.0	10.0	10.0	0.45	0.35	0.65	0.85
3.1	7.5.0	8.0	9.0	9.0	10.0	0.45	0.25	0.6	0.85
2.1	7.5	7.5	7.5	7.5	8.5	0.48	0.25	0.55	0.8

Table 4. Minima of cross structure functions D , of all frequencies, cross 430 MHz, and their lag

<i>Source</i>	$D_{318 \times 430}$	<i>lag</i>	$D_{606 \times 430}$	<i>lag</i>	$D_{880 \times 430}$	<i>lag</i>	$D_{1400 \times 430}$	<i>lag</i>
0056 + 00	0.9	4	0.8	7	0.8	7	0.9	7
0116 + 31	0.8	0	0.9	3	0.9	3	0.9	3
0202 + 14	0.6	1	0.8	2	0.8	6	0.8	6
0256 + 07	0.6	5	0.4	0	0.7	0	0.8	0
0446 + 11	1.0	0	0.6	5	0.6	5	0.8	5
0723 - 00	0.8	5	0.4	5	0.4	1	0.7	1
1039 + 02	1.0	0	0.9	2	0.9	2	0.9	3
1055 + 01	1.0	0	0.9	2	1.0	0	1.0	0
1117 + 14	0.5	0	0.4	1	0.8	1	0.9	3
1345 + 12	0.6	1	0.9	0	1.0	0	1.0	0
1422 + 20	1.0	0	1.0	0	1.0	0	1.0	0
1548 + 05	0.9	0	0.9	1	0.9	1	0.9	4
1611 + 34	0.25	0	0.25	0	0.6	0	0.8	0
1901 + 31	0.9	0	0.8	0	0.9	6	0.9	6
2144 + 09	0.9	0	0.7	1	0.6	6	0.9	5
2230 + 11	0.5	0	0.7	0	0.6	0	0.8	4
2319 + 27	0.2	6.0	0.95	1	1.0	0	1.0	0

- a. Lag is in units of .17 years for all sources except 1117 + 14, where lag is in units of .2 years.
- b. The cross structure functions are normalized to one

We use the minimum of the cross-structure function to classify the sources, since the minimum gives the degree of correlation between the various frequencies and 430 MHz. This cross-frequency correlation, or lack thereof, is the most important feature of the observations. We separate the sources into 4 groups, A,B,C and bonafide intrinsic and show the behavior of the minima of the cross-structure functions, as well as that of the model ($\alpha = 3.6$) in figure 51.

We can see that sources in group A, 2144+09, 0256+07, 1117+14, 0723-00, 1611+34, 0446+11, 0333+32, in which the variability diminishes with frequency, exhibit the same behavior, that is they all approach 1 as we approach 1400 MHz, where there is no correlation at all.

Sources in group B, 2230+11, 0202+14, 1345+12, 0851+20, 2251+15, also have light curves consistent with variability due to refractive effects. These sources show little correlation between 606-430 or 880-430, but 318-430 shows a strong correlation. A possible explanation is that both intrinsic and extrinsic variability are present. Extrinsic dominates at 606 and especially 880 and 1400. Thus, these higher frequencies will not correlate well with 430.

Sources in group C, 0056-00, 0116+31, 1039+02, 1055+01, 1422+20, 1548+05, 1901+31, do not show any correlation between 430 MHz and the other frequencies and it is difficult to characterize them, in reference to refractive scintillation. The sources 0735+17 and 0235+16 on the other hand, show strong correlation between light curves at 430 MHz and the higher frequencies, especially source 0735+17, and they are grouped under the intrinsic group. O'Dell et al. (1988) showed conclusively that 0235+16 is an intrinsic variable.

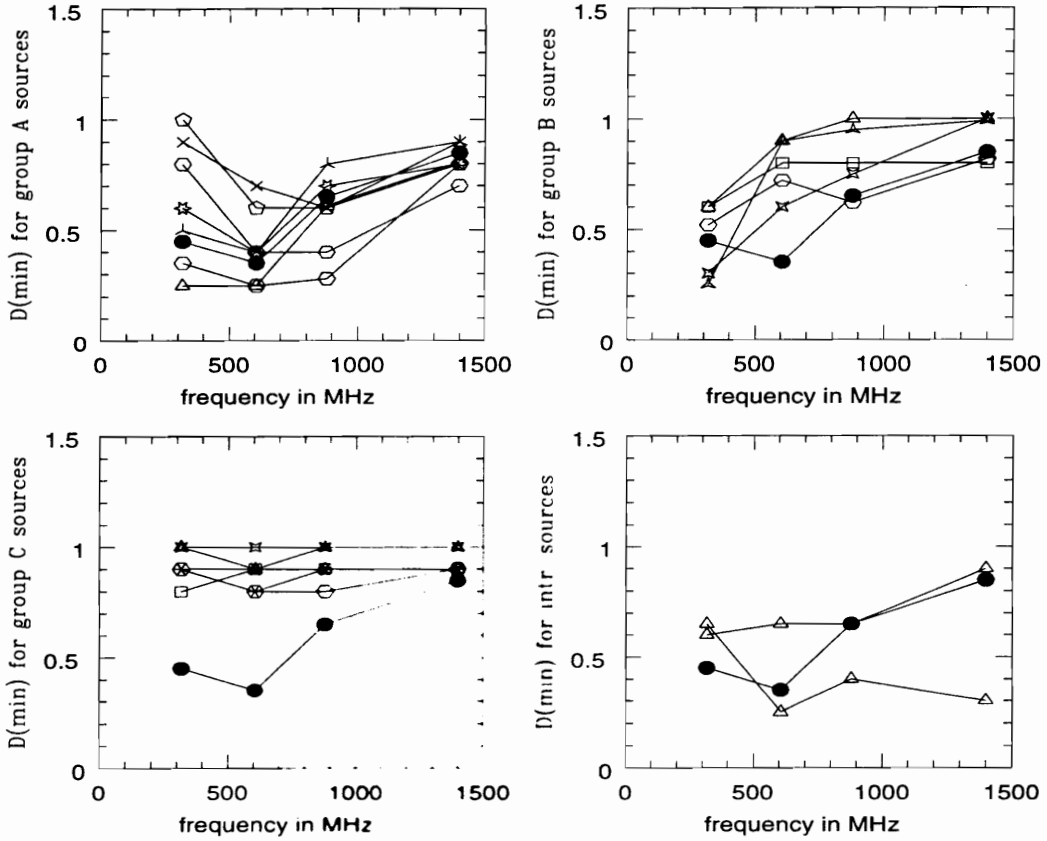


Fig. 51 Cross structure-function minima for the model, denoted by \bullet , and sources as listed in groups A.B.C. and intrinsic.

We also look at the ratio of the timescale of the variability of the sources to the measured angular size, by using the timescales for the frequency 318 MHz, as calculated from the auto-structure functions of the data, and the angular size as measured at 326 MHz by Altschuler et al. (1995). For this discussion, we use a simple heuristic model described by Mantovani et al. (1990). We can find the effective screen distance by using $z_c = \frac{\tau v}{\theta_m}$ where v is the velocity of the cloud and θ_m the measured angular size of the source. We take v to be 50 km s^{-1} and τ from table 1. Figure 52 shows the correlation of the scattering angular size with galactic latitude for sources

in groups A and B, as expected, with the exception of the source 1611 + 34. It should be noted that the source 1611 + 34 is probably anomalous as indicated by the fact that the component size used contains only 37% of the total flux density (Altschuler et al. 1995). We see the same trend with galactic latitude as in Mantovani et al. (1990).

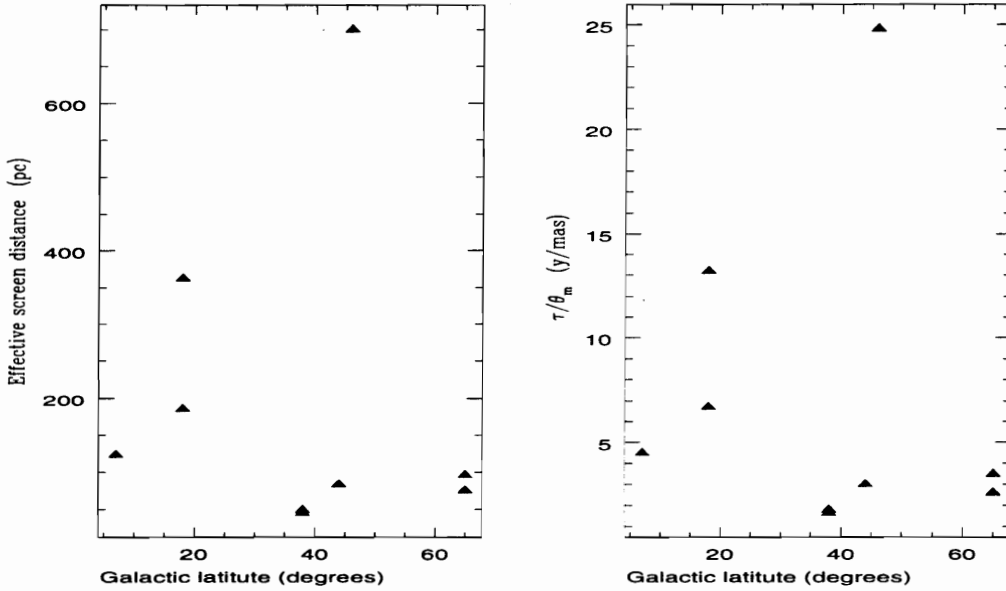


Fig. 52 Effective screen distance and source angular size versus galactic latitude.

This work has conclusively demonstrated that the refractive scintillation hypothesis accounts for the behavior of a significant subgroup of low frequency variables, and that this type of variability is correlated over only an octave of frequencies. Such variations are incompatible with intrinsic variability of synchrotron emitting components. The best agreement with our results is with the source 1117 + 14, which also has the same frequency dependent behavior as the modeled light curves. It is also possible that for some sources, both intrinsic effects and refractive effects are taking place, as evidenced by the group B sources. In these cases, the light curves have variability at high frequency, propagating to the lowest frequencies with diminished

amplitude, and the refractive variability dominates at the lowest frequencies.

References

1. Aller, H. D. and Aller, M. F. 1982, in *Low Frequency Variability of Extragalactic Radio Sources*, eds: W. D. Cotton and S. R. Spangler (NRAO: Green Bank) p. 105.
2. Altschuler, D.R., Gurvits, L.I., Alef, W., Dennison, B., Graham, D., Trotter, A.S. Carson, J.E., *Astron.Astrophys.Suppl.Ser.* 114, 197-214 ,1995.
3. Armstrong, J.W., Cordes, J.M. and Rickett, B.J., *Nature* , 291, No 5816, 561-564, 1981.
4. Condon, J.J., Ledden, J.E., O'Dell, S.L., Dennison, B., *Astron.J.* 84, no 1, 1-11.
5. Cordes, J.M., Weisberg, J.M. and Boriakoff, V. *Astrophys.J.* 288, 221-247, 1985.
6. Dennison, B., Condon, J.J. *Astrophys.J.* 246, 91-99 1981.
7. Dennison B., Broderick, J.J., O'Dell, S.L., Mitchell, K.J., Altschuler, D.L., H.E. Payne, H.E. and Condon, J.J., *Astrophys.J.* 281,L55-L58,1984.
8. Fanti,C., Fanti,R., Ficarra, A., Mantovani, F., Padrielli, L., and Weiler, K.,W. 1981, *Astrophys.J.Sup.*, 45,61.
9. Gwinn, C.R., Moran, J.M., Reid, M.J. and Schneps, M.H. *Astrophys.J.*, 330, 817-827, 1988.
10. Hunstead, R.W. 1972, *Astrophys.Let.* 12,193.
11. Jackson J.D *Classical Electrodynamics*, Wiley 2nd edition.
12. Mantovani, F., Fanti, R., Gregorini, L., Padrielli, L., and Spangler, S. 1990, *Astron. Astrophys.*, 233,535-539.
13. Marscher, A. P. 1982, in *Low Frequency Variability of Extragalactic Radio Sources*, eds: W. D. Cotton and S. R. Spangler (NRAO: Green Bank) p. 83.
14. Marscher, A.P., *Astrophys.J.* 228, no 1, 27-33.
15. Mitchell, D.L, Dennison, B., Condon, J. J., Altschuler, D. R., Payne, H.E., O'Dell, S. L. and Broderick, J.J. 1994, *Astrophys.J.Sup.* 93, 441.

