Fixed Optics Fourier Transform Spectrometers with Photodetector-Arrays

A thesis for the degree of

Master of Science

Submitted to

Dublin City University

by

Dirk Christian Kalisch B.Sc.

Research Supervisor

Prof. Martin Henry
School of Physical Sciences
Dublin City University

October 1997

Declaration

I hereby certify that this material, which I know submit for assessment on the programme of study leading to the award of Master of Science is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work

Signed _____ Date _____

Dirk Kalisch

Table of Contents

Acknowledgements

First, I would like to thank Dr Martin Henry, my supervisor, for his advise and encouragement

I would like to thank my fellow students and friends Shane, Terri, Pat, Conor, Steve, Kate and the other students in the department for affording me a deeper (if occasionally blurred) insight into the Irish culture

Thanks also to Enda, Siobhán, John Costello, Des, Al and Alan for their indespensible advise and assistance

Abstract

Two different types of stationary Fourier transform spectrometer, one of them based on a triangular interferometer and the second one based on a Savart-plate interferometer, were built and tested The theoretical background that underlies the operation of fixed optics spectrometers is given as well as a description of the mathematical routines used for the reconstruction of the spectrum from the interferograms produced by a particular light source Finally, the spectra of various light sources were measured and the performance limits and accuracy of the two instruments were established

Chapter 1

Introduction

1.1 Detector-Arrays in Spectroscopy

Spectrometers are widely used in many areas of manufacturing, research and development They have widespread applications ranging from process analysis and quality control to fundamental scientific research Sensitivity to vibrations and shocks has always been a constraining feature of traditional spectrometer design This may result in repeated requirements of adjustment and calibration The fundamental reason for this sensitivity lies in the use of some scanning mechanism whereby spectral information is obtained The requirement of robustness, reproductivity and accuracy impose contrasting demands on the instrument design, so that a measure of compromise is always involved The ability to dispense with moving parts can play a major role in avoiding such compromises Sometimes, size is another design criterion, especially for instruments made for field use, where small, handheld devices are desirable

Detector arrays have been used in spectroscopy for more than a decade now Since their introduction they have provided a major boost to the development of many new products in this area They more and more supersede photographic films or photomultiplier tubes Photographic films, which have been an adequate means to record a spectrogram since the invention of spectroscopy itself, provide a high sensitivity over a wide spectral range due to the possibility of long exposure times Modern holographic films are capable of recording more than 5000 lines per millimetre, giving a very high resolution The main disadvantages are the necessity of wet film processing leading to a high time expense and difficulties with intensity calibration Also, time resolved spectroscopy becomes impossible, especially if long exposure times are required

Photomultiplier tubes provide real-time signal detection, and they can be designed to have imaging capabilities also Those products, however, are limited in the spectral ranges over which they operate Therefore, photomultiplier tubes are mostly used to detect the intensity of radiation at only one fixed point in space To record a full spatial distribution of intensities, it is necessary that the spectrometer has at least one moving part allowing it to scan a full spectrum This is implemented for example in a Czerny - Turner type spectrometer, in which an echelle grating can be rotated around its vertical axis The very high sensitivity of

PM tubes is achieved at the expense of a reduced signal to noise ratio due to quantum effects inside the tube.

Photodiode- or CCD arrays can overcome the problems mentioned above. Array detectors possess a very high sensitivity for a given wavelength range, if the right semiconducting material is chosen (they can even detect single photons but the quantum efficiency is smaller than 100%). The diagram below shows typical relative sensitivities of various semiconductor detectors over their spectral operating range.

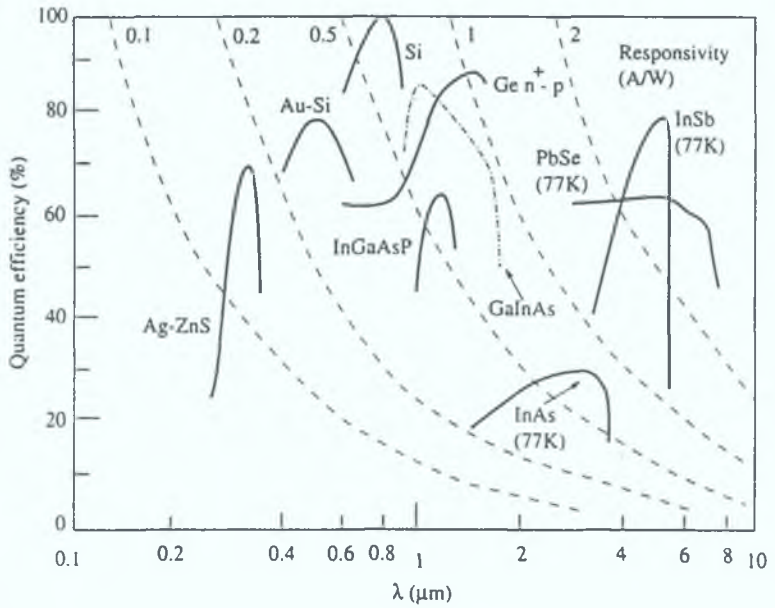


Fig. 1.1 Responsivity and quantum efficiency for various semiconducting matials. (Adapted from P.Bhattacharya, *Semiconductor Optoelectronic Devices*, Prentice Hall)

Silicon photodiode arrays are available both as a one dimensional string with up to approx. 10000 elements and as two dimensional arrays with 2048*2048 elements, each of which is typically 25 microns. The associated low noise electronics are capable of reading out and digitising the signal from the entire array in several milliseconds, making limited time resolved observations possible. Of those materials shown in figure 2.1 silicon is most commonly used for the making of photodiode arrays. Less common but also available are array detectors based on Ge, InCdAs, HgCdTe and PbS/PbSi.

The use of detector arrays in conjunction with dispersion grating spectrometers is quite common and many instruments of this type have already been commercialised. They are usually called Multichannel Dispersive Spectrometers and feature no grating scanning. Depending on the grating used, they allow the instantaneous measurement of spectra either over a short spectral window at high resolution or if a broad spectrum is measured the resolution decreases.

Detector arrays have been found to allow for the elimination of moving parts in Fourier Transform Spectrometers (FTS), also This idea was originally proposed by Okamoto, Kawata and Minami (1984) The researchers used various types of stationary interferometers to produce an interferogram at a certain point in space There, it is picked up by the detector and the spectrum is reconstructed by performing the Fourier transform on the digitised output As in conventional FTS, the resolution mainly depends on the maximum path difference that can be introduced between the two interfering beams In practise, it means that the resolution is linearly proportional to the number of pixels in the detector, which is too small for many scientific applications However, if one is only interested in the main spectral features of a light source rather than its fine structures, a stationary Fourier transform spectrometer may be employed

In the following chapter various kinds of spectrometers using array detectors are described and some of their advantages and disadvantages are pointed out A distinction is drawn between the use of dispersion and interferometric optical systems The optical prism, the oldest dispering element, shall not be dealt with since it has been replaced largely by gratings and is normally found in spectrometers as a wavelength preselection device to prevent order overlap Furthermore, methods for resolution enhancement and dynamic range improvement are outlined also Finally, two different stationary Fourier transform spectrometers were designed and built and the results are discussed in the last section

Chapter 2

Theory

2.1 Introduction

In this chapter the theoretical backgrounds of both wavelength dispersion instruments and Fourier transform spectrometers are described. Here, the emphasis is laid on the application of both techniques and is not intended to be a complete reference to e.g. the optics of a diffraction grating. Instead, this chapter should provide sufficient information to design real spectrometers using 'off the shelf ' optical components. Some practical concepts such as resolution and optical throughput are also explained, since these are important issues for the experimental work described later.

2.2 Dispersion instruments

The Czerny-Turner configuration is most commonly used in grating spectrometers without multichannel detectors. It comprises two concave mirrors and a plane diffraction grating (Fig. 2.1)

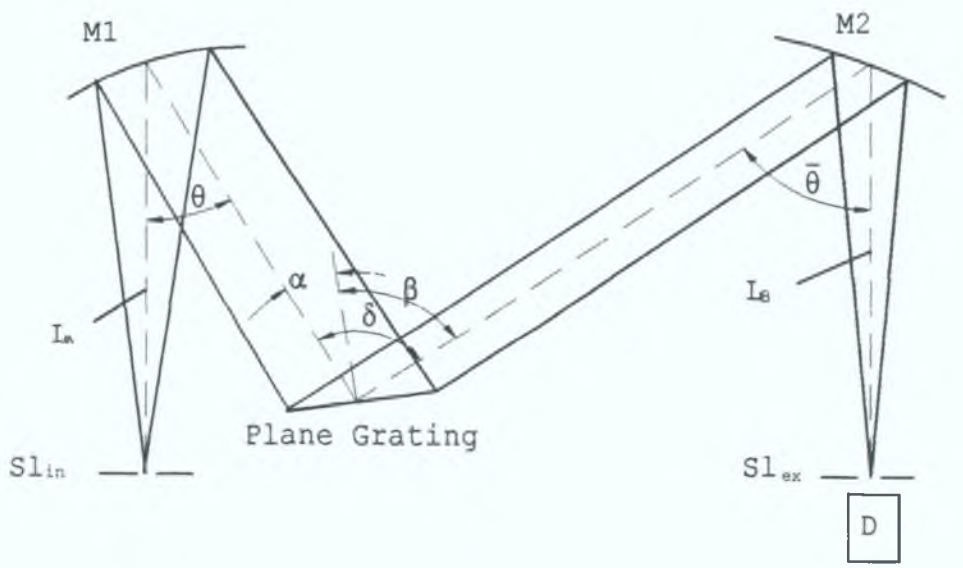


Figure 2.1 Czerny-Turner Configuration

The basic equations which underlie the operation of those spectrometers are provided in many text books, e.g. "The Principles of Optics" (Born, M. and Wolf, E., 1975). The following equations were taken and summarized from "The Optics

of Spectroscopy" (Lerner, J M and Thevenon, A , 1988), a tutorial which provides the designer of a spectroscopic system with a reference to the essential equations

The basic diffraction grating equation is

$$\sin\alpha + \sin\beta = k n \lambda \qquad (2\ 1)$$

where α = angle of incidence,
β = angle of diffraction,
k = order of diffraction,
n = groove density in grooves per unit length

The plane grating is rotatable and the light source can be scanned in wavelength If one differentiates and rearranges equation 2 1 with respect to β one obtains the angular dispersion

$$\frac{d\beta}{d\lambda} = \frac{k n}{\cos\beta} \qquad (2\ 2)$$

More useful in most applications is the linear dispersion, which defines the extent to which a spectral interval is spread out in the focal field of the spectrometer It is usually expressed in nm/mm and given by

$$\frac{\Delta\lambda}{\Delta x} = \frac{\cos\beta}{k n L_B} \qquad (2\ 3)$$

where L_B is the effective exit focal length and Δx is the distance between two lines having a spectral separation $\Delta\lambda$ The resolving power of a diffraction limited ideal grating instrument is given by

$$R = \frac{\lambda}{\Delta\lambda} = k n W_g = k N \qquad (2\ 4)$$

where λ = the central wavelength of the spectral line to be resolved,
W_g = the illuminated width of the grating,
N = the total number of grooves on the grating

In order to determine the actual instrumental resolving power a spectrum is recorded of a sufficiently monochromatic line source (e g laser or low pressure discharge lamp) The full width at half maximum (FWHM) is defined as the bandpass of the system The shape of the instrumental line profile is related to the convolution of various functions each representing one of the following parameters

- the width of the entrance slit,
- the width of the exit slit or of one pixel in the case of a multichannel detector,
- diffraction phenomena,
- aberrations,
- the quality of the system's components and alignment

Whereas most of these factors are always present in most spectrometers, recent advances in holographic grating technology now permits complete correction of all optical aberrations in a diffraction grating spectrometer

Before linear array detectors became commercially available, photographic films were commonly used to record the spectrum in a grating spectrometer Usually, these films were flexible and could be bent to a certain extent This allowed the design of instruments that dispensed with focusing lenses and mirrors but relied on concave gratings in order to focus and disperse the light The Paschen-Runge design (fig 2 2) makes use of the Rowland circle Here, the grooves on the concave grating are equally spaced to a plane projection of the surface