16. O'Dell, S.L., Dennison, B., Broderick, J.J., Altschuler, D.R., Condon, J.J, Payne, H.E., Mitchell, K.J., Aller, H.D., Aller, M.F., Hodge, P.E., *Astrophys.J.*, 326, 668-679, 1988.
17. O'Dell, S. L. 1982, in *Low Frequency Variability of Extragalactic Radio Sources*, eds: W. D. Cotton and S. R. Spangler (NRAO: Green Bank) p. 89.
18. Pacholczyk, A.G., *Radio Astrophysics*, W.H. Freeman and Company, 1970.
19. Pack, J. *Num. simulation of optical wave propagation through random media* thesis, VPI& SU 1988.
20. Parks G.K. *Physics of Space Plasmas*, Addison-Wesley.
21. Papoulis A. *The Fourier Integral and its Applications* N.Y McGraw-Hill,1962.
22. Press, W.H., B.P Flannery, S.A Teukolsky, and W.T. Vetterling, *Numerical Recipes* Cambridge University Press, New York, 1988.
23. Renka J.R. *Algorithm 624* ACM Transactions on Math. Software, Vol. 10,No.4, December 1984, pp 440-442.
24. Rickett, B.J., *Astrophys.J.* 307,564-574,1986.
25. Rickett, B.J., *Annu. Rev. Astron. Astrophys* 28,561-605,1990.
26. Rickett,B.J., Coles, W.A., and Bourgois.G. 1984, aap ,134,390.
27. Shapirovskaya, N. Ya 1978, Soviet Astron. 22,544.
28. Shklovsky, I. S. 1965, Soviet Astron. 9,22.
29. Sholomitskii, G. B. 1965,Soviet Astron. 9,516.
30. Steinbring, D. and Condon, J.J. *Astrophys.J.*, 352, 207-221, 1990.
31. Verschuur, G.L. and Kellerman, K.I., *Galactic and Extra-Galactic Radio Astronomy*, Springer-Verlag, New York, 1974.
32. Weng Cho Chew *Waves and Fields in Inhomogeneous Media*, Van Nostrand Reinhold, N.Y.
33. Williamson R.S, Jones D.L, *Radio Science* vol.25, Num.5, pp 743-755, 1990.

Appendix A

A.1 Plasmas. For high frequencies, the electric permittivity is given by:

$$\epsilon(\omega) \approx 1 - \frac{\omega_p^2}{\omega^2}$$

where $\omega_p^2 = \frac{4\pi Nze^2}{m}$ is the plasma frequency and Nz is the number of electrons per unit volume. For frequencies lower than ω_p the wave number is purely imaginary. Such waves incident on plasmas are reflected. In this work, we are interested in the refraction of the waves, thus we are looking at frequencies $\omega > \omega_p$, where ω is the observation frequency, which for our data is in the range $2\pi(300 - 1400)$ MHz. Since $\omega = 2\pi f = 2\pi 1400 \text{ MHz} = 8.79 \times 10^6 \text{ sec}^{-1}$ for an ISM density, $n_e = 10^3 \text{ cm}^{-3}$ we have $\omega_p = \left(\frac{4\pi \times 10^3 e^2}{m_e}\right)^{\frac{1}{2}} = 1.87 \times 10^{-2} \text{ sec}^{-1}$, for $e = 4.8 \times 10^{-10} \text{ esu}$ and $m_e = 9.11 \times 10^{-28} \text{ g}$, which justifies the above assumption.

Moreover we are considering a collisionless(lossless) plasma (appendix A 1.1). To see this, we can consider the coolest and most dense plasma to maximize collisions, a plasma with $T = 10^3 \text{ }^\circ\text{K}$ and $n_e = 10^6 \text{ cm}^{-3}$. The collision frequency is given by:

$$\omega_c \approx \frac{n_e e^4}{m^{\frac{1}{2}} [3KT]^{\frac{3}{2}}}$$

With the above temperature and density we get $\omega_c = 10 \text{ sec}^{-1}$.

Since the frequency and the magnitude of the wave vector are related by $k = \frac{\omega}{v} = \sqrt{\mu\epsilon} \frac{\omega}{c}$ we have with $\mu = 1$:

$$ck = \left(1 - \frac{\omega_p^2}{\omega^2}\right)^{\frac{1}{2}} \omega$$

and consequently the dispersion relation in plasmas:

$$\omega^2 = k^2 c^2 + \omega_p^2 = (kc)^2 + \frac{4\pi n e^2}{m}$$

Solving for $\lambda = 2\pi/k$ and expanding the square root we get:

$$\lambda = \frac{2\pi c}{\sqrt{\omega^2 - \omega_p^2}} = \frac{2\pi c}{\omega} \left(1 + \frac{1}{2} \frac{\omega_p^2}{\omega^2}\right)$$

Where λ is the wavelength. The phase is then given by:

$$\Phi = \frac{2\pi z}{\lambda} = \frac{\omega}{c} z \left(1 - \frac{1}{2} \frac{\omega_p^2}{\omega^2}\right)$$

The first term is not dependent on ω_p , so we ignore it since is constant. The change in phase will be given by:

$$d\Phi = -\frac{1}{2c} \frac{4\pi n e^2}{m\omega} dz$$

$$\Phi = -\frac{\pi e^2}{mc\omega} \int_0^z n(\vec{r}) dz$$

$$\Phi = -\frac{2\pi r_e c}{\omega} \int_0^z n(\vec{r}) dz$$

As mentioned above, we are considering collisionless plasmas. A formal way to see this is by considering the number of particles in a Debye sphere:

$$N_D = \frac{4\pi n_o \lambda_D^3}{3}$$

The plasma parameter (G.Parks) is defined as $g = \frac{1}{N_D}$, with the following in mind:

When $g \ll 1$ (plasma approximation), we have a number of particles large enough to apply concepts of statistics, and at the same time it defines the condition that the binary collisions are rare.

We know that the electrostatic potential at a distance r from a charged particle is $\Psi = \frac{q}{4\pi\epsilon_o r}$. But this relation is modified in the presence of a plasma. The plasma

exhibits a collective behavior (Parks), and due to that the electrostatic field is confined to a shorter range $\Psi = \frac{q}{4\pi\epsilon r} e^{-\frac{r}{\lambda_D}}$, where:

$$\lambda_D^2 = \frac{kT_e \epsilon_o}{n_o q_e^2}$$

the Debye length, and n_o the plasma density.

From the above relations we get:

$$g \approx \frac{n_o^{\frac{1}{2}}}{T^{\frac{3}{2}}}$$

In the limit of $g \rightarrow 0$, the plasma is collisionless.

A.2 Evaluation of $\langle \phi^2 \rangle$

The variance of the phase in terms of the power spectrum is

$$\langle \phi^2 \rangle = \frac{1}{2\pi} \int \int d^2\bar{q} P_\Phi(\bar{q})$$

$$\begin{aligned} \langle \phi^2 \rangle &= \frac{1}{2\pi} \int P_\Phi(\bar{q}) 2\pi q dq \\ &= C_\Phi^2 \int_{q_1}^{q_2} q^{1-\alpha} dq \\ &= C_\Phi^2 \frac{(q_2^{2-\alpha} - q_1^{2-\alpha})}{2-\alpha} \\ &\approx C_\Phi^2 \frac{q_1^{2-\alpha}}{\alpha-2} \end{aligned}$$

for $q_2 \gg q_1$ and $\alpha > 2$.

A.3 Evaluation of $\langle (\nabla^2 \phi)^2 \rangle$ The first term is

$$\langle (\nabla^2 \phi)^2 \rangle = \lim_{L \rightarrow \infty} \frac{1}{L^2} \frac{1}{(2\pi)^2} \int_{-\frac{L}{2}}^{\frac{L}{2}} \int_{-\frac{L}{2}}^{\frac{L}{2}} dx dy \left[- \int \int q_x^2 \Phi(q_x, q_y) e^{i\bar{q} \cdot \bar{r}} dq_x dq_y \right]^2$$

$$\begin{aligned}
&= \lim_{L \rightarrow \infty} \frac{1}{(2\pi)^2} \frac{1}{L^2} \int \int dx dy \left[\int \int q_x^2 \Phi(q_x, q_y) e^{i\bar{q} \cdot \bar{r}} dq_x dq_y \int \int q_x'^2 \Phi(q_x', q_y') e^{i\bar{q}' \cdot \bar{r}} dq_x' dq_y' \right] \\
&= \frac{1}{L^2} \int \int q_x'^4 |\Phi(q_x', q_y')|^2 dq_x' dq_y' \\
&= \frac{1}{(2\pi)} \int \int q_x'^4 P(q') dq_x' dq_y'
\end{aligned}$$

Similarly we get the other terms. Then using $q^4 = (q^2)^2 = (q_x^2 + q_y^2)^2 = q_x^4 + q_y^4 + 2q_x^2 q_y^2$ and the power spectrum as the common factor, we have in polar coordinates

$$\begin{aligned}
\langle (\nabla^2 \phi)^2 \rangle &= \frac{1}{(2\pi)} \int \int q'^4 P(q') dq_x' dq_y' \\
&= \frac{1}{(2\pi)} \int q'^4 C_{\Phi}^2 q'^{-\alpha} (2\pi) q' dq' \\
&= \int_{q_1}^{q_2} q'^{(5-\alpha)} C_{\Phi}^2 dq' \\
&= C_{\Phi}^2 \left(\frac{q_2^{6-\alpha}}{6-\alpha} - \frac{q_1^{6-\alpha}}{6-\alpha} \right) \\
&\approx C_{\Phi}^2 \frac{q_2^{6-\alpha}}{6-\alpha}
\end{aligned}$$

for $\alpha \neq 6$ and $q_2 \gg q_1$.

A.4 Evaluation of $\langle (\nabla \phi) \cdot (\nabla \phi) \rangle$

$$\begin{aligned}
\langle (\bar{\nabla} \phi) \cdot (\bar{\nabla} \phi) \rangle &= \lim_{L \rightarrow \infty} \frac{1}{L^2} \int \int dx dy \frac{1}{4\pi^2} \left[- \int \int d^2 \bar{q} q_x e^{i\bar{q} \cdot \bar{\rho}} \phi(\bar{q}) \int \int d^2 \bar{q}' q_x' e^{i\bar{q}' \cdot \bar{\rho}} \phi(\bar{q}') \right. \\
&\quad \left. - \int \int d^2 \bar{q} q_y e^{i\bar{q} \cdot \bar{\rho}} \phi(\bar{q}) \int \int d^2 \bar{q}' q_y' e^{i\bar{q}' \cdot \bar{\rho}} \phi(\bar{q}') \right] \\
&= \frac{1}{L^2} \int \int d^2 \bar{q} (q_x^2 + q_y^2) \phi^*(\bar{q}) \phi(\bar{q}) \\
&= \frac{1}{2\pi} \int \int d^2 \bar{q} (q_x^2 + q_y^2) P_{\phi}(\bar{q})
\end{aligned}$$

$$\langle (\bar{\nabla} \phi) \cdot (\bar{\nabla} \phi) \rangle = \int_{q_1}^{q_2} q'^{(3-\alpha)} C_{\Phi}^2 dq'$$

Appendix B

B.1 Documentation for the phaethon code:

A fortran 77 code, that propagates plane radio waves through a turbulent interstellar medium, and accounts for refractive effects at the limit of geometrical optics as described in this thesis.

main:phaethon.f, and subroutines therein

subprograms:

c

c :triangl.f

c :cell~interpol.f

c :simple~convolution.f

and subroutines therein.

include files:

c phaeth1.inc

initialization(parameter) files:

c phaethon.ini

command files:

c comfile

c comfile2

c comfile3

Shell:

change mode(to exe) file:

c cx

The programs and macro files below are needed to plot the phase screen, intensity output from cell-interpol.f and final image and trajectories from simple-convolution.f

The programs and macro files with pr, at the end of the name, are producing postscript files.

c

plotting programs(in fortran) using Super Mongo
graphics subroutines :

c plotxy.f

c plotxyj.f

c plotxypr.f

c plotxyprj.f

c plotflux.f

c plotfluxpr.f

c

executable files:

c

c in comfile:interpolate

c in comfile2 :smooth,plot,plotj

c in comfile3 :filei,filej

super mongo interactive input macro files:

c

c surface.in (postscript:phase.ps)

c surface8.in (" " :intens.ps)

c surface9.in (" " :zoom.ps)

c surface12.in(" " :smooth.ps)

c contour12.in(" " :contsmooth.ps)

c contour.in(" " :contphase.ps)

c traj.in(" " :trajectory.ps)

c

to run the code type at the prompt:cx comfile

c :comfile

c

to smooth with a gaussian and see some trajectories: comfile2

c

to file those trajectories in postscript: comfile3

c

for statistical analysis of the model gen data:

c

structure~function~auto.f: an interactive program, needs X windows

c and Super Mongo plotting software.

c It calculates auto structure functions

c for a trajectory in all five frequencies,

c plots it on an X window and creates also

c a postscript file of the plot.

c

structure~function~cross.f:same as above for cross structure

c functions.