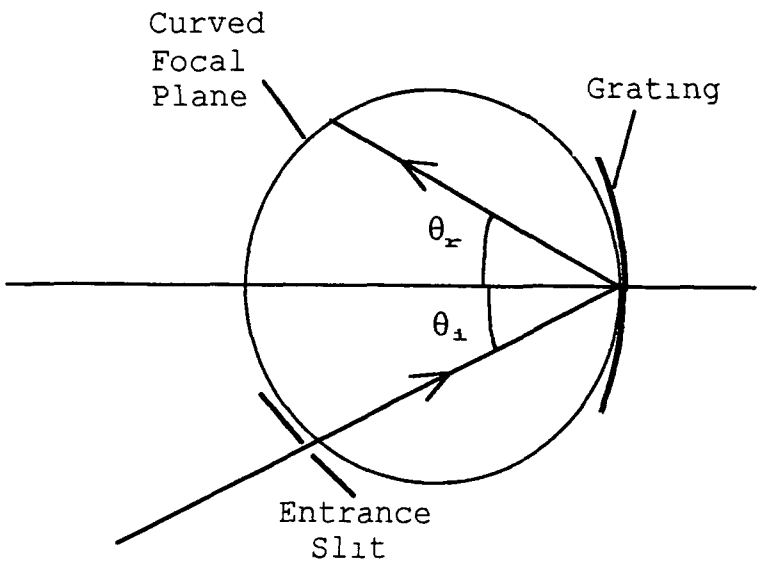


Fig. 2.2 Rowland Circle

The grating is tangent at its centre with the Rowland circle having a diameter equal to the radius of curvature of the grating A slit source placed anywhere on the circle gives well focused spectral lines that fall also on the circle Although curved array detectors have become available, most modern spectrographs employ rigid linear detector arrays The user should be aware that the focal plane may be tilted by a small angle γ (see Fig 2 3) and hence the pixel position normally occupied by the exit slit may not mark the normal to the focal plane Furthermore, the dispersion and image magnification may vary over the focal plane Therefore, the number of pixels per bandpass may vary not only across the focal plane, but will also depend on the wavelength coverage

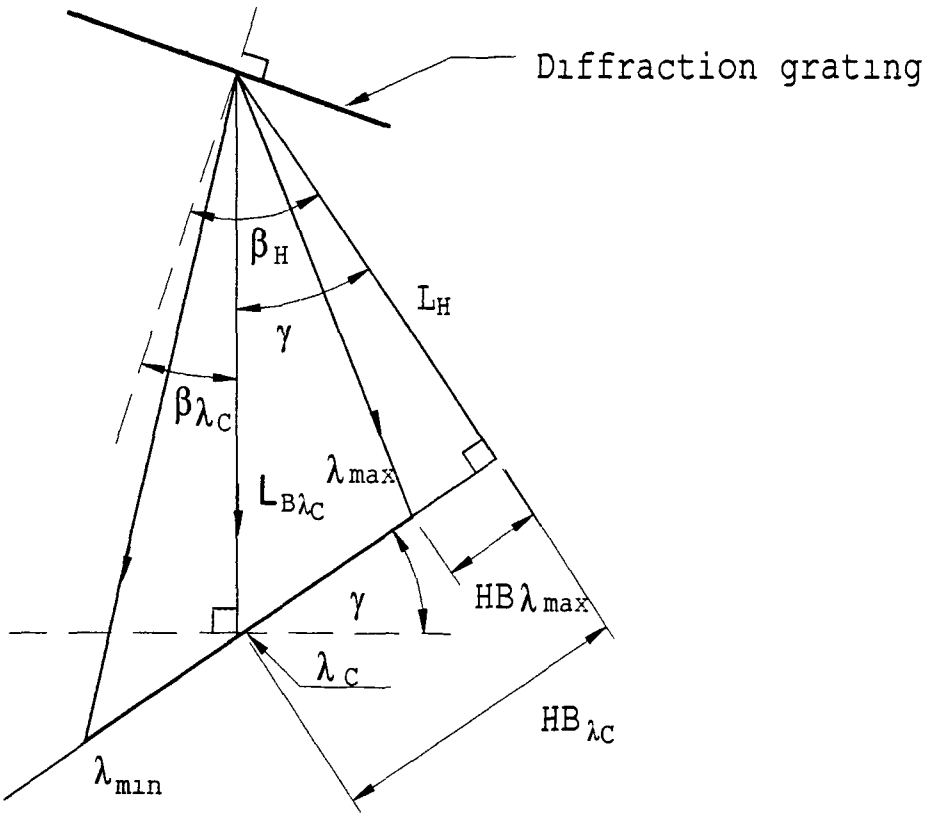


Fig. 2.3 Spectrograph with Focal Plane inclined at an angle γ

If the tilt of the focal plane in a system is unknown, it may be necessary for the user to deduce the value of γ This is most easily achieved by taking a well known spectrum and iteratively substituting incremental values of $\pm\gamma$ until the wavelength appearing at each pixel corresponds to the calculated value

The following formulae can be used to calculate the angle of diffraction in a Czerny Turner type configuration, when γ is known

$$L_H = F\cos\gamma \ (F = L_{B\lambda_c}) \quad (2\,5)$$

$$\beta_H = \beta_{\lambda_c} + \gamma \quad (2\,6)$$

$$HB_{\lambda_c} = F\sin\gamma \quad (2\,7)$$

$$HB_{\lambda_N} = P_W(P_\lambda - P_C) + HB_{\lambda_c} \quad (2\,8)$$

where P_W is the pixel width, P_λ and P_c are the pixel numbers at λ_n (wavelength n) and λ_c (wavelength at centre of array)

$$\beta_{\lambda_N} = \beta_H - \arctan\left(\frac{HB_{\lambda_N}}{L_H}\right) \quad (2\,9)$$

The corrected value β_{λ_N} can now be inserted into equation 2 1

$$\lambda_N = \frac{\sin\alpha + \sin\beta}{kn} \quad (2\,10)$$

2.2.1 Etendue

Etendue is a measure for the capacity of an optical system to transmit energy and is determined by the least optimised segment (the segment with the smallest etendue) of the whole system For maximum efficiency and a minimum of stray-light inside the spectrometer, the etendue should be matched for each of its parts The etendue of the entrance optics of a grating instrument is defined as

$$E = A_S\ \Omega_C \quad (2\,11)$$

where A_S is the area of the source and Ω_C is the solid angle subtended at the imaging lens L_1

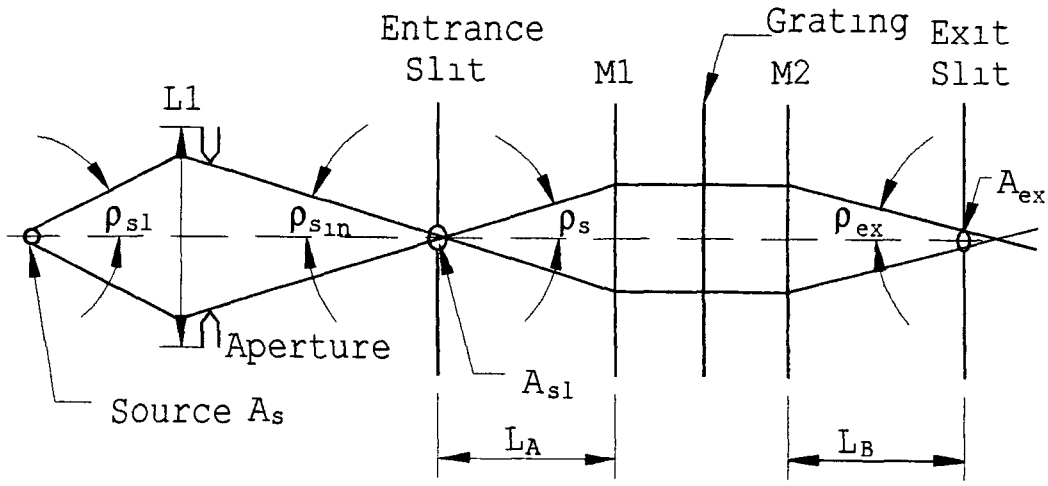


Fig. 2.4 Typical Grating Monochromator

Etendue will be optimised if

$$E = A_S\ \Omega_{Sl} = A_{Sl}\ \Omega_{Sin} = A_{Sl}\ \Omega_{Sout} = A_{ex}\ \Omega_{ex} \qquad (2\ 12)$$

or, if the set-up is symmetrical about the optical axis

$$E = A_S\ \pi\ \sin^2 \rho_{Sc} = A_{Sl}\ \pi\ \sin^2 \rho_{Sin} = \qquad (2\ 13)$$

The area of the entrance slit can also be expressed in terms of bandpass and other parameters of the system

$$A_{Sl} = h\ \frac{k\ n\ F\ BP}{\cos\alpha} \qquad (2\ 14)$$

where h is the slit height Now, the etendue of the entrance optics can be calculated using the following formula for the solid angle subtended by the slit

$$\Omega_{Sout} = \frac{\text{effective width of the grating}}{F^2} = \frac{A_{grating}\ \cos\alpha}{F^2} \qquad (2\ 15)$$

and

$$E = \frac{h\ n\ k\ A_{grating}}{F} \qquad (2\ 16)$$

The ratio h/F implies that etendue may be increased by enlarging the height of the entrance slit and by decreasing the focal length of the imaging system In practise, however, these measures will increase stray light and may also reduce resolution from an increase in system aberration

2.3 Fourier-Transform Spectrometers (FTS)

As Jacqumot pointed out in one of his early papers (1954), the energy throughput of a spectrometer is considerably higher if the system is symmetrical about the optical axis, so no entrance or exit slits have to be used In particular, he compares the Fabry-Perot spectrometer with a grating instrument with the same resolving power and effective area and reports that a gain in flux between 30 and 400 can be achieved by using the etalon Basically, the same applies to Fourier transform spectrometers, where the higher energy throughput leads in principle to a

more precise measurement of intensity under noise limited conditions It also allows in absorption spectroscopy a wide choice of sample thickness where grating spectrometers often impose limitations Although Jacquinot's considerations do not strictly apply to stationary FTS (the etendue is limited by the pixel size of the detector array and the divergence of the exiting light beam) the optical throughput can still be much higher than that of a grating spectrometer

The other main advantage of FTS, however, the multiplex or Fellgett advantage, is fully applicable to fixed optics FTS, hence resulting in an improved signal-to-noise ratio For a classical spectrometer the signal-to-noise ratio is given by (Mertz, 1965)

$$SNR_{SC}(\upsilon)=f(\upsilon)\Delta\upsilon\frac{E_{SC}}{a}t^{\frac{1}{2}}\left(\frac{\upsilon_{max}-\upsilon_{min}}{\Delta\upsilon}\right)^{-\frac{1}{2}} \qquad (2\ 17)$$

where $f(\upsilon)$ is the spectral density function, $\Delta\upsilon$ is the resolution, E_{SC} is the spectrometer efficiency, a is the detector noise, t is the total time available for measurement and υ_{min} and υ_{max} are the wavenumber scan limits The term in brackets on the right hand side could also be written as N, since it reflects the fact that only 1/N of the total time is available for looking at each resolution element In FTS, all the resolution elements are being observed at the same time but the noise also increases as $t^{\frac{1}{2}}$, whence one can calculate the SNR of FT-spectrometry as

$$SNR_{FT}=f(\upsilon)\Delta\upsilon\, t^{\frac{1}{2}}\left(\frac{E_{FT}}{a}\right) \qquad (2\ 18)$$

making the multiplex or Fellgett advantage

$$F_g(\upsilon)=\frac{SNR_{FT}(\upsilon)}{SNR_{SC}(\upsilon)}=\left(\frac{E_{FT}}{E_{SC}}\right)N^{\frac{1}{2}} \qquad (2\ 19)$$

Hence the signal-to-noise ratio increases with the square root of the number of sampling points

2.3.1 Some Theoretical Aspects of Fourier Transform Spectroscopy

A conventional FTS usually comprises a two beam interferometer such as a Michelson or Twyman and Green interferometer with a fixed and a movable mirror, the latter of which can be translated along the axis perpendicular to its reflecting surface A physical detector such as a photocell or a photomultiplier tube detects

the intensity of the centre fringe of the fringe pattern formed. The intensity at the detector for a given optical path difference x in the interferometer can be calculated by

$$I(x)=\int_0^\infty B(\sigma)\,d\sigma+\int_0^\infty B(\sigma)\cos(2\pi\sigma x)\,d\sigma \qquad (2.20)$$

where $B(\sigma)$ is the spectral power of the light of wavenumber σ after subtraction of the various losses within the interferometer (absorption, reflection, etc.). The first term on the right does not depend on x and can be considered as background radiation. The second term is the Fourier transform of the spectral distribution of the source. Inverse transformation yields

$$B(\sigma)=\int_0^{x_{max}} I(x)\cos(2\pi\sigma x)dx \qquad (2.21)$$

Therefore, if the intensity distribution is recorded for a range of optical path differences, the spectrum of the source can be reconstructed.

From its definition (2.20) I(x) is perfectly symmetrical about x=0 but instrumental imperfections will lead to a slightly asymmetric interferogram. Although this will usually be only a small effect, an algorithm for its correction first established by Forman, Steel and Vanasse (1966) will be introduced later. Furthermore, a finite maximum path difference between the two interfering beams and a discrete sampling of the interferogram will also cause the interferometer output to differ from the ideal case. In the following, however, the perfect case is considered first and then account will be given on the main parameters that cause the spectrum measured to be different from the actual one. It should be pointed out here, that the following theorems and calculations are taken from 'The Principles of Interferometric Spectroscopy' (Chamberlain, 1979) which is a very comprehensive source of information to the FT spectroscopist.

If one multiplies the intensity of a light source at a given wavenumber by the area A illuminated by the source, one obtains the power spectrum $L(\sigma)$

$$L(\sigma)=\int_A I(x,y,\sigma)dA \qquad (2.22)$$

In a Michelson interferometer the detected spectral power is the product of the spectral power $L(\sigma)$ and an overall transmission factor $\tau_o(\sigma)$, which is mainly a function of the reflection and transmission factors of the beamsplitter.

$$B(\sigma)=\frac{1}{2}L(\sigma)\tau_0(\sigma) \tag{2.23}$$

A general real function that is neither even or odd can be expressed as the sum of an even and an odd function. The detected power spectrum $B(\sigma)$ exists only for values of σ greater than zero and can be resolved into odd and even components.

$$B(\sigma)=B_e(\sigma)+B_o(\sigma) \tag{2.24}$$

The time variant part of the integral 2.21 can therefore be rewritten as

$$F_e(x)=\int_{-\infty}^{\infty}\left(B_e(\sigma)+B_o(\sigma)\right)\cos(2\pi\sigma x)d\sigma$$

$$=2\int_0^{\infty}B_e(\sigma)\cos(2\pi\sigma x)\,d\sigma \tag{2.25}$$

The inverse Fourier transform of 2.25 yields

$$B_e(\sigma)=2\int_0^{\infty}F_e(x)\cos(2\pi\sigma x)\,dx \tag{2.26}$$

and since

$$B(\sigma)=2B_e(\sigma) \qquad (\text{for } \sigma>0) \tag{2.27}$$

one can write 2.26 as

$$B(\sigma)=4\int_0^{\infty}F_e(x)\cos(2\pi\sigma x)dx \tag{2.28}$$

The Fourier integral 2.28 covers the entire semi infinite range $0\le x\le\infty$. In practise, however, the interferogram can only be observed over a range $0\le x\le D$. If the light source is strictly monochromatic the recorded interferogram will be

$$F_e(x)=2\xi_e(\sigma)\cos(2\pi\sigma x) \tag{2.29}$$

where $\xi_e(\sigma)$ is the recorded power of each partial beam in the interferometer. Fourier transformation of 2.29 with the boundaries 0 and D yields

$$B_e^c(\sigma) = 4\int_0^D \xi_e(\sigma_0)\cos(2\pi\sigma_0 x)\cos(2\pi\sigma x)\,dx \qquad (2\ 30)$$

$$= 2D\xi_e(\sigma)\left[\mathrm{sinc}\,2(\sigma_0+\sigma)D + \mathrm{sinc}\,2(\sigma_0-\sigma)D\right] \qquad (2\ 31)$$

One can see that the calculated spectrum has two broad features with a full width at half maximum depending on the maximum optical path difference D They are accompanied by two side lobes on either side of each of them and can easily be mistaken for spectral features A technique called weighting has been introduced to mitigate the effect of these side lobes This is done by multiplying the interferogram by a slowly falling weighting function $W\left(\frac{x}{D}\right)$ The simplest weighting function is

$$W\left(\frac{x}{D}\right) = 1 - \left|\frac{x}{D}\right| \qquad (2\ 32)$$

which is basically a straight line with values ranging from unity at the origin to zero at x=D The Fourier integral 2 28 then becomes

$$B_e^c(\sigma) = 2\int_0^D F_e(x)\,W\left(\frac{x}{D}\right)\cos(2\pi\sigma x)\,dx \qquad (2\ 33)$$

$$= 2\int_0^\infty F_e(x)\,W\left(\frac{x}{D}\right)\Pi\left(\frac{x}{D}\right)\cos(2\pi\sigma x)\,dx \qquad (2\ 34)$$

where $\Pi\left(\frac{x}{D}\right)$ denotes a function with the cut-off at x=0 and x=D and is defined as

$$\Pi\left(\frac{x}{D}\right) = \begin{cases} 1 & 0 \le x \le D \\ 0 & x<0, x>D \end{cases} \qquad (2\ 35)$$

Application of the convolution theorem permits equation 2 34 to be rewritten as

$$B_e^c(\sigma) = B_e(\sigma) * A(\sigma D) \qquad (2\ 36)$$

where

$$A(\sigma D) = 2\int_0^\infty W\left(\frac{x}{D}\right)\Pi\left(\frac{x}{D}\right)\cos(2\pi\sigma x)\,dx \qquad (2\ 37)$$

which is callled the apodized spectral window The diagram below (fig 2 4) shows how various weighting functions affect the spectral window and how effectively the side lobes have been removed One can also see that apodization increases the width of the central peak and hence diminishes the resolution of the spectrometer

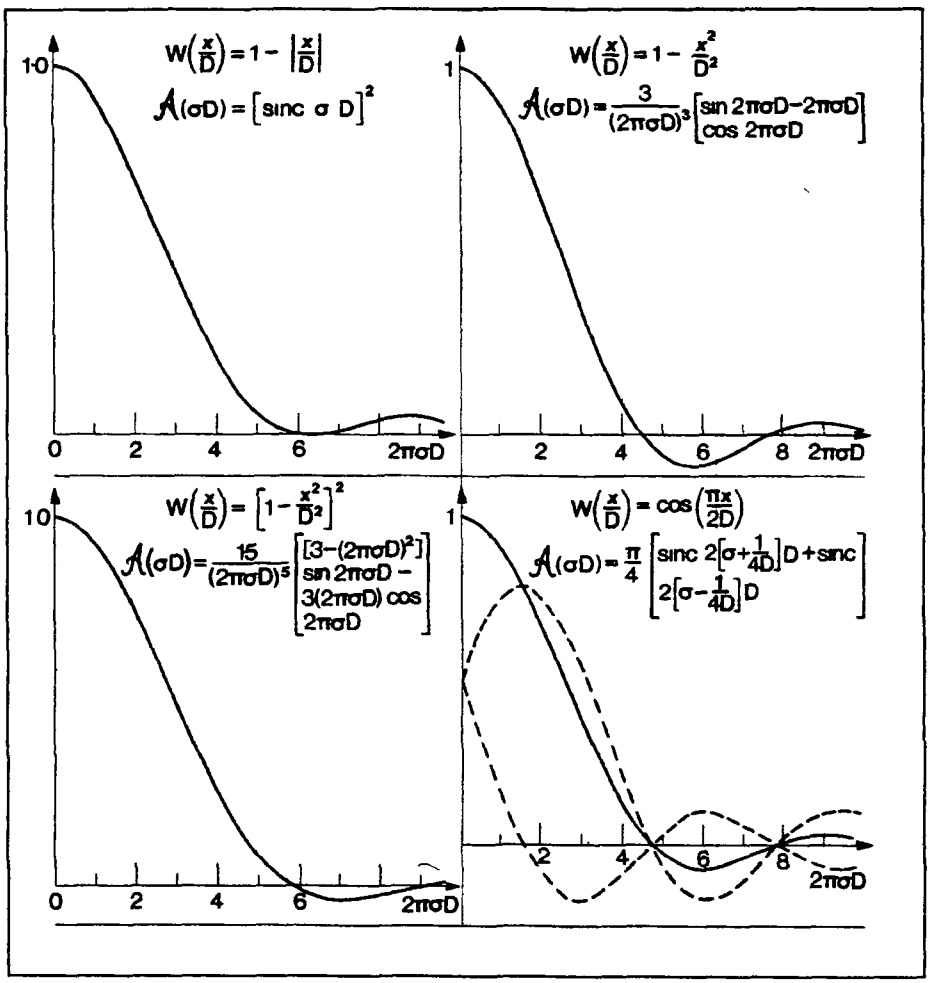


Fig. 2.5 Some practical weighting functions and their resulting normalised apodized spectral windows (adapted from Chamberlain, 1979) The dashed curves in the lower right inset are the component sinc functions which add to give the solid curve

The interferogram function is usually not continuous but rather sampled at discrete intervals of path difference β The Dirac δ-function can conveniently be used to represent sampling in the Fourier integral 2 34 If sampling takes place at discrete x-values $x=r\ \beta$, where r denotes an integer $0\leq r\leq N$, and $D=N\ \beta$ is the maximum path difference, then eqution 2 34 can be rewritten as

$$B_e^c(\sigma)=\int_0^{N\beta} F_e(x)W\left(\frac{x}{D}\right)\Pi\left(\frac{x}{D}\right)\beta\sum_{r=-\infty}^{\infty}\delta(x-r\beta)\cos(2\pi\sigma x)\,dx \qquad (2\,38)$$

$$=B_e(\sigma)*A(\sigma N\beta)*\mathcal{F}\left\{\beta\sum_{r=-\infty}^{\infty}\delta(x-r\beta)\right\} \qquad (2\,39)$$

The Fourier transform of an infinite train of impulses is again an infinite train of impulses (for a detailed derivation see e g P V O'Neil, 1984) The exact solution is

$$\mathcal{F}\left\{\beta\sum_{r=-\infty}^{\infty}\delta(x-r\beta)\right\}=\beta\sum_{r=-\infty}^{\infty}\delta\left(\sigma-\frac{2\pi r}{\beta}\right) \qquad (2\,40)$$

If the light to be measured is strictly monochromatic, i e $B(\sigma)=\xi_0\delta(\sigma-\sigma_0)$ then the computed unapodized spectrum becomes

$$B_e^c(\sigma)=N\beta\xi_0\left[\mathrm{sinc2}(\sigma+\sigma_0)N\beta+\mathrm{sinc2}(\sigma-\sigma_0)N\beta\right]*\sum_{r=-\infty}^{\infty}\delta\left(\sigma-\frac{2\pi r}{\beta}\right)$$

$$=N\beta\xi_0\sum_{r=-\infty}^{\infty}\left[\mathrm{sinc2}\left(\sigma-\frac{2\pi r}{\beta}+\sigma_0\right)N\beta+\mathrm{sinc2}\left(\sigma-\frac{2\pi r}{\beta}-\sigma_0\right)N\beta\right] \qquad (2\,41)$$

It means, that the reconstructed spectrum consists of a series of features which are replicated at a period $2\pi/\beta$ In order to make sure that these aliases do not overlap, β must be equal or less than $\frac{\pi}{(\sigma_{max}-\sigma_{min})}$, where σ_{max} and σ_{min} are the upper and lower band limits of the light source respectively

As mentioned before, the interferogram recorded in a practical interferometer is normally slightly asymmetric about x=0 and/or the sampling point that corresponds to the zero path difference position is displaced by a small amount δ Both effects may be viewed as a phase difference $\phi(\sigma)$ in the spectrum, so one can write

$$F_{asy}(x)=\mathrm{Re}\int_{-\infty}^{\infty}B_e(\sigma)\exp^{(-i\phi(\sigma))}\exp^{(2\pi i\sigma x)}\,d\sigma \qquad (2\,42)$$

The ideal spectrum is

$$B_e(\sigma)=\int_{-\infty}^{\infty}F_e(x)\exp^{(-2\pi i\sigma x)}\,dx \qquad (2\,43)$$

Inversion yields

$$F_e(x) = \int_{-\infty}^{\infty} B_e(\sigma) \exp^{(2\pi i \sigma x)} d\sigma$$

$$= \int_{-\infty}^{\infty} B_e(\sigma) \exp^{(-i\phi(\sigma))} \exp^{(i\phi(\sigma))} \exp^{(2\pi i \sigma x)} d\sigma \qquad (2\ 44)$$

$$= F_{asy}(x) * \chi(x) \qquad (2\ 45)$$

where

$$\chi(x) = \int_{-\infty}^{\infty} \exp^{i\phi(\sigma)} \exp^{(2\pi i \sigma x)} d\sigma \qquad (2\ 46)$$

This means that if $\phi(\sigma)$ is known, $\chi(x)$ can be calculated and the even (symmetrical) interferogram can be reconstructed by convoluting $\chi(x)$ and the asymmetrical recorded interferogram $F_{asy}(x)$ In order to evaluate $\phi(\sigma)$ only a small portion of the interferogram in the vicinity of zero path difference is needed

$$\phi^c(\sigma, l) = \arctan \frac{q^c(\sigma, l)}{p^c(\sigma, l)} \qquad (2\ 47)$$

where

$$q^c(\sigma\ l) = \int_{-l}^{l} F_{asy}(x) \sin(2\pi\sigma x) dx \qquad (2\ 48)$$

and

$$p^c(\sigma, l) = \int_{-l}^{l} F_{asy}(x) \cos(2\pi\sigma x) dx \qquad (2\ 49)$$

This forms the basis for the algorithms that were written in C as part of a computer program used for data aquisition and processing in conjunction with stationary FTS (chapter 4 and appendix A)