c

for analysis of the observational data:

c

main : read-data-spline.f:

c subprograms:structure-function.f and subroutines

c therein

c an interactive program, X windows and

c Super Mongo necessary. Searches the

c LFVAR data set, for a particular source,

```
c          files all observational data related
c          to the source in a file named after the
c          source, plots flux versus time, on an
c          X window, or makes a postscript of the
c          plot, identifies data gaps, calculates
c          the second derivatives of the curve, asks
c          for the desired lag, does a cubic spline
c          interpolation and calculates the flux
c          at those points, plots or files (postscript)
c          the result, and then goes on and calculates
c          and plots auto structure functions for
c          the source at the desired frequencies.
```

```
c
main: read~data~cross.f:
```

```
c          subprogram:structure~cross.f
```

```
c          same as above for cross structure functions.
```

```
c
detailed information on specifics are given in comments in the
main,modules and .inc , .ini files.
```

```
c
```

```
Maria J Pantazopoulou, Dec. 28 1995 , VT.
```


B.2 Fortran77 listing of read_data_spline.f

```
c   Program to read data from the Lowvar Dataset
program read_data_spline
c   Maria J. Pantazopoulou ,Jan 10 1996, Virginia Tech.
c
c   This program reads data from the Lowvar Dataset,
c   and files them in different files named by the source
c   coordinates. There is option for interpolation
c   of gaps in the data, using a cubic spline fit,
c   as well as interpolating the data to ? week intervals
c   as needed for the structure_function.f program.
c
implicit none
integer out,i,j,k,kmax,nlines,NPTS,NDAT,nxy,max
parameter(nlines=3070,NPTS=100,NDAT=NPTS,nxy=100)
integer f_code(nlines),frequency_code,IW
integer year(nlines),month(nlines),day(nlines)
integer days_of_mo
logical plotting,hcopy
character*7 source_name,source(0:nlines)
character*1 ch,ans
real flux_density(nlines),flux_er(nlines)
real ggdata(NDAT),date(NDAT),per_date(NDAT)
real ly1,ly2,period,per_data(NDAT)
real flux1(NDAT),flux2(NDAT),flux3(NDAT),flux4(NDAT)
real flux5(NDAT), fluxi(NDAT),xpoint,ypoint,xplot1
real yp1,ypn,y2(NDAT).xplot(nxy),yplot(nxy),yplot1
c
c   **initialize**
source(0)='pandora'
out=14
j=1
do i=1,NDAT
ggdata(i)=0.
date(i)=0.
per_date(i)=0.
per_data(i)=0.
flux1(i)=0.
flux2(i)=0.
flux3(i)=0.
flux4(i)=0.
flux5(i)=0.
```

```

fluxi(i)=0.
end do
c   open graphics window for plots
print*, 'want a window for plots? y/n'
read(5,29) ans
if(ans .eq. 'y')then
plotting=.true.
call plot_in
else
plotting=.false.
end if !ans
c
c   open device for postscript file of the plot
c   if needed
print*, 'want a postscript file? y/n'
print*, 'remember:type n if window is yes'
read(5,29) ans
if(ans .eq. 'y')then
hcopy=.true.
call plot_print
else
hcopy=.false.
end if !ans
c   ** end initialize **
c
c   read in data file:contains data for many different sources
c
open(10,file='LFVAR.DAT')
do i=1,2
read(10,11) ch
end do
c
c   sort data by source identifier, and create files named after
c   each source and put data related to those sources in each
c   equivalent file.
c
do while(.true.)
read(10,12,end=222) source_name,f_code(j),year(j),month(j),day(j)
*,flux_density(j),flux_er(j)
source(j)=source_name
if(source(j-1) .ne. source(j))then
if(source(j) .ne. '      ')then
out=out+1
open(out,file=source(j))

```

```

write(out,*) source(j)
c
c
end if
end if !source(j-1)
write(out,13) f_code(j),year(j),month(j),day(j),flux_density(j)
*,flux_er(j)
if(source(j) .eq. '      ')then
goto 222
else
j=j+1
end if !source(j) eq ' '
end do !while true
11 format(A1)
12 format(A7,I1,I2,I2,I2,F6.2,F5.2)
13 format(I1,I2,I2,I2,F6.2,F5.2)
222 continue
close(10)
c
c start reading source files created above, and read data,i.e
c frequency, observation dates,fluxes, flux error.
c In this stage the search is for data with the same frequency
c therefore the identifier is the frequency code.
c
c read in source file:
10 print*, 'type source name'
read(5,24) source_name
print*, 'type frequency code'
read(5,25) frequency_code
pause
open(21,file=source_name)
j=1
k=1
read(21,26) ch
do while(.true.)
read(21,23,end=333) f_code(j),year(j),month(j),day(j)
*,flux_density(j),flux_er(j)
if(f_code(j) .eq. frequency_code)then
ggdata(k)=flux_density(j)
c write the date of observation here as a date array, i.e
c in terms of year with desimals to facilitate calculations
c later on during data analysis.
c
c correct for leap years

```

```

c
if(month(j) .eq. 2)then
if((year(j) .eq. 80) .or. (year(j) .eq. 84))then
days_of_mo=29
else
days_of_mo=28
end if !year(j) leap
else if((month(j) .eq. 4) .or. (month(j) .eq. 6) .or.
* (month(j) .eq. 8) .or. (month(j) .eq. 10) .or.
* (month(j) .eq. 12))then
days_of_mo=30
else
days_of_mo=31
end if !month(j)
c
if((year(j) .eq. 80) .or. (year(j) .eq. 84))then
date(k)=real(year(j))+
* real((month(j)-1)*days_of_mo+day(j))/366.
else
date(k)=real(year(j))+
* real((month(j)-1)*days_of_mo+day(j))/365.
end if !year(j) leap
c      write(*,*) ggdata(k)
k=k+1
end if !f_code(j)
if(f_code(j) .eq. ' ')then
goto 333
else
j=j+1
end if !f_code(j)
end do !while true
23  format(I1,I2,I2,I2,F6.2,F5.2)
24  format(A7)
25  format(I1)
26  format(A1)
333  continue
close(21)
kmax=k-1
c
c      end of data extraction and sorting according to source and
c      frequency.
c      ***** start data analysis *****
do k=1,kmax
c      write(3,*) date(k),ggdata(k)

```

```

write(*,*) ggdata(k),date(k)
end do
print*,'kmax=',kmax
c   ***** plotting on an x window *****
c   **and or creating a postscript file *****
if(plotting .or. hcopy)then
c   set limits for plot, y axis(look at numbers on screen
c   print*,'type in limits for flux, format f5.2'
print*,'ly1 (f5.2)'
read(5,28) ly1
print*,'ly2 (f5.2)'
read(5,28) ly2
c
c   read in window y section
print*,'type window section (i1),max 5'
read(5,27) IW
call sm_limits(80.0,85.0,ly1,ly2)
call sm_window(1,-5,1,IW)
if(IW .eq. 1)then
call sm_box(1,2,0,0)
call sm_xlabel('year')
c   call sm_ylabel('flux')
else if(IW .eq.3)then
call sm_box(0,2,0,0)
call sm_ylabel('S(Jy)')
else
call sm_box(0,2,0,0)
c   call sm_ylabel('flux')
end if !IW
call sm_ptype(40.,1)
call plot(kmax,date,ggdata)
c
27 format(I1)
28 format(f5.2)
end if !plotting or hcopy .true.
c   choose period for interpolation
period=0.
print*,'set period in days'
read(*,31) period
write(*,*) 'period=',period
per_date(1)=80.5
c   find first point
xpoint=per_date(1)
c   find first derivatives of first and last point

```

```

yp1=(ggdata(2)-ggdata(1))/(date(2)-date(1))
ypn=(ggdata(kmax)-ggdata(kmax-1))/(date(kmax)
* -date(kmax-1))
call spline(date,ggdata,kmax,yp1,ypn,y2)
print*, 'the second derivatives are'
c   make a smooth graph of the spline here
xplot(1)=80.5
xplot1=xplot(1)
call splint(date,ggdata,y2,kmax,xplot1,yplot1)
yplot(1)=yplot1
do i=2,nxy-5
xplot(i)=xplot(i-1)+.05
xplot1=xplot(i)
call splint(date,ggdata,y2,kmax,xplot1,yplot1)
yplot(i)=yplot1
end do !smooth graph
do i=1,kmax
write(*,*) y2(i)
end do
period=period/365.
j=1
call splint(date,ggdata,y2,kmax,xpoint,ypoint)
per_data(1) = ypoint
write(*,*) ' per_data(1)', per_data(1)
do while(per_date(j) .lt. 84.6)
j=j+1
per_date(j)=per_date(j-1)+period
c   use these dates for interpolation
xpoint=per_date(j)
call splint(date,ggdata,y2,kmax,xpoint,ypoint)
per_data(j)=ypoint
max=j
if(frequency_code .eq. 1)then
flux1(1)=per_data(1)
flux1(j)=per_data(j)
end if
if(frequency_code .eq.2)then
flux2(1)=per_data(1)
flux2(j)=per_data(j)
end if
if(frequency_code .eq.3)then
flux3(1)=per_data(1)
flux3(j)=per_data(j)
end if

```

```

if(frequency_code .eq.4)then
flux4(1)=per_data(1)
flux4(j)=per_data(j)
end if
if (frequency_code .eq.5)then
flux5(1)=per_data(1)
flux5(j)=per_data(j)
end if
end do ! while per_date
write(*,*) 'max=',max
if(plotting .or. hcopy)then
call sm_ptype(63.,1)
call plot(nxy,per_date,per_data)
call sm_ptype(11.,1)
call plot(nxy,xplot,yplot)
call sm_conn(xplot,yplot,nxy-12)
end if !plotting
print*,'want to continue? y/n'
read(5,29) ans
29 format(A1)
if(ans .eq. 'y')then
do k=1,max
per_data(k)=0.
per_date(k)=0.
end do
do k=1,kmax
date(k)=0.
ggdata(k)=0.
end do
do k=1,nxy
xplot(k)=0.
yplot(k)=0.
end do
go to 10
else
if(hcopy)then
call sm_relocate(10.-5)
call sm_label(source_name)
call plot_print_close
end if !hcopy
end if !ans
c plotting interolated data
print*,'plotting interpolated data'
if(plotting)then

```

```

call sm_erase
call sm_window(1,1,1,1)
end if !plotting
6  print*, 'type in frequency code of the data you want to see'
read(*,30) I
do k=1,max
fluxi(k)=0.
end do
if(i .eq. 1) then
do k=1,max
fluxi(k)=flux1(k)
end do
else if(i .eq. 2) then
do k=1,max
fluxi(k)=flux2(k)
end do
else if(i .eq. 3)then
do k=1,max
fluxi(k)=flux3(k)
end do
else if(i .eq. 4)then
do k=1,max
fluxi(k)=flux4(k)
end do
else if(i .eq. 5)then
do k=1,max
fluxi(k)=flux5(k)
end do
else
goto 444
end if
pause
if(plotting)then
write(*,*) 'per_date  'fluxi'
do k=1,max
write(*,*) per_date(k).fluxi(k)
end do
call sm_limits(per_date(1).per_date(max),0.0,20.)
call sm_box(1,2,0,0)
call plot(nxy,per_date.fluxi)
call sm_conn(per_date.fluxi.nxy)
pause
call sm_erase
end if !plotting

```



```

goto 6
444 continue
pause
if(plotting)then
call plot_close
end if !plotting
print*, 'calculating auto structure functions'
print*, 'do you want to see a plot?/else hardcopy'
read(5,*) ch
if(ch .eq. 'y')then
call plot_in
else
call plot_print
end if
5 print*, 'type in freq code of the data for which you want Str. f'
read(*,30) I
do k=1,max
fluxi(k)=0.
end do
30 format(i1)
if(i .eq. 1)then
do k=1,max
fluxi(k)=flux1(k)
end do
end if
if(i .eq. 2)then
do k=1,max
fluxi(k)=flux2(k)
end do
end if
if(i .eq. 3)then
do k=1,max
fluxi(k)=flux3(k)
end do
end if
if(i .eq. 4)then
do k=1,max
fluxi(k)=flux4(k)
end do
end if
if(i .eq. 5)then
do k=1,max
fluxi(k)=flux5(k)
end do

```

```

end if
if(i .eq. 9)then
if(ch .eq. 'y')then
call plot_close
else
call plot_print_close
end if
end if
pause
call structure_function(max,fluxi,fluxi,source_name,
*I)
go to 5
31 format(f5.2)
stop
end
c
c
subroutine plot_in
integer sm_device
c   real ly2,lx2
c   ly2=15.
c   lx2=85.
if(sm_device('x11') .lt. 0)then
print*, 'can"t open device'
stop
endif
call sm_graphics
call sm_defvar("TeX_strings", "1")
call sm_expand(1.)
C   call sm_limits(80.0,lx2,0.0,ly2)
C   call sm_box(1,2,0,0)
end
c
subroutine plot_print
call sm_device('POSTSCRIPT :SY mv F"output filename.ps?")
end
subroutine plot(nxy,x,y)
INTEGER NXY
REAL X(NXY),Y(NXY)
C   REAL YCALC(NXY)
C
N=NXY
do 3 i=1,2
CALL SM.POINTS(X,Y,NXY)