2.4 Stationary FT-Spectrometers

Various kinds of stationary interferometers are possible There follows here descriptions of a variety of these For each type some performance limits will be discussed The use of extended sources can lead to a reduced fringe visibility in some of the arrangements if the light emerging from different coherent point-pairs on the source form interferograms at different positions in the detector plane In the analysis below (M -L Junttila, 1991) the contribution of each point-pair to the final interferogram is taken into account in order to obtain a mathematical expression for the interferometer output as a function of the source size From that expression the optimum size of the entrance aperture and hence the instruments etendue can be deduced For this discussion different spectrometer types will be classified into two groups, namely amplitude division and wavefront division instruments the latter of which have no need for beamsplitter devices Examples of amplitude devision spectrometers including stationary Michelson, Mach-Zehnder, Savart-plate and triangle interferometers will be analysed first Then the discussion will proceed to the double-mirror and biprism interferometers in which the use of a beamsplitter is avoided Finally, some methods for resolution and dynamic range enhancement are described in the last section

2.4.1 Stationary Michelson Interferometer

The diagram of a Michelson interferometer below (fig 2 5), shows that light emerging from the extended source S is collimated by lens L_1, before it is divided into two beams, each of which reflected back at the plane mirrors M_1 and M_2 respectivily The reflected beams then interfere at detection plane P

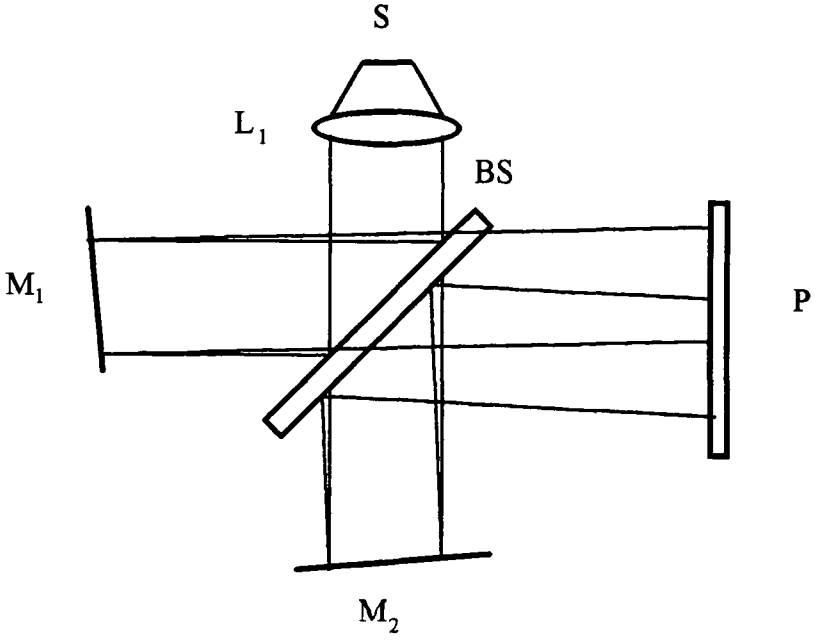


Fig. 2.6 Michelson interferometer with tilted mirrors M_1, M_2

The mirrors are tilted about their vertical axes to an angle 2α with respect to each other This introduces an optical path difference between off axis rays falling on P In the schematic diagram below (fig 2 7), the light rays coming from source S are collimated by lens L Because the beamsplitter was neglected in this figure, the reflected light interferes in the plane of the lens

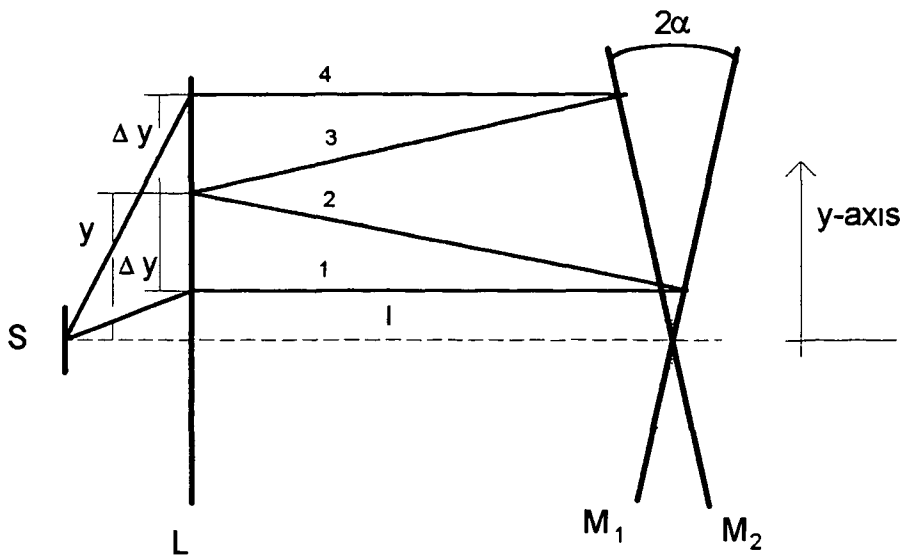


Fig. 2.7 Optical paths of light rays interfering at point y in Michelson interferometer with tilted mirrors

The optical path difference is

$$x(y)=(\text{path length 1})-(\text{path length 4})+(\text{path length 2})-(\text{path length 3})$$

$$=(l+(y-\Delta y)\sin\alpha)-(l+(y+\Delta y)\sin\alpha)$$

$$+\left(1+\frac{(y+\Delta y)\sin\alpha}{\cos\alpha}\right)-\left(1+\frac{(y-\Delta y)\sin\alpha}{\cos\alpha}\right) \qquad (2\ 50)$$

If α is small, first order approximation can be applied and equation 2 50 reduces to

$$x(y)\cong 2y\alpha \qquad (2\ 51)$$

The maximum path difference is $x_{max}=L\alpha$ where L is the width of the interferometer aperture

In case of a spatially coherent and quasi-monochromatic point source, this describes the formation of a line fringe interferogram in the detection plane By using equation 2 20 one gets the expression for the intensity distribution at P

$$I(y)= I(\sigma)_{tot}(1+\cos(4\pi\sigma\alpha y)) \qquad (2\ 52)$$

where $I(\sigma)_{tot}$ is the total power emitted by the source at one wavelength For polychromatic sources, equation 2 52 has to be rewritten as follows

$$I(y)=\int_0^\infty B(\sigma)\,d\sigma+\int_0^\infty B(\sigma)\cos(2\pi\sigma\alpha y)\,d\sigma \qquad (2\ 53)$$

This has the same form as equation 2 20, thus $B(\sigma)$ can be evaluated from the inverse Fourier transform of I(y)

If the instrument has a large aperture, i e the source is extended, rays entering the spectrometer off the optical axis are inclined to an angle β after collimation

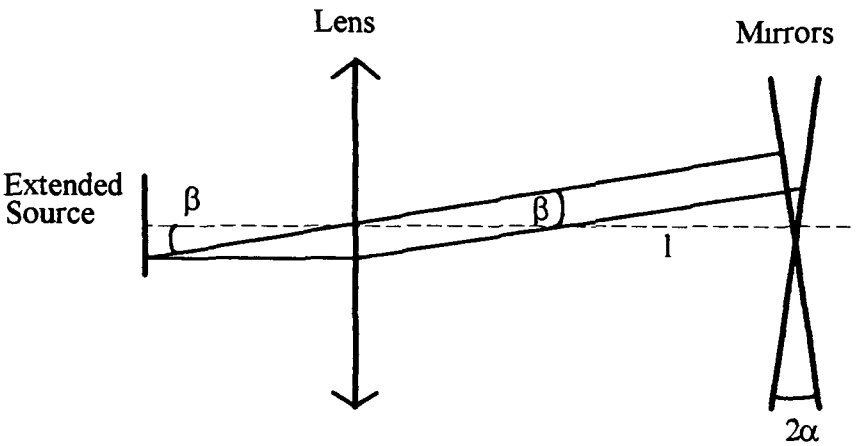


Fig. 2.8 Interferometer with extended source

Due to this angle, off-axis rays hit the mirrors at a position slightly shifted by approximately $l\beta$ with respect to the on-axis beam If this is taken into account in equation 2 50, one gets an expression for the optical path difference between off-axis rays

$$x(y,\beta)=(l+(y-\Delta y+\beta l)\sin\alpha)-(l+(y+\Delta y+\beta l)\sin\alpha)$$

$$+l+\left(\frac{(y+\Delta y+\beta l)\sin\alpha}{\cos\alpha}\right)-\left(l+\left(\frac{(y-\Delta y+\beta l)\sin\alpha}{\cos\alpha}\right)\right)$$

$$\cong 2\alpha(y+\beta l) \qquad (2\ 54)$$

The intensity, carried by the rays entering the interferometer, from a small area dA_S of the source is given by

$$dI_S = I_S \frac{dA_S}{A_S} \qquad (2\ 55)$$

where I_S is the intensity of the entire source The interferogram formed by monochromatic on- and off- axis rays can now be calculated by

$$I(y) = \int_{A_S} \frac{I_S}{A_S} dA_S + \frac{I_S}{A_S} \int_{A_S} \cos(4\pi\sigma\alpha(y+l\beta))\, dA_S \qquad (2\ 56)$$

Suppose dA_S represents a small slice of the source, then its size can be expressed as a function of the azimuthal angle φ (see also fig 2 8 below)

$$dA_S = r_S^2 \sin^2\varphi d\varphi \qquad (2\ 57)$$

and

$$\beta = \frac{r_S}{f}\cos\varphi \qquad (2\ 58)$$

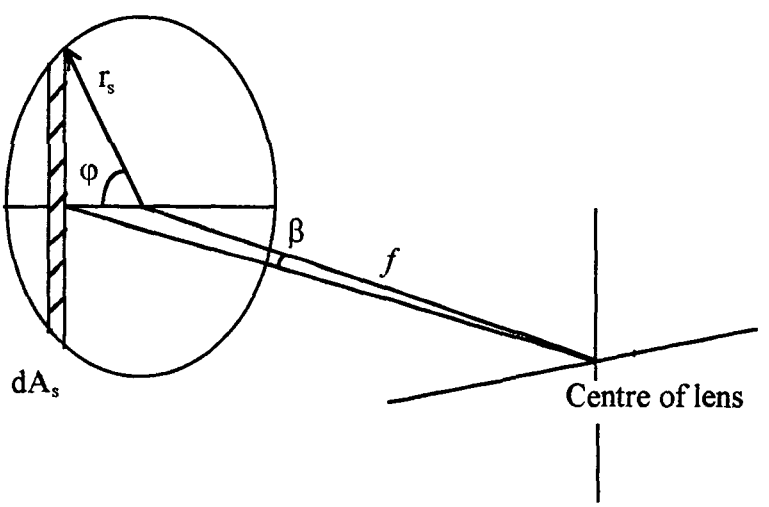


Fig. 2.9 Nomenclature for the circular source

Inserting equations (2 57) and (2 58) in (2 56) yields

$$I=I_S+\frac{I_S}{A_S}\int_0^{\pi}\cos\left(4\pi\sigma\alpha\left[y+\frac{lr_S\cos\varphi}{f}\right]\right)r_S^2\sin^2\varphi d\varphi \tag{2.59}$$

$$=I_S+\frac{I_S}{2\pi}\int_0^{\pi}\left(e^{i(a+b\cos\varphi)}+e^{-i(a+b\cos\varphi)}\right)\sin^2\varphi d\varphi \tag{2.60}$$

where $a=4\pi\sigma\alpha y$ and $b=\frac{4\pi\sigma\alpha lr_S}{f}$. By using the formula

$$J_\upsilon(z)=\frac{\left(\frac{z}{2}\right)^\upsilon}{\Gamma(\upsilon+\frac{1}{2})\Gamma(\frac{1}{2})}\int_0^{\pi}e^{\pm iz\cos\varphi}\sin^2\varphi d\varphi \tag{2.61}$$

equation (2.60) can be rearranged to

$$I=I_S+I_S\frac{2J_1(b)}{b}\cos a \tag{2.62}$$

This is the expression for an interferogram of spatial frequency $4\pi\sigma\alpha$ modulated by a first order Bessel function independent of y. Thus, the visibility is the same for the entire interferogram. If l and L are matched, b can be written as a function of the solid angle subtended by the source ($\Omega\approx\frac{\pi r_s^2}{f^2}$) and the maximum path difference x_{max}

$$b=4\sigma x_{max}\sqrt{\Omega\pi} \tag{2.63}$$

The full width at half maximum of a spectral line is $\Delta\lambda/\lambda^2=1.21/2x_{max}$, determined by the truncation sinc. Inserting equation 2.63 into this expression gives the resolving power as a function of Ω

$$R=\frac{\lambda}{\Delta\lambda}=\frac{b_{opt}}{2.4(\Omega\pi)^{1/2}} \tag{2.64}$$

The optimum value of b for fringe contrast can be found by maximising J(b). It reaches its first maximum at $b=1.85=b_{opt}$. Equation 2.64 suggests that it is sufficient merely to decrease the solid angle Ω in order to improve the instruments resolution. In fact, this will also change the value of b. This means, if b is to be kept at its optimum value of 1.85 the maximum path difference has to be increased. This, of course, leads to a better resolution.

2.4.2 Modified Mach-Zehnder Interferometer

The modified Mach-Zehnder interferometer (fig 2 9) below consists of one beam splitter and three mirrors adjusted in a way that the two virtual sources S_1, S_2 are inclined to an angle 2α to each other

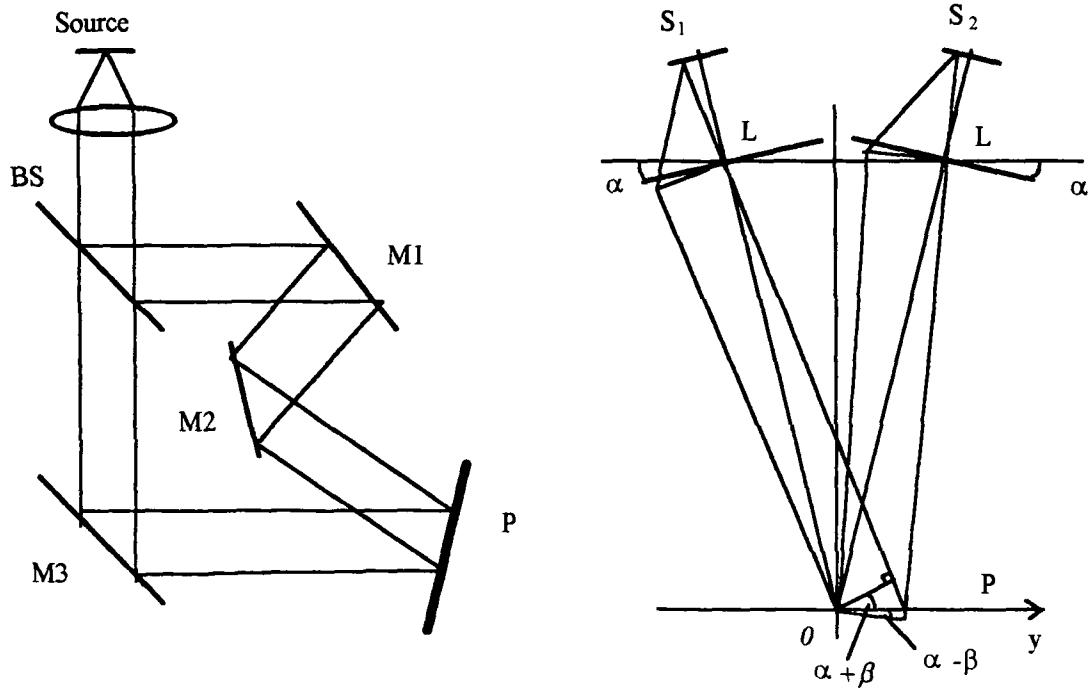


Fig. 2.10 Modified Mach-Zehnder interferometer with extended source and its virtual source drawing

The optical path difference for the rays falling on the detection plane is

$$\begin{aligned} x(y,\beta) &= y\sin(\alpha+\beta)+y\sin(\alpha-\beta) \\ &= 2y\sin\alpha(1-2\sin^2\frac{\beta}{2}) \end{aligned} \tag{2 65}$$

First order approximation yields

$$x(y,\beta) = 2\alpha y - \alpha y\beta^2 \tag{2 66}$$

Thus

$$I = I_s + \frac{I_s}{r_s^2\pi}\int_0^{\pi}\cos(4\pi\sigma y\alpha\left[1-\frac{r_s^2\cos^2\varphi}{2f^2}\right])2r_s^2\sin^2\varphi d\varphi$$

$$= I_s + I_s[J_0(b)\cos(z_0) - J_1(b)\sin(z_0)] \tag{2 67}$$

where $b = \frac{x(y)\Omega}{2\lambda}$ and $z_0 = \frac{2\pi}{\lambda}x(y)(1-\frac{\Omega}{4\pi})$ The fringe constrast is maximum at b=0 Unless the light source is infinitesimally small ($I_S \to 0$) the zero order Bessel

function on the left hand side of equation 2 67 modulates the interferogram resulting in a variant fringe visibility Furthermore, the interferogrm is distorted by the sine term the perturbation of which increases with increasing optical path difference

For light of a given wavenumber $\sigma_0=1/\lambda$, the lineshape in the spectogram is obtained by the inverse transformation of the cosine-modulation term in equation 2 67

$$S(\sigma)=S'(\sigma)+S'(-\sigma)$$

where

$$S'(\sigma)=\begin{cases}\left[\dfrac{\Omega\sigma_0}{2\pi(\sigma_0-\sigma)-1}\right]^{\frac{1}{2}} & \text{if } \sigma_0(1-\Omega/2\pi)<\sigma<\sigma_0 \\ 0 & \text{otherwise}\end{cases} \qquad (2\ 68)$$

The resolution can by approximated by

$$R=\frac{2\pi}{\Omega} \qquad (2\ 69)$$

It should be mentioned here, that some research has been undertaken to improve the fringe pattern generated by a modified Mach-Zehnder interferometer S Leon [5] developed a Mach-Zehnder / Fresnel biprism hybrid system that produced an approximately constant fringe contrast of 0 8 or 0 9 (with respect to 1 0 at the centre fringe) 20000 fringes away from the zero path difference position In comparison, the interferogram generated by a conventional Mach-Zehnder interferometer has a contrast ranging from 1 0 to 0 5 over the same region Leon reports that this improvement makes the hybrid interferometer well suited for the manufacturing of low noise holographic elements It can be assumed that his design would also enhance the performance of the Mach- Zehnder based FTS described above

2.4.3 Triangle Spectrometer

In a triangle spectrometer (fig 2 10a and virtual source diagram fig 2 10b), the optical path difference between the two coherent point pairs only depends on the angle of inclination, thus each point pair forms an interferogram of the same shape at the same position in the detection plane

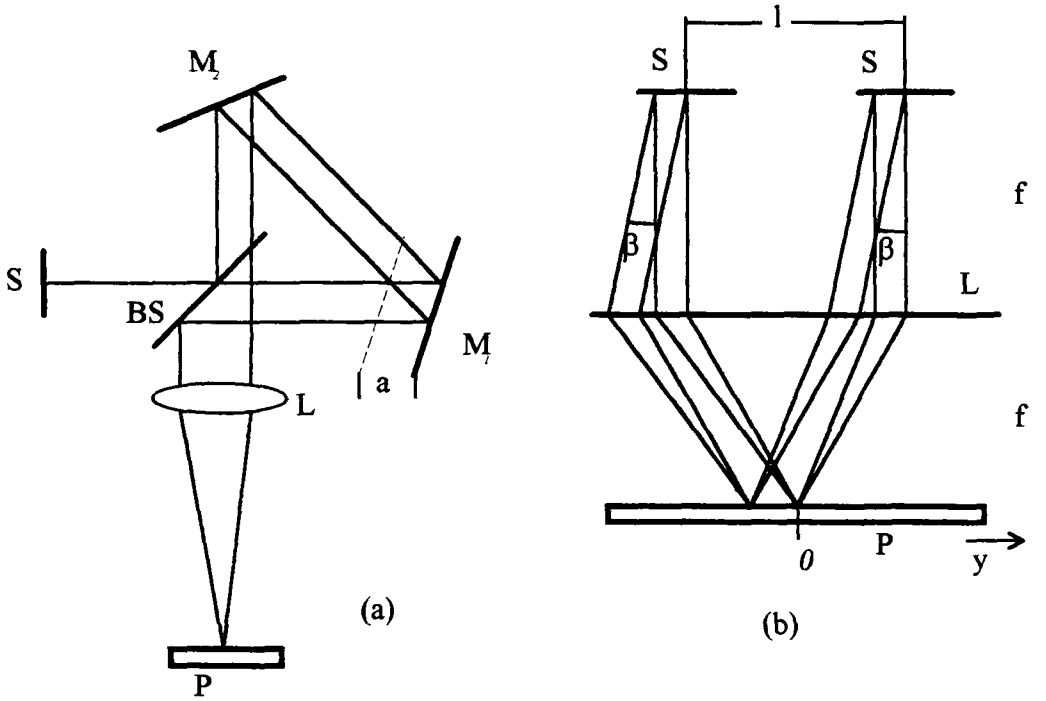


Fig. 2.11 Set-up (a) and virtual source diagram (b) of triangle interferometer with extended source

The optical path difference between the two coherent rays is given by

$$x = l\ \beta \tag{2 70}$$

and since $\beta = \frac{y}{f}$ for small angles

$$x(y) = \frac{l y}{f} \tag{2 71}$$

The separation between two coherent points is

$$l = \sqrt{2}a \tag{2 72}$$

where a is the shift of mirror 2 The resultant interferogram is

$$I = I_s + I_s \cos(k\frac{ly}{f}) \qquad (2\ 73)$$

As in the case of the stationary Michelson interferometer, the instrumental line profile is given by

$$\frac{\Delta\lambda}{\lambda^2} = \frac{1\ 21}{2x_{max}} \qquad (2\ 74)$$

where $x_{max} = \frac{lr_C}{f}$ is the maximum path difference and r_c is the aperture radius

Thus, the resolving power is

$$R = \frac{\lambda}{\Delta\lambda} = \frac{lr_C}{0\ 6\lambda f} \qquad (2\ 75)$$

Hence the resolution does not depend on the source size

2.4.4 Savart-Plate Interferometer

The key component in this type of interferometer is a Savart-plate (fig 2 11) It consists of two uniaxial crystals of the same thickness They are cut so that their optic axes are aligned at 45 degrees to the optical axis of the system The optic axes are lying in planes which are perpendicular to each other

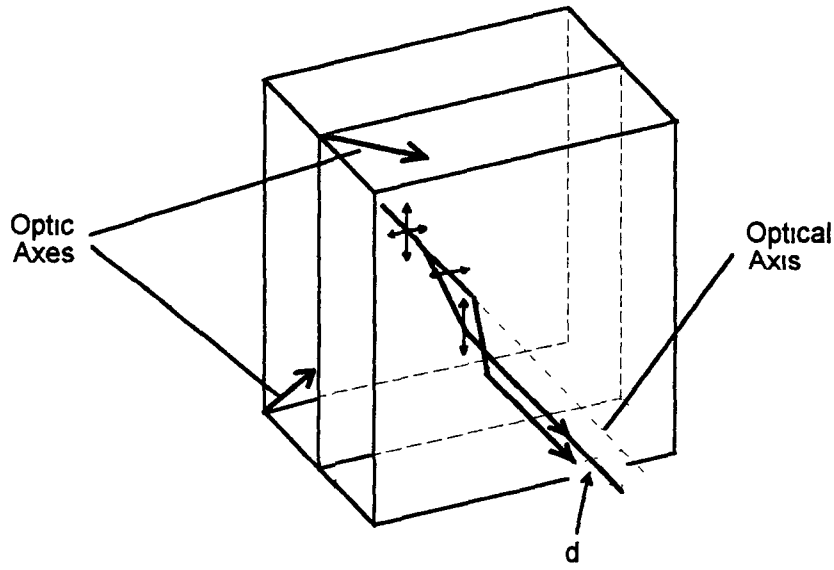


Fig. 2.12 A Savart-plate splitting a beam of light into two orthogonally polarised components

Light incident to the first crystal is refracted twice and two beams emerge, labeled the extraordinary and the ordinary beam The extraordinary ray emerges polarised in a direction perpendicular to the polarisation of the ordinary ray Both rays undergo another double refraction in the second crystal but since the optic axis has a different orientation the ordinary beam in the first crystal becomes the extraordinary ray in the second one and vice versa As a result, phase differences that were introduced between the two beams m the first crystal are canceled out The output rays are laterally sheared but not longitudinally sheared The seperation of the two beams is given by

$$d=\frac{(n_o^2-n_e^2)t}{\sqrt{2}(n_o^2+n_e^2)} \qquad (2\ 76)$$

where n_o and n_e are the ordinary and extraordinary refractive indices and t is the thickness of the Savart-plate

Unpolarised light from a single source is split to produce two virtual sources An analyser and a lens can be used to produce an interferogram in the focal plane of the lens (fig 2 12 below) To maximise the fringe contrast a second polariser is installed in front of the crystals to match the intensities in each of the beams

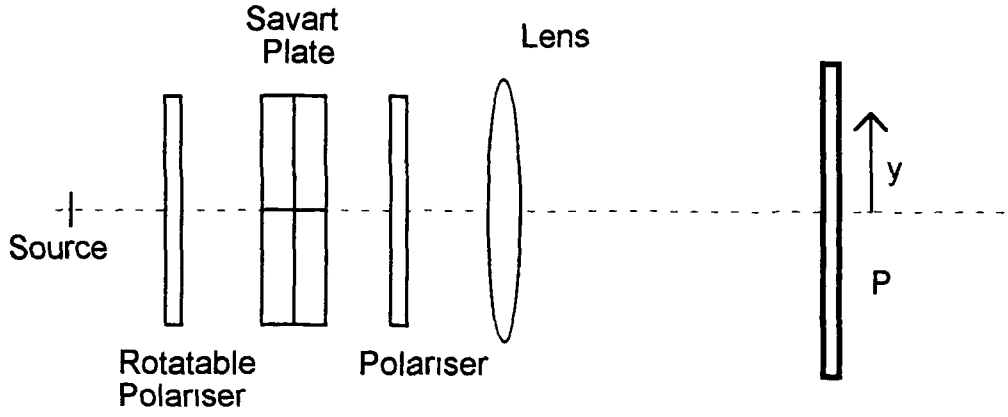


Fig. 2.13 A Savart-plate interferometer

The virtual source diagram and hence the equations giving the resolving power are identical to those for the triangular interferometer introduced in the previous paragraph Only the distance between the virtual sources formerly given by equation 2 72 has to be calculated using equation 2 76