```

```

C
C   CALL SM.CONN(X,Y,NXY)
3   CONTINUE
return
end
c
subroutine plot_close
integer i
CALL SM_ALPHA
C   CALL SM.HARDCOPY
PRINT *, 'HIT A NEGATIVE NUMBER TO EXIT'
CALL SM_REDRAW(0)
READ(5,*) I
return
end
c
subroutine plot_print_close
integer i
CALL SM_ALPHA
CALL SM_HARDCOPY
PRINT*, 'HIT A NEGATIVE NUMBER TO EXIT POSTSCR.DEVICE'
CALL SMREDRAW(0)
READ(5,*) I
return
end
c
SUBROUTINE LINEFIT(X,Y,YCALC,A,B,N)
INTEGER N,I
REAL X(1),Y(1),YCALC(1),A,B,SXY,SYY,SXX
REAL SUMX,SUMY,SUMXY,SUMX2,SUMY2,XI,YI
C
SUMX=0.0
SUMY=0.0
SUMXY=0.0
SUMX2=0.0
SUMY2=0.0
DO 10 I=1,N
XI=X(I)
YI=Y(I)
SUMX=SUMX+XI
SUMY=SUMY+YI
SUMXY=SUMXY+XI*YI
SUMX2=SUMX2+XI*XI
SUMY2=SUMY2+YI*YI

```

```

10 CONTINUE
SXX=SUMX2-SUMX*SUMX /N
SXY=SUMXY-SUMX*SUMY /N
SYY=SUMY2-SUMY*SUMY /N
B=SXY /SXX
A=(SUMY-B*SUMX)/N
DO 20 I=1,N
YCALC(I)=A+B*X(I)
20 CONTINUE
RETURN
END
C
SUBROUTINE spline(x,y,n,yp1,ypn,y2)
INTEGER n,NMAX
REAL yp1,ypn,x(n),y(n),y2(n)
PARAMETER (NMAX=500)
INTEGER i,k
REAL p,qn,sig,un,u(NMAX)
if (yp1.gt..99e30) then
y2(1)=0.
u(1)=0.
else
y2(1)=-0.5
u(1)=(3./(x(2)-x(1)))*((y(2)-y(1))/(x(2)-x(1))-yp1)
endif
do 11 i=2,n-1
sig=(x(i)-x(i-1))/(x(i+1)-x(i-1))
p=sig*y2(i-1)+2.
y2(i)=(sig-1.)/p
u(i)=(6.*((y(i+1)-y(i))/(x(i+
*1)-x(i))-(y(i)-y(i-1))/(x(i)-x(i-1)))/(x(i+1)-x(i-1))-sig*
*u(i-1))/p
11 continue
if (ypn.gt..99e30) then
qn=0.
un=0.
else
qn=0.5
un=(3./(x(n)-x(n-1)))*(ypn-(y(n)-y(n-1))/(x(n)-x(n-1)))
endif
y2(n)=(un-qn*u(n-1))/(qn*y2(n-1)+1.)
do 12 k=n-1,1,-1
y2(k)=y2(k)*y2(k+1)+u(k)
12 continue

```

```

return
END
C (C) Copr. 1986-92 Numerical Recipes Software !" ;'(1z9.
c
SUBROUTINE splint(xa,ya,y2a,n,x,y)
INTEGER n
REAL x,y,xa(n),y2a(n),ya(n)
INTEGER k,khi,klo
REAL a,b,h
klo=1
khi=n
1  if (khi-klo.gt.1) then
k=(khi+klo)/2
if(xa(k).gt.x)then
khi=k
else
klo=k
endif
goto 1
endif
h=xa(khi)-xa(klo)
if (h.eq.0.) pause 'bad xa input in splint'
a=(xa(khi)-x)/h
b=(x-xa(klo))/h
y=a*ya(klo)+b*ya(khi)+((a**3-a)*y2a(klo)+(b**3-b)*y2a(khi))*(h**
*2)/6.
return
END
C (C) Copr. 1986-92 Numerical Recipes Software !" ;'(1z9.

```

B.3 Fortran77 listing of structure_function.f

```
subroutine structure_function(NDAT,fdata,ggdata,source_
*name,frequency_code)
C   driver for routines moment, dfginfinity,dfg
C   Maria J. Pantazopoulou, Dec. 29 1995, Virginia Tech.
C
REAL PI
INTEGER NDAT,NPTS,mdat,Tshift,nshift
PARAMETER(PI=3.14159265,NPTS=512)
INTEGER i,j,frequency_code
character*7 source_name
REAL adev,ave,curt,fdata(NPTS),sdev,skew,var
REAL x(0:NPTS),y(NPTS),gdata(NPTS),ggdata(NPTS)
REAL structure_f,str_f(0:NPTS),meang,meanf,varg,varf
REAL dinfinity,ycalc(NPTS)
C
if(frequency_code .eq. 1)then
freq='318 MHz'
else if(frequency_code .eq. 2)then
freq='430 MHz'
else if(frequency_code .eq. 3)then
freq='606 MHz'
else if(frequency_code .eq. 4)then
freq='880 MHz'
else
freq='1400 MHz'
end if
write(*,*) 'frequency_code:',freq
do i=1,NDAT
x(i)=i
y(i)=i
end do
do i=1,NDAT
write(*,*) fdata(i),ggdata(i)
end do
C   read in data for comparison, freq 430 MHz data.
c   open(10,file='slicei90f1.dat')
c   do i=1,NDAT
c   read(10,*) x(i),fdata(i)
c   end do
c   close(10)
write(*,'(1x,a/)') 'Moments of the data'
```

```

call moment(fdata,NDAT,ave,adev,sdev,var,skew,curt)
write(*,'(1x,t29,a/)') 'Calculated'
write(*,'(1x,a,t25,f12.4)') 'Mean :',ave
write(*,'(1x,a,t25,f12.4)') 'Average Deviation :',adev
write(*,'(1x,a,t25,f12.4)') 'Standard Deviation :',sdev
write(*,'(1x,a,t25,f12.4)') 'Variance :',var
write(*,'(1x,a,t25,f12.4)') 'Skewness :',skew
write(*,'(1x,a,t25,f12.4)') 'Kurtosis :',curt
meanf=ave
varf=var
C
C   read in data at other frequencies:
c   open(20,file='slicei90f2.dat')
c   do i=1,NDAT
c   read(20,*) y(i),ggdata(i)
c   end do
c   close(20)
write(*,'(1x,a/)') 'Moments of the data:gdata'
call moment(ggdata,NDAT,ave,adev,sdev,var,skew,curt)
write(*,'(1x,t29,a/)') 'Calculated'
write(*,'(1x,a,t25,f12.4)') 'Mean :',ave
write(*,'(1x,a,t25,f12.4)') 'Average Deviation :',adev
write(*,'(1x,a,t25,f12.4)') 'Standard Deviation :',sdev
write(*,'(1x,a,t25,f12.4)') 'Variance :',var
write(*,'(1x,a,t25,f12.4)') 'Skewness :',skew
write(*,'(1x,a,t25,f12.4)') 'Kurtosis :',curt
meang=ave
varg=var
pause
C
C   normalize
call dfginfinity(meanf,meang,varf,varg,dinfinity)
write(*,*) '***'
write(*,'(1x,a,t25,f12.4)') 'Dinfinity :',dinfinity
C
C   shift gdata by Tshift
do nshift=1,NDAT/2 +1
Tshift=nshift-1
mdat=NDAT-Tshift
do j=1,mdat
gdata(j)=ggdata(j+Tshift)
end do
C   calculate the structure function
call dfg(fdata,gdata,mdat,structure_f)

```

```

C
C   write(*,'(1x,a,t25,f12.4)') 'Structure_f :',structure_f
C   normalize
C
x(Tshift)=real(Tshift)
str_f(Tshift)=structure_f/dinfinity
C   write(*,*) '***'
C   write(21,*) x(Tshift),str_f(Tshift)
C   write(*,'(1x,a,t25,f12.4)') 'Str_f/Dinf :',str_f(Tshift)
end do
call plot1(NDAT/2+1,x,str_f,source_name,frequency_code)
C   call parabolic_fit(x,str_f,ycalc)
C   call plot1(NDAT/2,x,ycalc)
END
C
SUBROUTINE moment(data,n,ave,adev,sdev,var,skew,curt)
INTEGER n
REAL adev,ave,curt,sdev,skew,var,data(n)
INTEGER j
REAL p,s,ep
if(n.le.1)pause 'n must be at least 2 in moment'
s=0.
do 11 j=1,n
s=s+data(j)
11 continue
ave=s/n
adev=0.
var=0.
skew=0.
curt=0.
ep=0.
do 12 j=1,n
s=data(j)-ave
ep=ep+s
adev=adev+abs(s)
p=s*s
var=var+p
p=p*s
skew=skew+p
p=p*s
curt=curt+p
12 continue
adev=adev/n
var=(var-ep**2/n)/(n-1)

```



```

sdev=sqrt(var)
if(var.ne.0.)then
skew=skew/(n*sdev**3)
curt=curt/(n*var**2)-3.
else
pause 'no skew or kurtosis when zero variance in moment'
endif
return
END
C (C) Copr. 1986-92 Numerical Recipes Software !"i'(1z9.
c
subroutine dfginfinity(meanf,meang,varf,varg,dinfinity)
C calculates the structure function D_infinity of two sets
C of data, for normalization
C Maria J. Pantazopoulou, Dec. 29, Virginia Tech.
real meanf,meang,varf,varg,dinfinity
dinfinity=(meanf-meang)**2 +varf+varg
return
end
subroutine dfg(fdata,gdata,mdata,structure_function)
C calculates the structure function of two sets of data,mdata pairs.
C Maria J. Pantazopoulou, Dec. 29, Virginia Tech.
integer mdat,j
real fdata(mdat),gdata(mdat),structure_function
real sum
sum=0.
do 11 j=1,mdat
sum=sum+(fdata(j)-gdata(j))**2
11 continue
structure_function=sum/mdat
c
return
end
SUBROUTINE PLOT1(NXY,X,Y,source_name,frequency_code)
INTEGER NXY
C PARAMETER (NXY=512)
C INTEGER SM_DEVICE
INTEGER I,frequency_code
character*7 source_name,freq
c character*1 ch
REAL X(NXY),Y(NXY)
C REAL YCALC(NXY)
REAL LY2,LX2
C

```

```

N=NX Y
OUT=1
LY2=1.9
LX2=NX Y
c*****
DO 3 I=1,2
c    CALL SM_GRAPHICS
CALL SM_DEFVAR("TeX_strings", "1")
CALL SM_EXPAND(1.)
c    call sm_erase
CALL SM_LIMITS(0.0,LX2,0.0,LY2)
if(frequency_code .eq. 1)then
call sm_window(1,-5,1,1)
call sm_box(1,2,0,0)
call sm_expand(1.5)
call sm_xlabel('lag(.17 years)')
call sm_ylabel('D_318 x 318')
call sm_expand(1.)
else if(frequency_code .eq. 2)then
call sm_window(1,-5,1,2)
call sm_box(0,2,0,0)
call sm_expand(1.5)
call sm_ylabel('D_430 x 430')
call sm_expand(1.)
else if(frequency_code .eq. 3)then
call sm_window(1,-5,1,3)
call sm_box(0,2,0,0)
call sm_expand(1.5)
call sm_ylabel('D_606 x 606')
call sm_expand(1.)
else if(frequency_code .eq. 4)then
call sm_window(1,-5,1,4)
call sm_box(0,2,0,0)
call sm_expand(1.5)
call sm_ylabel('D_880 x 880')
call sm_expand(1.)
else
call sm_window(1,-5,1,5)
call sm_box(0,2,1,0)
call sm_expand(1.5)
call sm_ylabel('D_1400 x 1400')
call sm_xlabel(source_name)
call sm_expand(1.0)
end if