2.4.5 Double Mirror Spectrometer

The diagrams (fig 2 13a,b) show the set-up and the virtual sources for a double mirror spectrometer The source has to be placed in front of the plane mirror without shading the detection plane

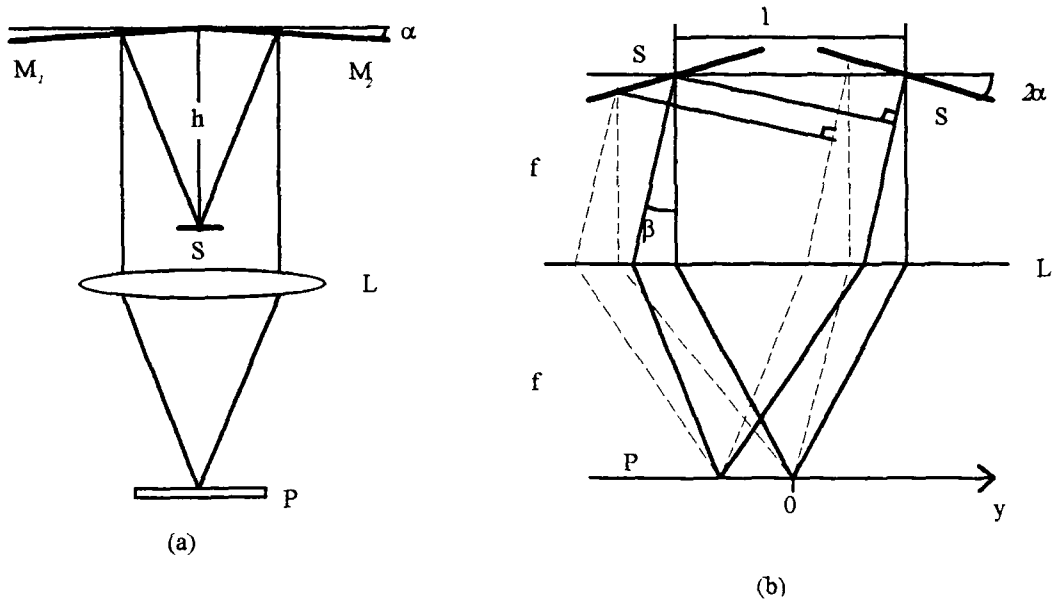


Fig. 2.14 Double mirror interferometer (a) and virtual source drawing (b)

The optical path difference of a coherent point pair at the back focal plane of lens L_1 is

$$x = l\beta \tag{2 77}$$

Using the geometrical relationships $l = 4h\alpha$ and $\beta = \frac{y}{f}$, equation (2 77) becomes

$$x(y) = 4h\alpha\frac{y}{f} \tag{2 78}$$

As it can be seen in figure 2 13, the coherent off-axis point pairs slide off the focal plane if the distance from the centre axis is increased The additional path difference for off-axis rays is $4\alpha r_s \cos\varphi$ (compare with fig 2 9) Thus, the total path difference for on and off axis rays is

$$x(y) = l\frac{x}{f} + 4\alpha r_s \cos\varphi \tag{2 79}$$

Performing the same integration as in the previous chapters, the expression for the interferogram is obtained

$$I = I_s + I_s \frac{2J_1(b)}{b} \cos(2\pi\sigma l \frac{y}{f}) \tag{2.80}$$

where $b = 16\pi\sigma\alpha r_s$ is again independent of y. As for the stationary Michelson interferometer, the instrumental function is of the sinc form and thus

$$\frac{\Delta\lambda}{\lambda^2} = \frac{1.21}{2x_{max}} \tag{2.81}$$

where $x_{max} = \frac{lr_C}{f}$. The resolving power is given by

$$R = \frac{r_C\, h\, b_{opt}}{1.2 f^2 (\Omega\pi)^{1/2}} \tag{2.82}$$

where again $b_{opt} = 1.85$.

Equation 2.82 resembles the equation for the resolution of the stationary Michelson interferometer, equation 2.64, but has two more parameters, the source distance h and the radius of the collimator r_C. An increase of either of the two figures leads to a greater maximum optical path difference and hence to a higher resolution of the instrument. The (theoretical) resolution is limited by the maximum path difference for a given range of wavelengths and cannot be improved merely by changing the geometrical layout of the spectrometer.

2.4.6 The Fresnel Biprism

A different type of wave front-splitting interferometer uses refraction rather than reflection to produce two virtual images of the source, coherent point pairs of which interfere in detection plane P. The optical device used to create these images is called Fresnel biprism and its design and function is shown in figure 2.14 overleaf. It is formed by two equal prisms of small refracting angle placed together base to base with their refracting edges parallel. A beam of light from a point source S is divided by refraction into two overlapping beams. These refracted beams are not strictly stigmatic, but because of the smallness of the refracting angle and of the angular aperture used, this aberration may be neglected and we suppose the prisms to form two virtual images S_1, S_2 of S.

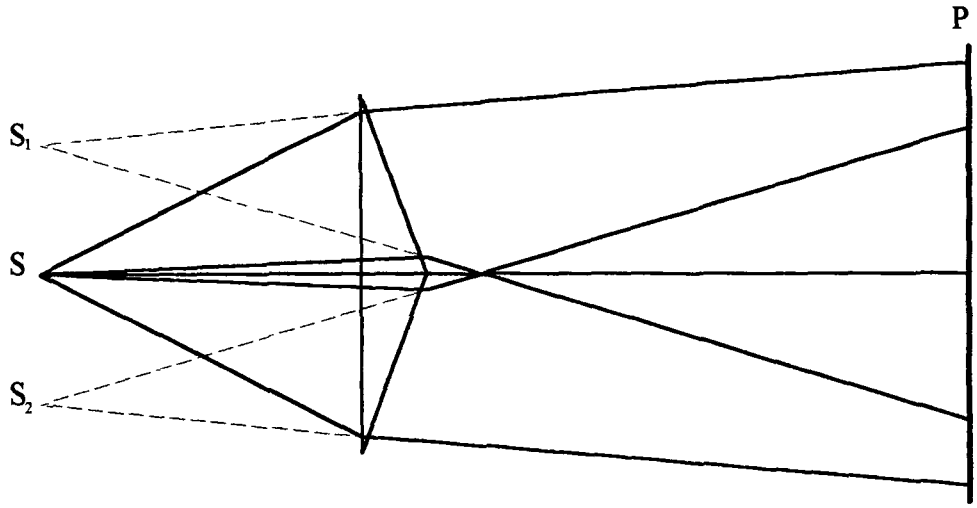


Fig. 2.15 Fresnels Biprism

As with all wavefront-splitting interferometers, the Fresnel biprism interferometer exhibits some difficulties if used with extended sources In fact, the fringe pattern loses visibility completely, if the source size exceeds its spatial wavelength, because the dark fringes created by marginal points on the source are then superimposed on the bright fringes produced by the centre point of the source, and vice versa

It was thought to overcome this problem by employing a lens to image the fringe pattern located at infinity onto its back focal plane With this arrangement only parallel light rays leaving the biprism under the same angle ω are focused down into the same point, thus the position of a fringe pattern does not depend on the position of its elementary source in extended source S Therefore all coherent point pairs on *S* produce a fringe pattern located in the same place on the back focal plane of the lens The optical path difference can be determined by the same assumptions valid for the triangle interferometer and is approximately given by

$$x = l\ \omega = \frac{l\ y}{f} \qquad (2\ 83)$$

where l is the separation of the two virtual sources, f the focal length of the lens and x the position on detector plane P As can be seen in diagram 2 15, only light rays incident at the angles α and α' for the upper and lower part of the biprism form an angle ω on the right hand side of the prism The relationships between those angles are

$$\alpha = \delta + \omega \quad \text{and} \quad \alpha' = \delta - \omega \qquad (2\ 84)$$

where $\delta = \gamma(n-1)$ is the angle of deviation of the prism with refractive index *n*.

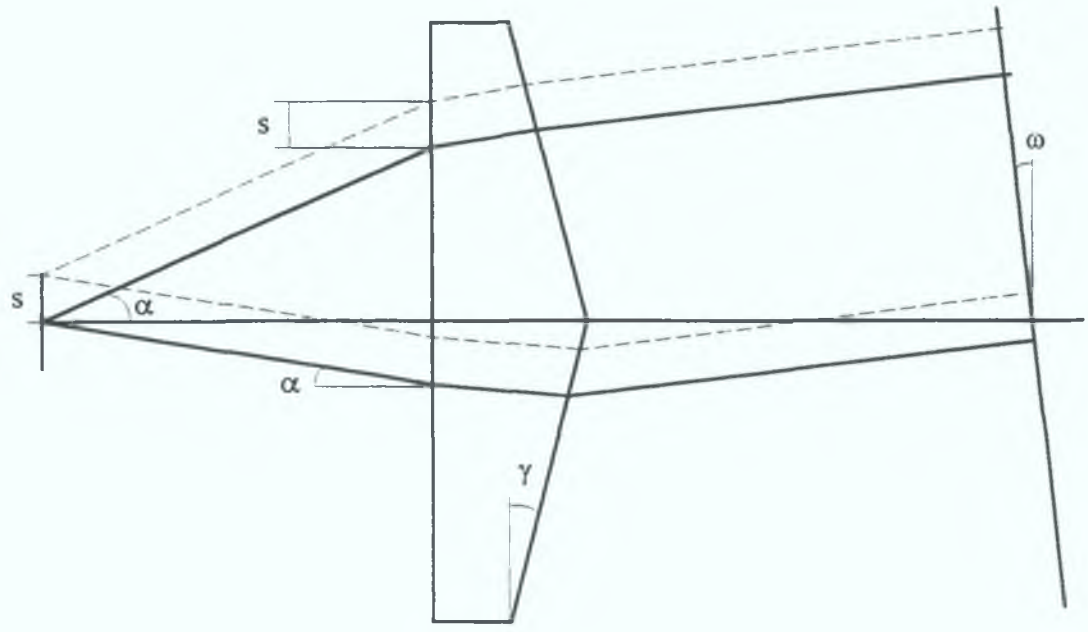


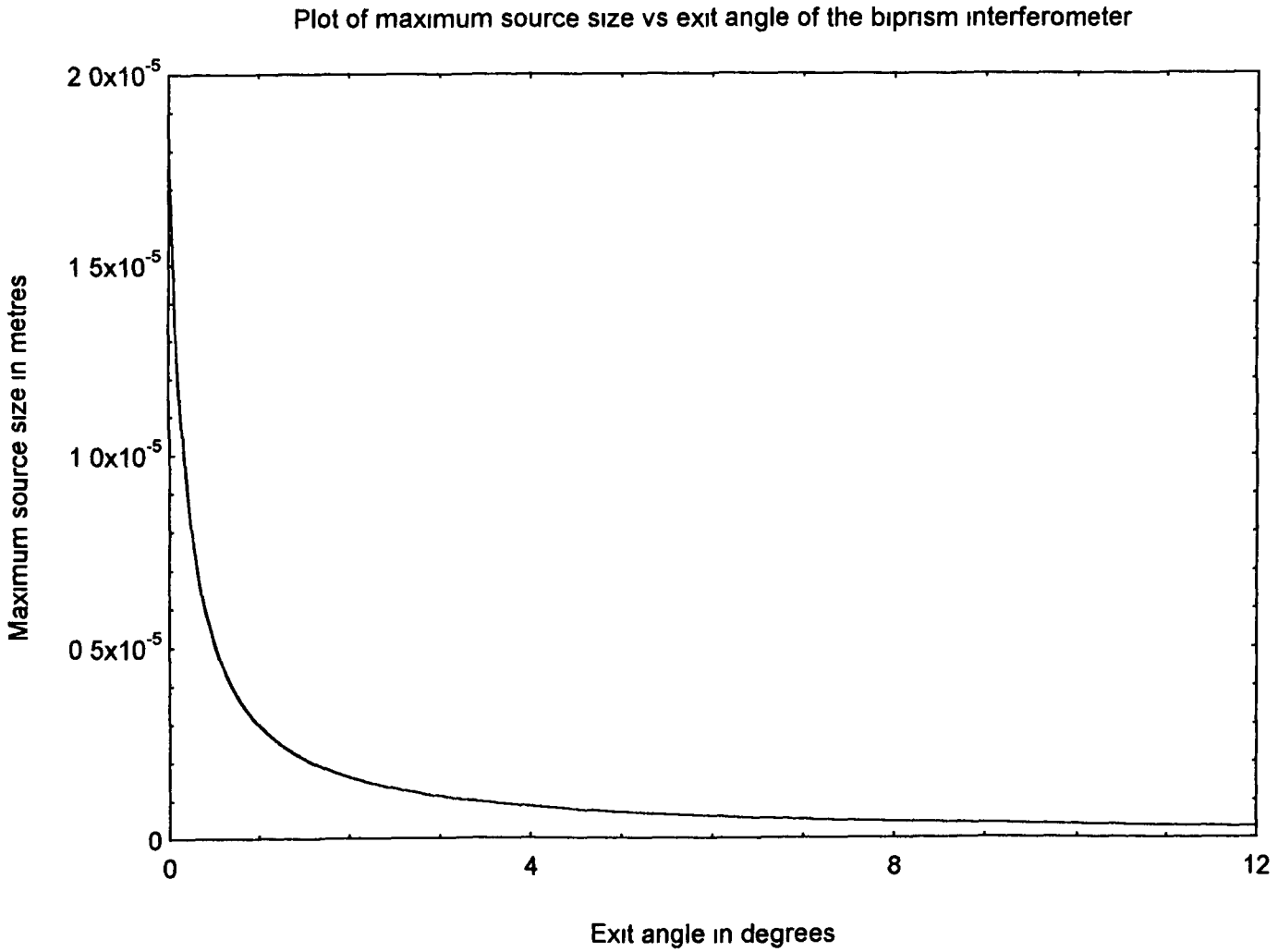
Fig. 2.16 Rays going out from centre and edge (dashed lines) of extended source under equal angles