```

```

CALL SM_PTYPE(63.,1)
CALL SM_POINTS(X,Y,NXY)
C   CALL LINEFIT1(X,Y,YCALC,A,B,N)
C   DO J=1,NXY
C   Y(J)=YCALC(J)
C   END DO
C
CALL SM_CONN(X,Y,NXY)
3   CONTINUE
return
END
SUBROUTINE LINEFIT1(X,Y,YCALC,A,B,N)
INTEGER N,I
REAL X(1),Y(1),YCALC(1),A,B,SXY,SYY,SXX
REAL SUMX,SUMY,SUMXY,SUMX2,SUMY2,XI,YI
C
SUMX=0.0
SUMY=0.0
SUMXY=0.0
SUMX2=0.0
SUMY2=0.0
DO 10 I=1,N
XI=X(I)
YI=Y(I)
SUMX=SUMX+XI
SUMY=SUMY+YI
SUMXY=SUMXY+XI*YI
SUMX2=SUMX2+XI*XI
SUMY2=SUMY2+YI*YI
10  CONTINUE
SXX=SUMX2-SUMX*SUMX /N
SXY=SUMXY-SUMX*SUMY /N
SYY=SUMY2-SUMY*SUMY /N
B=SXY /SXX
A=(SUMY-B*SUMX)/N
DO 20 I=1,N
YCALC(I)=A+B*X(I)
20  CONTINUE
RETURN
END
C

```

B.4 Fortran77 listing of Phaethon.f

```
C PROGRAM TO CALCULATE WAVE PROPAGATION IN
C ISM
PROGRAM PHAETHON
C MAIN ROUTINE
C
C
INCLUDE 'phaeth1.inc'
C
C ***** set defaults *****
CALL INIT~ PHAETHON(IDUM, A, LF, POWERSPEC, CROSSING)
C ***** change defaults *****
C infile:phaethon.ini
C
C read in:
C :the random seed, IDUM
C :the Kolmogorov constant A (alpha)
C :the Fresnel scale LF,
C :outfiles for plots choices: Power Spectrum plot,
C :choice of interpolation module(subprogram)
C
C INPUT FILE phaethon.ini
C *****
OPEN(2, FILE='phaethon.ini')
READ(2, *) IDUM
READ(2, *) A
READ(2, *) LF
READ(2, *) CH
IF (CH .EQ. 'T') THEN
POWERSPEC=.TRUE.
ELSE
POWERSPEC=.FALSE.
END IF
READ(2, *) CH
IF (CH .EQ. 'T') THEN
CROSSING=.TRUE.
ELSE
CROSSING=.FALSE.
END IF
C *****
CLOSE(2)
write(*, *) 'random seed:', IDUM
```

```

write(*,*) 'Kolmogorov constant:', A
write(*,*) 'Fresnel scale(parameter):',LF
write(*,*) 'outfile for debugging:fort.OUT'
C *****
C
C   outfile:
OUT=1
write(*,*) 'OUT=',OUT
DELTA=1.0
C
C   *** the following dimension parameters have to be set ***
C   *** in the declaration file :phaeth1.inc *****
C   The maximum row and column,i.e the size of the grid
C   MAXR
C   MAXC
C   NCOL=2*MAXR*MAXC
C   NPOINT=MAXR*MAXC
C   NROW=2*MAXR
C   *****set for the FFT routines *****
KR=2*MAXR
ISIGN=1
NDIM=2
NN(1)=MAXR
NN(2)=MAXC
C   ***** initialize arrays *****
DO I=1, MAXR
DO J=1, MAXC
G(I,J)=0.
END DO
END DO
DO I=1, NCOL
DATA(M)=0.
END DO
DO J=1,MAXR
DO I=1,MAXC
WAVENO(I,J)=0.
END DO
END DO
DO I=1,NPOINT
PSPECTRUM(I)=0.
END DO
C *****
C   WRITE(1,201)
201 FORMAT(2X,'set random grid, of dimension A1xA1'/

```

```

* 2X,'where A1 IS a power of TWO')
DO I=1, MAXR
DO J=1,MAXC
C   G(I,J)=DELTA*(0.5-1.*RAN1(IDUM))
G(I,J)=DELTA*RAN1(IDUM)
END DO
END DO
C   write out for debugging *****
C   DO I=1,MAXR
C   WRITE(1,101)(G(I,J),J=1,MAXC)
C 101 FORMAT(2X,8F7.3)
C   END DO
C
C           *****
C   WRITE(1,202)
202 FORMAT(2X,'—————')
WRITE(1,203)
203 FORMAT(2X,'input the data array of length 2xA1xA1 in'/
*2X,'a form of complex number. The real part is a grid point'/
*2X,'followed by the imaginary part which is zero.')
```

```

DO I=1, MAXR
DO J=1, MAXC
G1(I,2*J-1)=G(I,J)
G1(I,2*J)=0.0
C   WRITE(1,150) G1(I,2*J-1)
C   WRITE(1,150) G1(I,2*J)
END DO
END DO
150 FORMAT(E10.2)
CALL COMPLEXDAT(NCOL,MAXR,KR,G1,DATA)
WRITE(1,204)
204 FORMAT(2X,'*****')
WRITE(1,205)
205 FORMAT(2X,'The one dim.array DATA is replaced by its'/
*2X,'Fourier transform.')
```

```

C   write out for debugging *****
C   WRITE(1,104) (DATA(M),M=1,NCOL)
104 FORMAT(8E10.2)
C
C           *****
C
CALL FOURN(DATA,NN,NDIM,ISIGN)
C
C   write out for debugging *****
C   WRITE(1,102) (DATA(M),M=1,NCOL)
102 FORMAT(8E10.2/)
```

```

C   WRITE(1,230)
230 FORMAT(2X,'-----')
C
C           *****
WRITE(1,231)
231 FORMAT(2X,'separate DATA into real and imaginary')
C+++++
OPEN(80,FILE='MIXED1.DAT')
WRITE(80,232) (DATA(M),M=1,NCOL)
232 FORMAT(8E10.2)
CLOSE(80)
C   CALL REALDAT(NCOL,MAXR,KR,DATA,MIX)
OPEN(80,FILE='MIXED1.DAT')
DO I=1, NROW
READ(80,240) (MIX(I,J),J=1,MAXC)
240 FORMAT(8E10.2)
END DO
CLOSE(80)
C+++++
WRITE(1,235)
235 FORMAT(2X,'***** real *****')
OPEN(81,FILE='REAL1.DAT')
DO I=1,NROW
C   WRITE(1,234) (MIX(I,J),J=MAXC,2)
WRITE(81,250) (MIX(I,J),J=1,MAXC,2)
234 FORMAT(8E10.2)
250 FORMAT(E10.2)
END DO
CLOSE(81)
WRITE(1,237)
237 FORMAT(2X,'***** imaginary *****')
OPEN(82,FILE='IMAG1.DAT')
MAXR1=MAXR+1
DO I=1, NROW
C   WRITE(1,236) (MIX(I,J-1),J=3,MAXR1,2)
WRITE(82,255) (MIX(I,J-1),J=3,MAXR1,2)
236 FORMAT(8E10.2)
255 FORMAT(E10.2)
END DO
CLOSE(82)
OPEN(81,FILE='REAL1.DAT')
DO I=1, NPOINT
READ(81,*) MIX1(I)
END DO
CLOSE(81)

```

```

OPEN(82,FILE='IMAG1.DAT')
DO I=1, NPOINT
READ(82,*) MIX2(I)
END DO
CLOSE(82)
C
CALL FREQ(MAXR,FREQ1,FREQ2)
C   write out for debugging   *****
C   WRITE(1,300) (FREQ1(I),I=1,MAXR)
300 FORMAT(E10.2)
C   WRITE(1,502) (FREQ2(I),I=1,MAXR)
502 FORMAT(8E10.2)
WRITE(1,510)
510 FORMAT(2X,'UNWRAP WAVEVECTOR,IN FREQ.ORDER')
WRITE(1,511)
511 FORMAT(2X,'WAVEVEC',2X,'FREQ1',2X,'FREQ2')
C   *****
OPEN(90,FILE='WAVEVEC1.DAT')
DO J=1,MAXR
DO I=1,MAXC
WAVENO(I,J)=SQRT((FREQ1(I)**2)+(FREQ2(J)**2))
C   WRITE(1,503) WAVENO(I,J),FREQ1(I),FREQ2(J)
WRITE(90,504) WAVENO(I,J)
504 FORMAT(E10.2)
503 FORMAT(4E10.2)
END DO
END DO
CLOSE(90)
WRITE(1,206)
WRITE(1,207)
207 FORMAT(2X,'=====')
206 FORMAT(2X,'GIVE THE ARRAY WAVEVEC.DAT(M) THE
*DESIRED POWER LAW'/
*2X,'DEPENDENCE.')
```

C

C Give the inner scale cutoff $\exp(-k/k_2)$, where k_2 is
C the Nyquist frequency.

```

OPEN(90,FILE='WAVEVEC1.DAT')
DO M=1, NPOINT
READ(90,*) WAVEVEC(M)
END DO
DO M=2,NPOINT
WAVEVEC(M)=(WAVEVEC(M))**(-A/2.)*EXP(-(WAVEVEC(M)/.5)**2)
END DO
```



```

WAVEVEC(1)=0.
DO M=1,NPOINT
C   WRITE(1,515) WAVEVEC(M)
515 FORMAT(E12.4)
END DO
CLOSE(90)
WRITE(1,211)
211 FORMAT(2X,'—————')
OPEN(95,FILE='INPUTS.DAT')
DO I=1, NPOINT
MIX1(I)=MIX1(I)*WAVEVEC(I)
MIX2(I)=MIX2(I)*WAVEVEC(I)
C   WRITE(1,520) MIX1(I),MIX2(I)
WRITE(95,525) MIX1(I)
WRITE(95,525) MIX2(I)
520 FORMAT(2E12.4)
525 FORMAT(E12.4)
END DO
CLOSE(95)
C   prepare outfile for plot of the Power Spectrum
C   +++++
IF(POWERSPEC)THEN
OPEN(90,FILE='WAVEVEC1.DAT')
DO I=1, NPOINT
READ(90,*) WAVEVEC2(I)
END DO
CLOSE(90)
OPEN(99,FILE='PSPECTRUM.DAT')
DO I=2, NPOINT
PSPECTRUM(I)=ALOG10(MIX1(I)**2+MIX2(I)**2)
WAVEVEC2(I)=ALOG10(WAVEVEC2(I))
END DO
PSPECTRUM(1)=1.
WAVEVEC2(1)=0.
DO I=1,NPOINT
WRITE(99,700) WAVEVEC2(I),PSPECTRUM(I)
700 FORMAT(F7.3,1X,E10.2)
END DO
CLOSE(99)
END IF !powerspec true
C   +++++
C
WRITE(1,208)
C   *****INITIALIZE DATA ARRAY*****

```

```

DO M=1, NCOL
DATA(M)=0.
END DO
C *****
OPEN(95,FILE='INPUTS.DAT')
DO M=1, NCOL
READ(95,*) DATA(M)
END DO
WRITE(1,209)
208 FORMAT(2X,'*****')
209 FORMAT(2X,'USE THE INVERSE FOURIER TRANSFORM TO
* RECOVER'/
*2X,'THE SPACE SERIES')
ISIGN=-1
CALL FOURN(DATA,NN,NDIM,ISIGN)
C WRITE(1,530) (DATA(M),M=1,NCOL)
CLOSE(95)
WRITE(1,540)
540 FORMAT(2X,'*****')
OPEN(96,FILE='FINGRID1')
WRITE(1,545)
545 FORMAT(2X,'RECOVER REAL GRID')
C WRITE(1,535) (DATA(M),M=1,NCOL,2)
535 FORMAT(8E10.2)
530 FORMAT(4E12.4)
DO M=1,NCOL
DATA(M)=REAL(DATA(M)/NPOINT)
END DO
WRITE(96,536)(DATA(M),M=1,NCOL,2)
536 FORMAT(E10.2)
CLOSE(96)
C MAKE A FILE FOR XII AND YII FOR THE 3-D PLOT OF THE GRID
OPEN(12,FILE='XANDY')
DO I=1,MAXR
DO J=1,MAXC
II=J+(I-1)*MAXR
C WRITE(*,*) II
IXI(II)=J
IYI(II)=I
END DO
END DO
WRITE(12,1001) (IXI(II),IYI(II),II=1,NPOINT)
1001 FORMAT(2I3)
CLOSE(12)

```

```

C
C *****
C FIND THE WAVEVECTOR COMPONENTS FOR A COLUMN OF
C DATA(NCOL)
C IN X DIMENSION, I.E ,KX
CALL SETUPFR(MAXR,KR,NCOL,FREQ1,FREQ2,FFX,FREQX1)
C WRITE(1,701) (FFX(I),I=1,NCOL)
701 FORMAT(F7.3)
C IN Y DIRECTION, I.E ,KY
CALL SETUPFRY(MAXR,KR,NCOL,FREQ1,FFY)
C WRITE(1,702) (FFY(I),I=1,NCOL)
702 FORMAT(F7.3)
C *****
C CALCULATE THE SECOND DERIVATIVES OF THE PHASE(FINGRID1)
OPEN(95,FILE='INPUTS.DAT')
DO K=1, NCOL
READ(95,*) PHI(K)
END DO
CLOSE(95)
C *****INITIALIZE DATA ARRAY*****
DO I=1,NCOL
DATA(I)=0.
END DO
C *****
DO M=1,NCOL
DATA(M)=-4*9.8596*PHI(M)*FFX(M)**2
C WRITE(1,710) DATA(M)
710 FORMAT(E10.2)
END DO
ISIGN=-1
CALL FOURN(DATA,NN,NDIM,ISIGN)
WRITE(1,703)
703 FORMAT(2X,'*****PHIXX*NPOINT*****')
C WRITE(1,707) (DATA(M),M=1,NCOL)
707 FORMAT(8E10.2)
CALL REALDAT(NCOL,MAXR,KR,DATA,G1)
CALL OGRID(KR,MAXR,G1,G)
DO I=1,MAXR
DO J=1,MAXR
PHIXX(I,J)=G(I,J)
END DO
END DO
WRITE(1,709)
709 FORMAT(2X,'*****PHIXX*****')

```

```

DO I=1,MAXR
C   WRITE(1,711) (PHIXX(I,J),J=1,MAXR)
END DO
711 FORMAT(8E10.2)
C   *****INITIALIZE DATA ARRAY*****
DO I=1,NCOL
DATA(I)=0.
END DO
C   *****
DO M=1,NCOL
DATA(M)=-4*9.8596*PHI(M)*FFY(M)**2
C   WRITE(1,714) DATA(M)
714 FORMAT(E10.2)
END DO
ISIGN=-1
CALL FOURN(DATA,NN,NDIM,ISIGN)
WRITE(1,716)
716 FORMAT(2X,'*****PHIYY*NPOINT*****')
C   WRITE(1,717) (DATA(M),M=1,NCOL)
717 FORMAT(8E10.2)
CALL REALDAT(NCOL,MAXR,KR,DATA,G1)
CALL OGRID(KR,MAXR,G1,G)
DO I=1,MAXR
DO J=1,MAXR
PHIYY(I,J)=G(I,J)
END DO
END DO
WRITE(1,718)
718 FORMAT(2X,'*****PHIYY*****')
DO I=1,MAXR
C   WRITE(1,719) (PHIYY(I,J),J=1,MAXR)
END DO
719 FORMAT(8E10.2)
C   *****INITIALIZE DATA ARRAY*****
DO I=1,NCOL
DATA(I)=0.
END DO
C   *****
DO M=1,NCOL
IF(FFX(M) .EQ. 0.5) THEN
FFX(M)=0.
END IF
IF(FFY(M) .EQ. 0.5) THEN
FFY(M)=0.

```

```

END IF
DATA(M)=-4*9.8596*PHI(M)* FFY(M)*FFX(M)
C   WRITE(1,720) DATA(M)
720 FORMAT(E10.2)
END DO
ISIGN=-1
CALL FOURN(DATA,NN,NDIM,ISIGN)
WRITE(1,721)
721 FORMAT(2X,'*****PHIXY*NPOINT*****')
C   WRITE(1,722) (DATA(M),M=1,NCOL)
722 FORMAT(8E10.2)
CALL REALDAT(NCOL,MAXR,KR,DATA,G1)
CALL OGRID(KR,MAXR,G1,G)
DO I=1,MAXR
DO J=1,MAXR
PHIXY(I,J)=G(I,J)
END DO
END DO
WRITE(1,723)
723 FORMAT(2X,'*****PHIXY*****')
C   DO I=1,MAXR
C   WRITE(1,724) (PHIXY(I,J),J=1,MAXR)
C   END DO
724 FORMAT(8E10.2)
C
C   CALCULATE THE EIGENVALUES OF THE TRASFORMATION MA-
TRIX
C   TO FIND THE INTENSITY AT THE OBSERVER'S PLANE
C   *****
CALL EIGENV(MAXR,LF,PHIXX,PHIYY,PHIXY,LAMBDA1,LAMBDA2)
WRITE(1,799)
799 FORMAT(2X,'*****LAMBDA1*****')
C   DO I=1,MAXR
C   WRITE(1,800) (LAMBDA1(I,J),J=1,MAXC)
C   END DO
WRITE(1,801)
801 FORMAT(2X,'*****LAMBDA2*****')
C   DO I=1,MAXR
C   WRITE(1,800) (LAMBDA2(I,J),J=1,MAXC)
C   END DO
CALL INTENSITY(MAXR,LAMBDA1,LAMBDA2,INTENS)
WRITE(1,802)
802 FORMAT(2X,'*****INTENSITY*****')
C   DO I=1,MAXR

```

```

C   WRITE(1,800) (INTENS(I,J),J=1,MAXR)
C   END DO
C   prepare surface points (1-D array) for the
C   interpolation module triangl.f (Dec. 16 1994)
DO I=1,MAXR
DO J=1,MAXC
II=J+(I-1)*MAXR
Z(II)=INTENS(I,J)
END DO
END DO
WRITE(1,850)
850  FORMAT(2X,'*****Z(I) ARRAY*****')
WRITE(1,*) (Z(I),I=1,NPOINT)
800  FORMAT(8E10.2)
C   FIND FIRST DERIVATIVE PHIX
C   *****
DO I=1,NCOL
DATA(I)=0.
END DO
OPEN(95,FILE='INPUTS.DAT')
DO K=1,NPOINT
READ(95,*) PHI1(K),PHI2(K)
END DO
CLOSE(95)
OPEN(38,FILE='FIRSTDER.DAT')
DO K=1,NPOINT
PHI1(K)=-PHI1(K)
WRITE(38,730) PHI2(K)
WRITE(38,730) PHI1(K)
730  FORMAT(E12.4)
END DO
CLOSE(38)
DO I=1,NCOL
PHI(I)=0.
END DO
OPEN(38,FILE='FIRSTDER.DAT')
DO M=1,NCOL
READ(38,*) PHI(M)
END DO
CLOSE(38)
DO M=1,NCOL
IF(FFX(M) .EQ. 0.5) THEN
FFX(M)=0.
END IF

```

```

DATA(M)=2*3.14*PHI(M)*FFX(M)
END DO
ISIGN=-1
CALL FOURN(DATA,NN,NDIM,ISIGN)
WRITE(1,725)
725 FORMAT(2X,'*****PHIX*NPOINT*****')
WRITE(1,726) (DATA(M),M=1,NCOL)
726 FORMAT(8E10.2)
CALL REALDAT(NCOL,MAXR,KR,DATA,G1)
CALL OGRID(KR,MAXR,G1,G)
DO I=1,MAXR
DO J=1,MAXR
PHIX(I,J)=G(I,J)
END DO
END DO
WRITE(1,740)
740 FORMAT(2X,'*****PHIX*****')
DO I=1,MAXR
WRITE(1,741) (PHIX(I,J),J=1,MAXR)
END DO
741 FORMAT(8E10.2)
C
C   FIND FIRST DERIVATIVE PHIY
C   *****
C   INITIALIZE DATA ARRAY
DO I=1,NCOL
DATA(I)=0.
END DO
C   *****
DO M=1,NCOL
IF(FFY(M) .EQ. 0.5) THEN
FFY(M)=0.
END IF
DATA(M)=2*3.14*PHI(M)*FFY(M)
END DO
ISIGN=-1
CALL FOURN(DATA,NN,NDIM,ISIGN)
WRITE(1,742)
742 FORMAT(2X,'*****PHIY*NPOINT*****')
WRITE(1,743) (DATA(M),M=1,NCOL)
743 FORMAT(8E10.2)
CALL REALDAT(NCOL,MAXR,KR,DATA,G1)
CALL OGRID(KR,MAXR,G1,G)
DO I=1,MAXR

```

```

DO J=1,MAXR
PHIY(I,J)=G(I,J)
END DO
END DO
WRITE(1,744)
744 FORMAT(2X,'*****PHIY*****')
DO I=1,MAXR
WRITE(1,745) (PHIY(I,J),J=1,MAXR)
END DO
745 FORMAT(8E10.2)
CALL POSITION(MAXR,NPOINT,LF,PHIX,PHIY,X,Y)
C ****prepare file for plot*****
OPEN(70,FILE='tridata')
WRITE(70,901)
901 FORMAT(2X,'**Z(I) array*****')
WRITE(70,746) (Z(I),I=1,NPOINT)
WRITE(1,747)
WRITE(70,747)
747 FORMAT(2X,'**X(I) array(position vec. not on grid point)**')
WRITE(1,746) (X(I),I=1,NPOINT)
WRITE(3,911) (X(I),Y(I),Z(I),I=1,NPOINT)
WRITE(70,746) (X(I),I=1,NPOINT)
WRITE(1,748)
WRITE(70,748)
911 FORMAT(3E10.3)
748 FORMAT(2X,'**Y(I) array(position vec. not on grid point)**')
WRITE(1,746) (Y(I),I=1,NPOINT)
WRITE(70,746) (Y(I),I=1,NPOINT)
746 FORMAT(4E10.2)
CLOSE(70)
C
IF(CROSSING)THEN
CALL INIT~ CELL ~ INTERPOL(FOLD, CAUSTIC)
CALL TETRAGONE(X,Y,NPOINT,MAXR,AVEINTENSITY)
END IF
C
IF(.NOT. (CROSSING))THEN
NPOINTSIX=6*NPOINT
CALL TRIANGULATION(X,Y,Z,NPOINT,MAXR,NPOINTSIX)
END IF
C
STOP
END
C

```



```

SUBROUTINE FOURN(DATA,NN,NDIM,ISIGN)
REAL*8 WR,WI,WPR,WPI,WTEMP,THETA
DIMENSION NN(NDIM),DATA(*)
NTOT=1
DO 11 IDIM=1,NDIM
NTOT=NTOT*NN(IDIM)
11 CONTINUE
NPREV=1
DO 18 IDIM=1,NDIM
N=NN(IDIM)
NREM=NTOT/(N*NPREV)
IP1=2*NPREV
IP2=IP1*N
IP3=IP2*NREM
I2REV=1
DO 14 I2=1,IP2,IP1
IF(I2.LT.I2REV)THEN
DO 13 I1=I2,I2+IP1-2,2
DO 12 I3=I1,IP3,IP2
I3REV=I2REV+I3-I2
TEMPR=DATA(I3)
TEMPI=DATA(I3+1)
DATA(I3)=DATA(I3REV)
DATA(I3+1)=DATA(I3REV+1)
DATA(I3REV)=TEMPR
DATA(I3REV+1)=TEMPI
12 CONTINUE
13 CONTINUE
ENDIF
IBIT=IP2/2
1 IF ((IBIT.GE.IP1).AND.(I2REV.GT.IBIT)) THEN
I2REV=I2REV-IBIT
IBIT=IBIT/2
GO TO 1
ENDIF
I2REV=I2REV+IBIT
14 CONTINUE
IFP1=IP1
2 IF(IFP1.LT.IP2)THEN
IFP2=2*IFP1
THETA=ISIGN*6.28318530717959D0/(IFP2/IP1)
WPR=-2.D0*DSIN(0.5D0*THETA)**2
WPI=DSIN(THETA)
WR=1.D0

```

```

WI=0.D0
DO 17 I3=1,IFP1,IP1
DO 16 I1=I3,I3+IP1-2,2
DO 15 I2=I1,IP3,IFP2
K1=I2
K2=K1+IFP1
TEMPR=SNGL(WR)*DATA(K2)-SNGL(WI)*DATA(K2+1)
TEMPI=SNGL(WR)*DATA(K2+1)+SNGL(WI)*DATA(K2)
DATA(K2)=DATA(K1)-TEMPR
DATA(K2+1)=DATA(K1+1)-TEMPI
DATA(K1)=DATA(K1)+TEMPR
DATA(K1+1)=DATA(K1+1)+TEMPI
15     CONTINUE
16     CONTINUE
WTEMP=WR
WR=WR*WPR-WI*WPI+WR
WI=WI*WPR+WTEMP*WPI+WI
17     CONTINUE
IFP1=IFP2
GO TO 2
ENDIF
NPREV=N*NPREV
18     CONTINUE
RETURN
END
C
SUBROUTINE FREQ(MAXR,FREQ1,FREQ2)
DIMENSION FREQ1(MAXR),FREQ2(MAXR)
INTEGER NR
C
C   INITIALIZE FREQUENCY ARRAY
DO I=1,MAXR
FREQ1(I)=0.
END DO
NR=(MAXR/2)+1
FREQ1(NR)=.5
DO I=1,MAXR
FREQ1(I)=REAL((I-1))/REAL(MAXR)
IF(I .GT. NR) THEN
FREQ1(I)=REAL(((I-1)-MAXR))/REAL(MAXR)
END IF
FREQ2(I)=FREQ1(I)
END DO
RETURN