In this arrangement, the angle ω cannot be bigger than δ, because α' would become negative in this case and interference would become impossible. The source size is limited by the fact that rays from different points on the source pass through different thicknesses of glass. The following equation gives the phase difference for neighbouring, parallel light rays, leaving the source from two points separated by the distance s:

$$\Delta p = 4\pi\frac{s}{\lambda}\cos\omega\,\tan(\omega-\delta)+$$

$$2\pi n\frac{s}{\lambda}\sin\left(\cos\left(\gamma-\arcsin\left(\frac{\sin(\omega+\delta)}{n}\right)\right)^{-1}+\cos\left(\gamma-\arcsin\left(\frac{\sin(\delta-\omega)}{n}\right)\right)^{-1}\right)$$

The graph on the following page is a plot of the maximum source size s_{max} versus the exit angle ω for a given phase shift of π/2, which is assumed to be small enough to maintain visibility in the fringe pattern. The prism parameters δ and γ are assumed to be 1.2 and 1.5 degrees respectively. It can be seen, that even for small angles of ω (corresponding to small optical path differences in the interferogram) the source size is limited to several tenths of microns.


Plot of maximum source size vs exit angle of the biprism interferometer
2 0x10-5
1 5x10-5
1 0x10-5
0 5x10-5
0
Maximum source size in metres
0
4
8
12
Exit angle in degrees

2.5 Methods of Performance Enhancement

As mentioned above, the resolution of Fourier Transform spectrometers is determined by the maximum path difference that can be introduced between the two interfering light beams The maximum optical path difference, though, is limited by the finite resolution of the detector array as an increased seperation between the two virtual sources leads to a higher spatial frequency in the fringe pattern However, if one is to measure light within a small spectral band steps can be taken to improve the resolution of the instrument considerably, at the expense of a reduced spectral range and some loss of throughput The two methods described below have been introduced by Okamoto, Kawata, Minami (1984) and T H Barnes (1986)

2.5.1 Resolution Enhancement Using a Heterodyne Technique

Consider figure 2 16 where virtual fringes formed by a Mach-Zehnder interferometer are imaged by a lens L onto a Moiré grating G

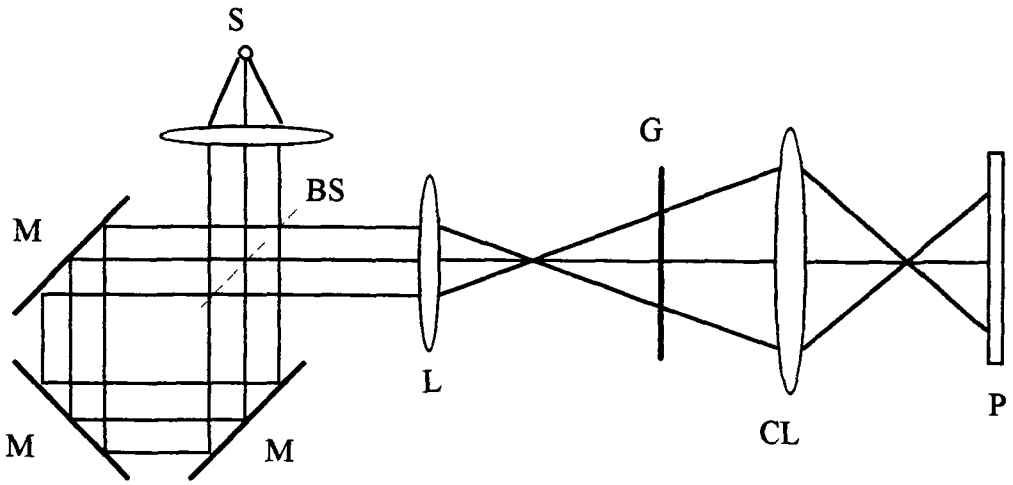


Fig. 2.17 Interferometer with Moiré grating G

If the grating is adjusted so that the grating lines are aligned with the fringes, Moiré fringes are formed between the grating and the interferogram Those Moire fringes are then imaged by lens L_2 onto the detector plane P

The Moiré grating consists of adjacent opaque and transparent areas with a transmission coefficient of 0 and 1 respectively The transmission may be written as a Fourier series

$$T(y)=\sum_{0}^{N} T_j \cos\left(2\pi\upsilon_0 j(y+d)\right) \qquad (2\,86)$$

where ν_0 is the fundamental grating frequency and d is the centering error of the grating

For a given wavelength λ fringes of spatial frequency ν are formed The fringe pattern may be written as

$$I(y)=I_0+I_0k(\upsilon)\cos(2\pi\upsilon y) \tag{2 87}$$

where $k(\upsilon)$ is the transmission factor of the imaging system The product of the two equations above yields an expression for the resultant fringe pattern at the detector array

$$\begin{aligned} I(y)=&\sum_{j=0}^{\infty} I_0T_j\cos(2\pi\upsilon_0(y+d)) \\ &+\sum_{j=0}^{\infty} I_0k(\upsilon)T_j\cos(2\pi(\upsilon+j\upsilon_0)y+2\pi\upsilon_0jd) \\ &+\sum_{j=0}^{\infty} I_0k(\upsilon)T_j\cos(2\pi(\upsilon-j\upsilon_0)y-2\pi\upsilon_0jd) \end{aligned} \tag{2 88}$$

The first term just represents an image of the grating The second term describes the formation of fringes at the sum frequency (interferogram of grating harmonic) The imaging system can be thought of as a linear filter so that the right choice of the modulation transfer function of the lenses ensures that those fringes at the sum frequency do not contribute to the heterodyne interferogram

It is the last term that represents the difference frequency heterodyne interferogram at the fundamental grating frequency (j=1) that is detected by the photodiode array Therefore, large optical path differences can be aligned while maintaining low spatial frequency Moiré fringes at the array

2.5.2 Degree of Resolution Enhancement

Let there be p grating lines across the interferogram and furthermore, let a wavelength of λ_0 give rise to p fringes in the original interferogram Any other wavelength λ will give rise to $\frac{\lambda_0}{\lambda}$p fringes in the original and $\left(\frac{\lambda_0}{\lambda}-1\right)$p fringes in the heterodyne interferogram The change in optical path difference between the interfering light rays at one fixed point in the interferogram is directly proportional to the spatial frequency of the interferogram itself Therefore, the optical path difference can be enlarged by a factor e until the heterodyne interferogram has the

same spatial frequency as the original one Because the change in optical path difference is linearly proportional to the spatial frequency, one may write

$$\left(\frac{\lambda_0}{\lambda}-1\right)=e\frac{\lambda_0}{\lambda} \text{ and } e=\frac{\lambda_0-\lambda}{\lambda_0} \qquad (2\ 89)$$

The resolution $\frac{\Delta\lambda}{\lambda^2}$ is inversely proportional to the maximum path difference, thus heterodyning improves the resolvin power by a factor $\frac{1}{e}=\frac{\lambda_0}{\lambda_0-\lambda}$

T H Barnes (1986) and his co-workers used a holographic grating with 830 lines across the interferogram and obtained 90 output fringes at 546nm Hence, λ_0 , corresponding to zero spatial frequency, was 605nm yielding a resolution enhancement of a factor of 10 25 This was sufficient to resolve the mercury doublet at 577-579nm using a photodiode array having only 512 elements The free spectral range extended from 452 to 605 nm

2.5.2 Resolution Enhancement using a Dispersion Element

This method introduced by Okamoto *et al* (1985) can be applied to most kinds of common path interferometer such as the triangular one used by the authors themselves Again, the free spectral range is traded off against an improved resolution Figure 2 17 shows the setup of a triangular common path interferometer, where an optical parallel is introduced into one of the arms of the interferometer

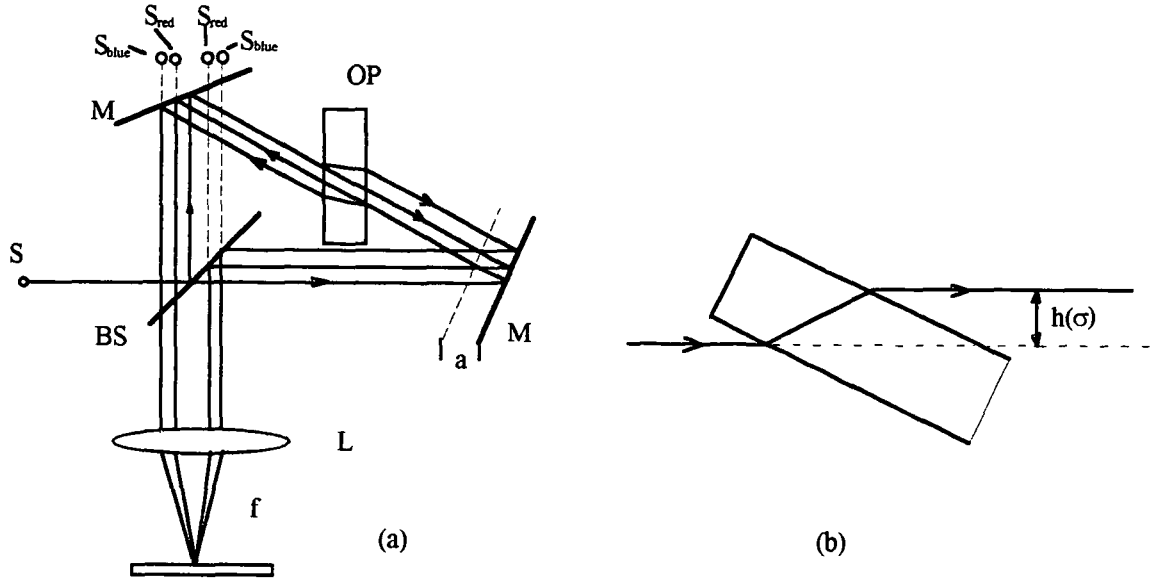


Fig.2.18 Triangular interferometer with optical parallel OP for resolution enhancement (a) and deviation of a light ray passing through an optical parallel (b)

As shown in figure 2.17b, if a light ray enters the parallel, it is refracted twice creating a shift $h(\sigma)$ with respect to the original direction. Because h depends on the wavelength of the refracted beam in a given spectral range, the positions of the virtual sources in figure 2.15 are also a function of the wavelength. In fact, the arrangement leads to even higher spatial frequencies in the detector plane for shorter wavelengths and lower spatial frequencies for longer wavelengths. As a consequence, the wavenumber dependence on the spatial frequency of the interferogram is enhanced.