```

```

END
FUNCTION RAN1(IDUM)
DIMENSION R(97)
PARAMETER (M1=259200,IA1=7141,IC1=54773,RM1=1./M1)
PARAMETER (M2=134456,IA2=8121,IC2=28411,RM2=1./M2)
PARAMETER (M3=243000,IA3=4561,IC3=51349)
DATA IFF /0/
IF (IDUM.LT.0.OR.IFF.EQ.0) THEN
  IFF=1
  IX1=MOD(IC1-IDUM,M1)
  IX1=MOD(IA1*IX1+IC1,M1)
  IX2=MOD(IX1,M2)
  IX1=MOD(IA1*IX1+IC1,M1)
  IX3=MOD(IX1,M3)
  DO 11 J=1,97
  IX1=MOD(IA1*IX1+IC1,M1)
  IX2=MOD(IA2*IX2+IC2,M2)
  R(J)=(FLOAT(IX1)+FLOAT(IX2)*RM2)*RM1
  11 CONTINUE
  IDUM=1
  ENDIF
  IX1=MOD(IA1*IX1+IC1,M1)
  IX2=MOD(IA2*IX2+IC2,M2)
  IX3=MOD(IA3*IX3+IC3,M3)
  J=1+(97*IX3)/M3
  IF(J.GT.97.OR.J.LT.1)PAUSE
  RAN1=R(J)
  R(J)=(FLOAT(IX1)+FLOAT(IX2)*RM2)*RM1
  RETURN
END
C
C   FIND THE WAVEVECTOR COMPONENTS FOR A COLUMN OF
C   DATA(NCOL) IN X DIM, I.E. KX
C
SUBROUTINE SETUPFR(MAXR,KR,NCOL,FREQ1,FREQ2,FFX,FREQX1)
C   JUNE 1 1993
DIMENSION FFX(NCOL),FREQ1(MAXR),FREQ2(MAXR)
REAL FREQX1(KR)
INTEGER KR
DO I=1,KR
  FREQX1(2*I-1)=FREQ1(I)
  FREQX1(2*I)=FREQ2(I)
END DO
C   INITIALIZE WAVEVECTOR(FREQUENCY) ARRAY

```

```

DO I=1,NCOL
FFX(I)=0.
END DO
DO I=1,MAXR
DO J=1,KR
FFX(J+(I-1)*KR)=FREQX1(J)
END DO
END DO
RETURN
END
C
SUBROUTINE SETUPFRY(MAXR,KR,NCOL,FREQ1,FFY)
C   JOUNE 2 1993
DIMENSION FFY(NCOL),FREQ1(MAXR)
INTEGER KR,MAXR,NCOL
DO I=1,NCOL
FFY(I)=0.
END DO
DO I=1,MAXR
DO J=1,KR
FFY(J+(I-1)*KR)=FREQ1(I)
END DO
END DO
RETURN
END
C
SUBROUTINE COMPLEXDAT(NCOL,MAXR,KR,G1,DATA)
REAL DATA(NCOL)
DIMENSION G1(MAXR,KR)
C
INTEGER KR,MAXR,NCOL
DO I=1,MAXR
DO J=1,KR
DATA(J+(I-1)*KR)=G1(I,J)
END DO
END DO
RETURN
END
C
SUBROUTINE REALDAT(NCOL,MAXR,KR,DATA,G1)
REAL DATA(NCOL)
DIMENSION G1(MAXR,KR)
REAL G1
INTEGER KR,MAXR,NCOL

```

```

C
DO I=1,MAXR
DO J=1,KR
G1(I,J)=DATA(J+(I-1)*KR)
G1(I,J)=REAL(G1(I,J)/(MAXR*MAXR))
END DO
END DO
RETURN
END
C
SUBROUTINE OGRID(KR,MAXR,G1,G)
REAL G1,G
DIMENSION G1(MAXR,KR),G(MAXR,MAXR)
INTEGER KR,MAXR
DO I=1,MAXR
DO J=1,MAXR
G(I,J)=G1(I,2*J-1)
END DO
END DO
RETURN
END
C
SUBROUTINE EIGENV(MAXR,LF,PHIXX,PHIYY,PHIXY,LAMBDA1
*,LAMBDA2)
C
DIMENSION PHIXX(MAXR,MAXR),PHIYY(MAXR,MAXR)
DIMENSION PHIXY(MAXR,MAXR)
DIMENSION LAMBDA1(MAXR,MAXR),LAMBDA2(MAXR,MAXR)
REAL A,B,C,D,LF,LAMBDA1,LAMBDA2
INTEGER MAXR
DO I=1,MAXR
DO J=1,MAXR
A=(LF**2)*PHIXX(I,J)+1.0
B=(LF**2)*PHIYY(I,J)+1.0
C=(LF**4)*(PHIXY(I,J)**2)
D=SQRT(((A+B)**2)-4.0*(A*B-C))
LAMBDA1(I,J)=(0.5)*(A+B+D)
LAMBDA2(I,J)=(0.5)*(A+B-D)
END DO
END DO
RETURN
END
C
SUBROUTINE INTENSITY(MAXR,LAMBDA1,LAMBDA2,INTENS)

```

```

DIMENSION LAMBDA1(MAXR,MAXR),LAMBDA2(MAXR,MAXR)
DIMENSION INTENS(MAXR,MAXR)
REAL INTENS,LAMBDA1,LAMBDA2
INTEGER MAXR
DO I=1,MAXR
DO J=1,MAXR
INTENS(I,J)=1./ABS(LAMBDA1(I,J)*LAMBDA2(I,J))
END DO
END DO
RETURN
END
C
SUBROUTINE POSITION(MAXR,NPOINT,LF,PHIX,PHIY,X,Y)
DIMENSION X(NPOINT),Y(NPOINT),PHIX(MAXR,MAXR)
DIMENSION PHIY(MAXR,MAXR)
REAL X,Y,PHIX,PHIY,LF,A,B
INTEGER MAXR,NPOINT
DO I=1,MAXR
DO J=1,MAXR
A=J+(LF**2)*PHIX(I,J)
B=I+(LF**2)*PHIY(I,J)
X(J+(I-1)*MAXR)=A
Y(J+(I-1)*MAXR)=B
END DO
END DO
RETURN
END
C
SUBROUTINE INIT~ PHAETHON(IDUM, A, LF, POWERSPEC,
*CROSSING)
C
INTEGER IDUM
LOGICAL POWERSPEC,CROSSING
REAL A,LF
C
POWERSPEC=.FALSE.
CROSSING=.FALSE.
IDUM=-1
A=3.6
LF=.45
WRITE(*,*) 'main: phaethon12.f,1992-1995'
WRITE(*,*) 'Maria J. Pantazopoulou, Virginia Tech'
RETURN
END

```

B.5 Listing of parameter and include files

```
-10    !the random seed ,IDUM (default -1)
3.6   !A (alpha) the Kolmogorov constant(default 3.6)
0.95  !the fresnel scale, LF= 0.45 at focus(lambda=1, default)
FALSE TRUE !POWERSPEC true , outfile:PSPECTRUM.DAT (K,PS)
TRUE  !CROSSING true,i.e, cell~interpolation module
```

file: phaethon.ini

c parameter and logical variables file.

For the LF parameter:

we have $(\lambda)^2 z = LF^2$ at $\lambda=1$, frequency=318 MHz

c $\lambda/(\lambda=1)=.739$ at frequency=430 MHz

c $\lambda/(\lambda=1)=.525$ at frequency=606 MHz

c $\lambda/(\lambda=1)=.361$ at frequency=880 MHz

c $\lambda/(\lambda=1)=.227$ at frequency=1400 MHz

Relation: $LF=(\text{ratio}) \times LF(\text{focus})$

The focus(LF) for non Kolmogorov values of α is different from .45

to find it run the program focus.f

Maria J. Pantazopoulou, Virginia Tech. 1995

```

C   file : phaeth1.inc
C   Program to calculate wave propagation in the
C   interstellar medium.
C   Declarations for the main routine phaethon12.f
C
INTEGER OUT,MAXR,MAXC,NROW,NCOL,NPOINT,IDUM
C   Where MAXR,MAXC are the dimensions of the grid
C   NROW=2*MAXR,NPOINT=MAXR*MAXC,and NCOL=2*MAXR*MAXC
C   ***** set dimension of the grid here *****
PARAMETER(MAXR=64,MAXC=64)
PARAMETER(NROW=128,NPOINT=4096,NCOL=8192)
C   *****
INTEGER ISIGN,NDIM,KR,NPOINTSIX
INTEGER IXI(NPOINT),IYI(NPOINT),I,J,II
LOGICAL POWERSPEC
LOGICAL CROSSING,FOLD,CAUSTIC
CHARACTER*1 CH
DIMENSION NN(2)
C   *****
C   **** some info and dimensioning of arrays used ****
C   The data array has length 2*MAXR*MAXC
DIMENSION DATA(NCOL)
C   The random matrix(grid): G(MAXR,MAXC)
DIMENSION G(MAXR,MAXC)
C   The random matrix(real and imaginary grid): G1(MAXR,2*MAXC)
DIMENSION G1(MAXR,NROW)
C   The wavevector array: WAVEVEC(MAXR*MAXC)
DIMENSION WAVEVEC(NPOINT)
C   The mixed grid: MIX(2*MAXR,MAXC)
DIMENSION MIX(NROW,MAXC)
C   The real grid array: MIX1(MAXR*MAXC)
DIMENSION MIX1(NPOINT)
C   The imaginary grid array: MIX2(MAXR*MAXC)
DIMENSION MIX2(NPOINT)
C   The freq. or wavevector in kx and ky directions: FREQ(MAXR)
DIMENSION FREQ1(MAXR)
DIMENSION FREQ2(MAXC)
C   The wavevector  $k=\sqrt{kx^2 + ky^2}$  : WAVENO(MAXR,MAXC)
DIMENSION WAVENO(MAXR,MAXC)
C   The Power Spectrum array : PSPECTRUM(MAXR*MAXC)
DIMENSION PSPECTRUM(NPOINT)
C   The wavevector array for logP-logK plot : WAVEVEC2(MAXR*MAXC)
DIMENSION WAVEVEC2(NPOINT)
C   The phase : PHI(2*MAXR*MAXC)

```



```

DIMENSION PHI(NCOL)
C   The wavevector kx component : FFX(NCOL)
DIMENSION FFX(NCOL)
C   The wavevector ky component: FFY(NCOL)
DIMENSION FFY(NCOL)
C   The frequency(wavevector) for xrow:  FREQX1(2*MAXR)
DIMENSION FREQX1(NROW)
C   The second derivatives of the phase:PHIXX(MAXR,MAXC), etc.
DIMENSION PHIXX(MAXR,MAXC),PHIYY(MAXR,MAXC)
DIMENSION PHIXY(MAXR,MAXC)
C   The first derivatives of the phase: PHIY(MAXR,MAXC) ,etc.
DIMENSION PHIY(MAXR,MAXC),PHIX(MAXR,MAXC)
C   The eigenvalues lambda1 and lambda2
DIMENSION LAMBDA1(MAXR,MAXC),LAMBDA2(MAXR,MAXC)
C
REAL MIX,MIX1,MIX2,PHI,FFY,FFX,DATA,PHIXX,PHIYY
REAL PHIXY,PHIY,PHIX,G1,LF,A
REAL PHI1(NPOINT),PHI2(NPOINT)
REAL LAMBDA1,LAMBDA2
C
C   The intensity array: INTENS(MAXR,MAXC),
C   NOTE: not at grid points.
DIMENSION INTENS(MAXR,MAXC)
C
REAL X(NPOINT),Y(NPOINT),Z(NPOINT)
REAL INTENS,DELTA,RAN1,AVE~ INTENSITY

```

B.6 Fortran listing of cell~interpol.f

```
C   Program to calculate intensities on the observers
C   plane, taking into account ray crossings.
program cell~interpol
C   Maria J. Pantazopoulou
SUBROUTINE TETRAGONE(X,Y,NPOINT,MAXR,AVE~INTENSITY)
INTEGER I,J,INEXT,JNEXT,NPOINT,MAXR,MAXC,L,K,M,OUT
INTEGER II,JJ,IZOOM,JZOOM,KZOOM,N2,N
LOGICAL FOLD,CAUSTIC
REAL X(*),Y(*),AVE~INTENSITY(NPOINT)
REAL X1,X2,X3,X4,Y1,Y2,Y3,Y4
REAL X~A,X~B,X~C,X~D,Y~A,Y~B,
*Y~C,Y~D
REAL B~A,B~B,B~C,B~D
REAL TAN~VEC~A,TAN~VEC~B,TAN~VEC~
*C,TAN~VEC~D
REAL CELL~AREA,INTENSITY
REAL FINE~INT~GRID(512,512),FLUX,FLUX0
C   REAL INTENSITY(MAXR,MAXR)
C
C   define the cells with respect to the grid
C   number of cells=MAXR-1 * MAXC-1
C   where MAXR*MAXC the dimensions of the grid.
C
C
MAXC=MAXR
KZOOM=8
IZOOM=512
JZOOM=512
FLUX0=0.
C   *****initialize arrays and counters*****
OUT=4
L=0
K=0
M=1
DO I=1,NPOINT
AVE~INTENSITY(I)=0.0
END DO
C   DO J=1,MAXR
C   DO I=1,MAXC
C   INTENSITY(I,J)=0.0
C   END DO
C   END DO
```

```

DO J=1,JZOOM
DO I=1,IZOOM
FINE~ INT~ GRID(I,J)=0.0
END DO
END DO
C      *****end initialize*****
C
C      set up the cell
C
C      (x4,y4)*—*(x3,y3)
C
C
C      (x1,y1)*—*(x2,y2)
C
C
DO J=0,MAXC-2
DO I=1,MAXR-1
CELL~AREA=0.0
X~A=0.0
Y~A=0.0
X~B=0.0
Y~B=0.0
X~C=0.0
Y~C=0.0
X~D=0.0
Y~D=0.0
X1=0.0
X2=0.0
X3=0.0
X4=0.0
Y1=0.0
Y2=0.0
Y3=0.0
Y4=0.0
INEXT=I+J*MAXR
JNEXT=I+(J+1)*MAXC
C
X1=X(INEXT)
X2=X(INEXT+1)
X3=X(JNEXT+1)
X4=X(JNEXT)
Y1=Y(INEXT)
Y2=Y(INEXT+1)
Y3=Y(JNEXT+1)

```

```

Y4=Y(JNEXT)
C   write out for debugging
C   WRITE(OUT,*) 'X1:',X1,' X2:',X2,' X3:',X3,' X4:',X4
C   WRITE(OUT,*) 'Y1:',Y1,' Y2:',Y2,' Y3:',Y3,' Y4:',Y4
C   calculate vectors that define tetragon.
X~A=X2-X1
X~B=X4-X1
Y~A=Y2-Y1
Y~B=Y4-Y1
X~C=X2-X3
X~D=X4-X3
Y~C=Y2-Y3
Y~D=Y4-Y3
C   find out if rays cross,ie are we on a caustic surface?
TAN~VEC~A=Y~A/(X~A)
TAN~VEC~B=Y~B/(X~B)
TAN~VEC~C=Y~C/(X~C)
TAN~VEC~D=Y~D/(X~D)
C   find y intercepts for the vectors
B~A=Y1-TAN~VEC~A*X1
B~B=Y1-TAN~VEC~B*X1
B~C=Y3-TAN~VEC~C*X3
B~D=Y3-TAN~VEC~D*X3
CALL TWIST(X1,X2,Y1,Y4,B~B,B~C,TAN~VEC~
*B,TAN~VEC~C,CAUSTIC)
IF(CAUSTIC)THEN
L=L+1
ELSE
CALL TWIST(X1,X2,Y1,Y4,B~A,B~D,TAN~VEC~
*A,TAN~VEC~D,CAUSTIC)
IF(CAUSTIC) THEN
L=L+1
END IF
END IF
C   IF(CAUSTIC)THEN
C   CELL~AREA=(1./2.)*ABS(X~A*Y~B-X~B*Y~
*A+X~C*Y~D-X~D*Y~C)
C   ELSE
CELL~AREA=(1./2.)*(ABS(X~A*Y~B-X~B*Y~
*A)+ABS(X~C*Y~D-X~D*Y~C))
C   END IF
C   WRITE(OUT,*) 'cell:',M,' cell area:', CELL~AREA
IF (CELL~AREA .EQ. 0.0) THEN
AVE~INTENSITY(M)=(M-(M-1))*NPOINT

```

```

ELSE
AVE~INTENSITY(M)=REAL(M-(M-1))/(ABS(CELL~AREA))
END IF
FLUX0=FLUX0+AVE~INTENSITY(M)*(ABS(CELL~AREA))
CALL AREAVECTOR(X1,X2,X3,X4,Y1,Y2,Y3,Y4,CELL~AREA,FOLD)
IF (FOLD)THEN
K=K+1
END IF
C
C   WRITE(OUT,*) 'caustic=',CAUSTIC,' fold=',FOLD
C   WRITE(OUT,*) '**'
C   try to find which of the grid points are falling
C   inside each cell and assign an intensity value to them.
DO JJ=1,MAXR
DO II=1,MAXC
IF((II .GE. X1) .AND. (II .LE. X2) .AND.
* (JJ .GE. Y1) .AND. (JJ .LE. Y4))THEN
INTENSITY=AVE~INTENSITY(M)
WRITE(8,*) II,JJ,INTENSITY
END IF
END DO
END DO
CALL ZOOM(X1,X2,X3,X4,Y1,Y2,Y3,Y4,IZOOM,JZOOM,
* KZOOM,AVE~ INTENSITY(M),FINE~ INT~ GRID)
C
M=M+1
END DO
END DO
C
WRITE(*,*) 'Total initial flux:',FLUX0
FLUX=0.
C   write out for debugging
c   DO I=1,M-1
c   WRITE(OUT,*) ' ofcells:',I,' Ave~intens/cell:'
*,AVE~INTENSITY(I)
c   END DO
c   WRITE(OUT,*) 'caustic: of twists:',L,' folds:',K
DO J=1,JZOOM
DO I=1,IZOOM
WRITE(9,*) I,J,FINE~ INT~ GRID(I,J)
OPEN(39,FILE='FINTENSITY.DAT')
WRITE(39,*) FINE~ INT~GRID(I,J)
FLUX=FLUX+FINE~ INT~GRID(I,J)
END DO

```

```

END DO
CLOSE(39)
WRITE(*,*) 'Flux calculated:',FLUX
WRITE(*,*) 'Flux calculated/Flux expected:',FLUX/(IZOOM**2)
RETURN
END
C
SUBROUTINE AREAVECTOR(X1,X2,X3,X4,Y1,Y2,Y3,Y4,CCELL~
AREA,FOLD)
C    find direction of the area vector, to determine folds.
LOGICAL FOLD
REAL CCELL~AREA, AREA~VECTOR,X1,X2,X3,X4
C
IF(CCELL~AREA .EQ. 0.0)THEN
AREA~VECTOR=1.0
ELSE
AREA~VECTOR=CCELL~AREA/ABS(CCELL~AREA)
END IF
C    IF(AREA~VECTOR .EQ. -1.)THEN
IF(((X2 .LE. X1) .AND. (X3 .LE. X4)) .OR. ((Y4 .LE. Y1)
* .AND. (Y3 .LE. Y2)))THEN
FOLD=.TRUE.
ELSE
FOLD=.FALSE.
END IF
RETURN
END
C
C
SUBROUTINE TWIST(AX1,AX2,BY1,BY4,BA,BC,TANA,TANC,CAUSTIC)
C
C    solve 2 linear equations simultaneously,to find the
C    point that the vectors A and C ,or the vectors B and D
C    intercept inside the tetragon, i.e if we have a twist.
C    twist will be any combination of the kind:
C
C
C    (x4,x3)*-----*(x3,y3)
C
C      \  /
C       \ /
C        ●(x,y) common point
C       / \
C      /  \
C    (x2,y2)*-----*(x1,y1)
C
C    and 3 more cyclic combinations
C

```

```

LOGICAL CAUSTIC
REAL BA,BC,TANA,TANC,X,Y,AX1,AX2,BY1,BY4
C   solve eqns of the form:tanA*x+bA=y,and tanC*x+bC=y,for (x,y)
IF((TANA-TANC) .EQ. 0.0)THEN
X=0.
ELSE
X=(BC-BA)/(TANA-TANC)
END IF
Y=TANC*X+BC
IF((AX1 .GE. AX2) .AND. (BY1 .LE.BY4))THEN
IF((X .LT. AX1) .AND. (X .GT. AX2) .AND.
* (Y .GT. BY1) .AND. (Y .LT. BY4))THEN
CAUSTIC=.TRUE.
ELSE
CAUSTIC=.FALSE.
END IF
END IF
IF((AX1 .LE. AX2) .AND. (BY1 .LE. BY4))THEN
IF((X .GT. AX1) .AND. (X .LT. AX2) .AND.
* (Y .GT. BY1) .AND. (Y .LT. BY4))THEN
CAUSTIC=.TRUE.
ELSE
CAUSTIC=.FALSE.
END IF
END IF
IF((AX1 .LE. AX2) .AND. (BY1 .GE. BY4))THEN
IF((X .GT. AX1) .AND. (X .LT. AX2) .AND.
* (Y .GT. BY4) .AND. (Y .LT. BY1))THEN
CAUSTIC=.TRUE.
ELSE
CAUSTIC=.FALSE.
END IF
END IF
WRITE(4,*) 'subroutine twist: "x and y intercept,    x1
*   x2'
WRITE(4,*) '          ',X,Y,AX1,AX2
RETURN
END
C
SUBROUTINE ZOOM(X1,X2,X3,X4,Y1,Y2,Y3,Y4,IZOOM,JZOOM,
* KZOOM,AVEINTENSITY,FINE~INT~GRID)
INTEGER IZOOM,JZOOM,I,J,KZOOM
REAL AVEINTENSITY,FINE~INT~GRID(IZOOM,JZOOM)
REAL X1,X2,X3,X4,Y1,Y2,Y3,Y4,FR,FR1

```

```

REAL FINEX,FINEY
C
DO J=1,JZOOM
DO I=1,IZOOM
FR=REAL (I)/REAL(KZOOM)
FR1=REAL (J)/REAL(KZOOM)
FINEX=FR
FINEY=FR1
IF((FINEX .GE. X1) .AND. (FINEX .LE. X2) .AND.
* (FINEY .GE. Y1) .AND. (FINEY .LE. Y4))THEN
FINE~INT~GRID(I,J)=AVEINTENSITY+FINE~INT~
GRID(I,J)
ELSE
FINE~INT~GRID(I,J)=FINE~INT~GRID(I,J)
END IF
END DO
END DO
RETURN
END
C
C
SUBROUTINE INIT~ CELL~INTERPOL(FOLD,CAUSTIC)
LOGICAL FOLD,CAUSTIC
WRITE(*,*) 'interpolation module:'
WRITE(*,*) 'cell~ interpol.f, April 1995'
WRITE(*,*) 'Maria J. Pantazopoulou, Virginia Tech'
FOLD=.FALSE.
CAUSTIC=.FALSE.
RETURN
END
C

```


vitae:

Born in Astros, Greece.

Ph.D Physics (Radio Astronomy), Virginia Tech, Blacksburg Virginia, August 1996;
M.S. Mathematics, May 1991, Tennessee Technological University, Cookeville,
Tennessee:

M.A. Physics, Kent State University, Kent, Ohio ,1982:

B.S. Physics, Patras University, Patras, Greece, 1980.

- Summer of 1995, I worked on the implementation of a scattering code, to include multiple scattering, simulating light scattering in planetary rings, at the Lunar and Planetary Laboratory at the University of Arizona, Tucson.
 - Converted a multi-thousand line computer code (magnetospheric-field model calculations of cosmic-ray particles' cut-off rigidities and asymptotic directions) from Pascal to Fortran, for the Naval Research Laboratory, Summer 1994
 - Laboratory instructor, Dept. of Physics, Hollins College, 1991-1992
 - Summer 1984 on a fellowship from the International Union of Pure and Applied Biophysics to attend the 8th International Biophysics Congress, Bristol, England .
- Publications and contributions
- 'Applications of Topology in the study of Defects in Condensed Matter Physics' M .J. Pantazopoulou, M. S. thesis in Mathematics, Tennessee Technological University, May 1991.
 - 'Light Fragment Yield Ratios from Heavy Ion-Induced Fragmentation in Atmospheric Collisions of Cosmic-Ray Primary Nuclei', M. J. Pantazopoulou, A. F. Barghouty, and R. A. Witt, *Canadian Journal of Physics* 69, 1481 (1991)
 - 'Effect of Multiple Scattering on Transition Radiation from a Dynamic Interface', A. F. Barghouty and M. J. Pantazopoulou, *Physical Review A* 44, 3083 (1991).
 - 'Physical Studies of Oxidized Cholesterol phosphatidylcholine mixtures', M. J. Pan-

tazopoulou, L. Lis, International Chemical Congress of Pacific Basin Societies, Honolulu, Hawaii, (December 1984).

- 'Interactions of lipids with cholesterol compounds' , M.J. Pantazopoulou, L. Lis, 8th International Biophysics Congress, Bristol, England, summer 1984.
- ' Structure and phase separation of lipids and cholesterol mixtures' , L. Lis, M.J. Pantazopoulou, Ohio Section meeting of the American Physical Society, Kent, Ohio (April, 1984).

Urania J. Pantazopoulou