2.5.3 Degree of Resolution Enhancement

In a specific spectral range, the spatial shift $h(\sigma)$ of the beam refracted by the optical parallel is approximately a linear function of σ

$$h(\sigma)=\alpha\sigma+\beta \tag{2.90}$$

The coefficients α and β are inherent to the material and of the optical parallel and should be positive. According to the geometry shown in figure 2.17, the distance $l(\sigma)$ between the virtual sources is given by

$$\begin{aligned} l(\sigma) &= 2h(\sigma) - \sqrt{2}a \\ &= 2\alpha\sigma + 2\beta - \sqrt{2}a \end{aligned} \tag{2.91}$$

where a is the shift of mirror M_1 from the position where the two beam paths coincide (see figure 2.17). The AC term of the interferogram I_m, detected by the mth element of N photodiodes at the focal plane of lens L is given by

$$\begin{aligned} I_m &= \int_{\sigma_{min}}^{\sigma_{max}} B(\sigma)\cos\left[2\pi\sigma l(\sigma)(dm-\phi)/f\right]d\sigma \\ &= \int_{\sigma_{min}}^{\sigma_{max}} B(\sigma)\cos\left[2\pi\sigma\left(2\alpha\sigma+2\beta-\sqrt{2}a\right)(dm-\phi)/f\right]df \end{aligned} \tag{2.92}$$

for m = 0,...,N-1,

where d is the pitch of the photodiode cells, and ϕ is the distance between the centre of the zeroth cell and the optical axis; σ_{min} and σ_{max} are the lower and upper bounds of the effective spectral range limited by the spectral response of the total system including the optical elements and the detector. The discrete Fourier transform of the interferogram obtained produces the discrete spectral distribution

$$B(\sigma)=\left|\sum_{m=0}^{N-1} I_m \exp\left(-2\pi \frac{\sigma l(\sigma) dm}{f}\right)\right| \qquad (2\ 93)$$

for $\sigma = \sigma_{min}$ to σ_{max}

In order to simplify the computation, let $k = \frac{dN\sigma l(\sigma)}{f}$ Then, equation 2 93 becomes

$$B(k)=\left|\sum_{m=0}^{N-1} I_m \exp\left(-2\pi \frac{km}{N}\right)\right| \qquad (2\ 94)$$

where k = 0, ,N

To reconstruct the spectrum, the relationship between the wavenumber σ and the discrete sampling point k must be known Without a dispersive element ($l \neq l(\sigma)$)

$$\sigma(k)=\frac{fk}{dNl} \qquad (2\ 95)$$

or

$$\sigma\left(2\alpha\sigma+2\beta-\sqrt{2}a\right)=\frac{fk}{dN} \qquad (2\ 96)$$

$$\sigma_k=\left(\frac{a}{\sqrt{2}}-\beta+\sqrt{\left(\frac{a}{\sqrt{2}}-\beta\right)^2+\frac{2\alpha fk}{dN}}\right)\frac{1}{2\alpha} \qquad (2\ 97)$$

for k = 0, ,N

If a spectral band ranging from σ_{min} to σ_{max} is to be examined, one has to set $\sigma_0 = \sigma_{max}$ and $\sigma_{N/2} = \sigma_{max}$ in equation 2 97 above to gain the most of the spectral information available and also to determine the coefficients α and β of the optical parallel

$$\alpha=\frac{f}{4d\sigma_{max}}(\sigma_{max}-\sigma_{min}) \qquad (2\ 98)$$

and

$$\beta=\frac{a}{\sqrt{2}}-\frac{f\sigma_{min}}{4d\sigma_{max}}(\sigma_{max}-\sigma_{min}) \qquad (2\ 99)$$

Now suppose no dispersive element is used Then, each discrete sampling point k represents a wavenumber σ_k ranging from 0 to σ_{max} Because the highest value of k is N/2, the distance between the two sampling points in wavelength terms is $2\sigma_{max}/N$ On the other hand, when the optical parallel is inserted, the free spectral range is limited to $(\sigma_{max} - \sigma_{min})$ The seperation of two sampling points then becomes approximately $2(\sigma_{max} - \sigma_{min})/N$ The improvement of resolution is given by the ratio of these two distances

$$RE = \frac{\sigma_{max}}{\sigma_{max} - \sigma_{min}} \qquad (2\ 100)$$

Okamoto and his co-workers (1985) report that the resolution was about three times improved by using an optical parallel made of high dispersive SF6 glass with a 25 43 Abbe number

2.5.3 Dynamic Range Enhancement

In any kind of interferometer described in one of the previous sections, noise produced by spurious spatial variations of intensity / sensitivity is added to the interferogram For example, photodiode arrays rarely have absolutely uniform diode-to-diode sensitivities and commonly exhibit so called fixed pattern noise, an effect giving rise to a detected pattern even when no light impinges on the array These effects, together with others such as spatial variations in the intensity of the light beam entering the interferometer, and variations in the transmission of the optics, give rise to systematic noise on the interferograms recorded by the system This noise effectively reduces the dynamic range of the interferogram and degrades the SNR of the final spectrum

Removal of this noise is most suitably achieved by background subtraction This technique can be applied e g to Mach-Zehnder or triangle interferometers where one of the mirrors is slightly tilted until the fringes in the interferogram disappear The remaining light signal can be considered as background noise and is subtracted from the interferograms obtained in subsequent measurements

The method has been applied successfully to the modified Mach-Zehnder spectrometer, described in a paper by T H Barnes (1985) and also to the double mirror spectrometer introduced by Marja-Leena Junttila (1992)

2.6 Summary

Stationary FT-spectrometers generally have low resolution limited by a relatively small number of pixels in the detector array. Therfore, in this section the optical throughput or etendue of the various instruments are compared and summarized, rather than their resolution. Recall that etendue was defined as

$$E = A_S \cdot \Omega_S = A_C \cdot \Omega_C \tag{2.101}$$

where A_C is the area of the aperture and Ω_C is the solid angle subtended at the source. The following estimates have been used:

$$R = \frac{\lambda}{\Delta\lambda} = 200$$

$r_C = 30\text{mm}$, or $A_C = 2730\text{mm}^2$

$r_S = 5\text{mm}$, or $A_S = 78.5\text{mm}^2$ for the triangular and Savart interferometer

$f = 300\text{mm}$

$h = 240\text{mm}$

The resulting throughputs are summarized in the table below.

	Resolv. Power	Etendue E	Result in mm^2
Stationary Michelson	$\frac{1.85}{2.4\sqrt{\pi\Omega}}$	$\frac{A_C}{3.4\pi R^2}$	$17.8\cdot10^{-6}$
Mach-Zehnder	$\frac{2\pi}{\Omega_S}$	$\frac{2\pi A_C}{R}$	22.2
Triangular	$\frac{1\cdot r_C}{0.6\lambda f}$	$\frac{A_S\cdot A_C}{2f^2}$	0.61
Savart	$\frac{1\cdot r_C}{0.6\lambda f}$	$\frac{A_S\cdot A_C}{2f^2}$	0.61
Double Mirror	$\frac{r_C\cdot h\cdot 1.85}{1.2f^2\sqrt{\pi\Omega}}$	$\frac{A_C}{11\cdot\pi\cdot R^2}$	$5.5\cdot10^{-6}$

It shows that the modified Mach-Zehnder, the triangular and the Savart interferometers have the highest throughputs at a resolution of 200 or 2nm at400nm wavelength. In case of the triangular and Savart interferometer the throughput can easily be improved by using a lens of shorter focal length or a greater source size. This and the fact that both instruments have no need for entrance slits were the reasons why it was decided to build one spectrometer of each type and compare their performance and practicality. This is described in the next chapter.

Chapter 3

Experimental

3.1 Introduction

In this chapter the design and testing of two different types of multichannel FT-spectrometer are described The triangular interferometer and the Savart plate interferometer (see previous chapter) were chosen and compared because both instruments provide easy optical alignment, robustness and a high optical throughput, qualities which are desirable in a general-purpose instrument Furthermore, both were built from "off the shelf" components in order to keep the costs low The data aquisition and processing was done by means of a PC because of the widespread use of PC compatible notebook computers, the size of which allow the design of a portable instrument It was also decided to write a computer program in C for the control of the detector array and the data sampling and processing routines Finally, the spectrum obtained was displayed on the computer screen The main difficulty was to interface the software and the detector The detector was connected to an evaluation board which provided signals to synchronise the photodiode array and the software In particular, a 'masterclock signal was the timebase for the detector's operation Here, it was also used to control the data read-out A routine written in assembly language and subsequently linked to the main C program was used for this task The video output of the evaluation board consisted of a train of boxcar pulses each pulse having a voltage depending on the light intensity incident to the pixel it represents The value of the voltage for each pulse was digitised and stored in an array Mathematical routines such as apodization, phase correction and discrete Fourier transform were applied to this array of numbers and finally the spectrum was reconstructed and displayed

3.2 Optical Set-Ups

The diagram overleaf (fig 3 1) shows the layout of the first instrument

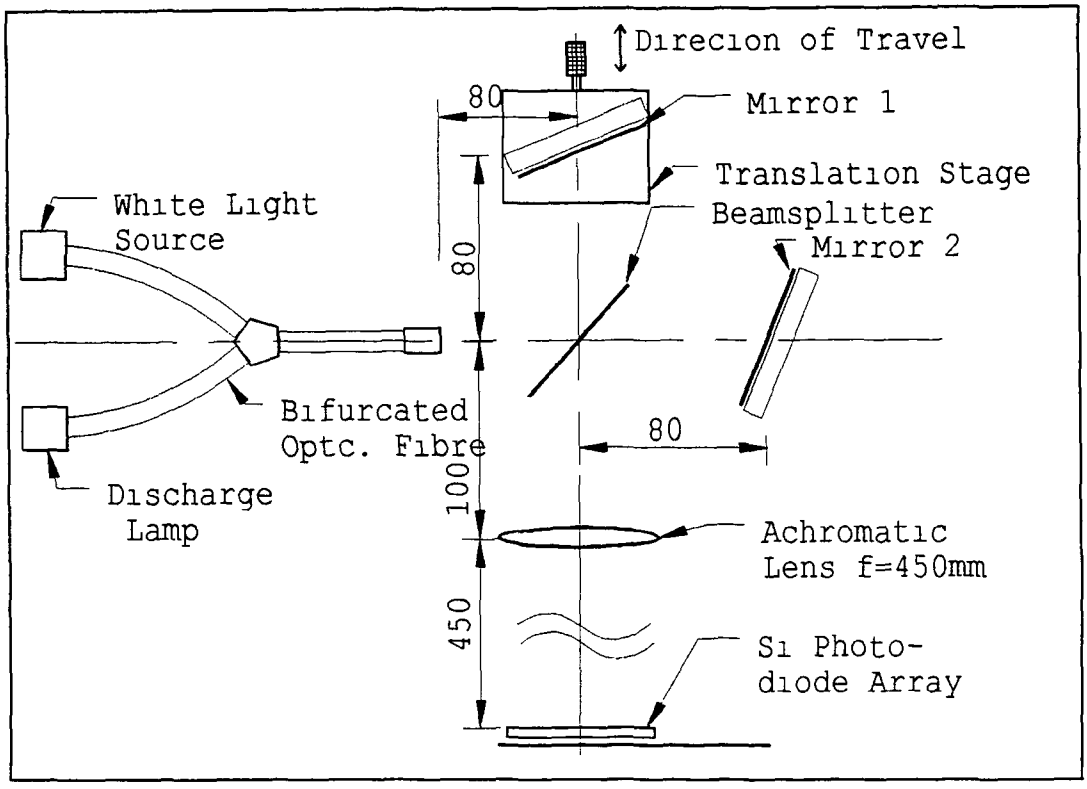


Fig. 3.1 Optical set-up of triangular interferometer (distances in mm)

The 1-inch diameter mirrors and beamsplitter were attached to the baseplate using adjustable kinematic mirror/beamsplitter mounts (Newport MM2-1A) In addition, mirror 1 was mounted onto a precision translation stage so it could be shifted along the z-axis as indicated in fig 3 1 A micrometer gauge allowed to measure the displacement to an accuracy of ±0 01mm The mirrors (Newport 20D20 AL 2) are stable broadband reflectors for general lab use The reflectivity is greater than 90% over the visible range combined with a surface flatness of $\lambda/10$

The probe beam was fed into the interferometer through a bifurcated optical fibre (Oriel 77533) This configuration allowed white light to be coupled into the probe beam so that the zero path difference position in the interferogram could be readily located The intensity of the white light source was just great enough to identify the origin without affecting the rest of the fringe pattern except for a small region near the zero path difference position A 2 inch diameter achromat of 450mm focal length was employed to localise the interference fringes at the detector array

The second instrument was intended to be used for teaching and demonstration purposes, so it should be robust and easy to handle Both the lens and the detector were translatable, therefore various lenses with different focal lengths could be used The Savart plate, supplied by Halbo Optics, Scotland, comprised two calcite

plates cemented together, each having less than λ/10 peak to peak wavefront distortion over the clear aperture. The diameter of the crystal was 15mm and the mutual displacement between the two virtual sources was specified as 0.6mm. However, later it was found out that the actual beam displacement was only 0.45mm so the original set-up shown in figure 3.2 was subjected to minor changes. In particular, the focusing lens was exchanged for a different one with a shorter focal length. Since the rest of the optics were layed out to be used in conjunction with a Savart plate producing a beam seperation of 0.6mm, the shortest focal length possible was now determined by the N.A. of the entrance optics.

In order to make sure that the entire array was filled, the focal length of the lens was to be not less than 235mm and, in fact, a lens of 260mm focal length was chosen. This led to a much reduced resolution of the instrument because this configuration did not privide the same maximum optical path difference as the original design.

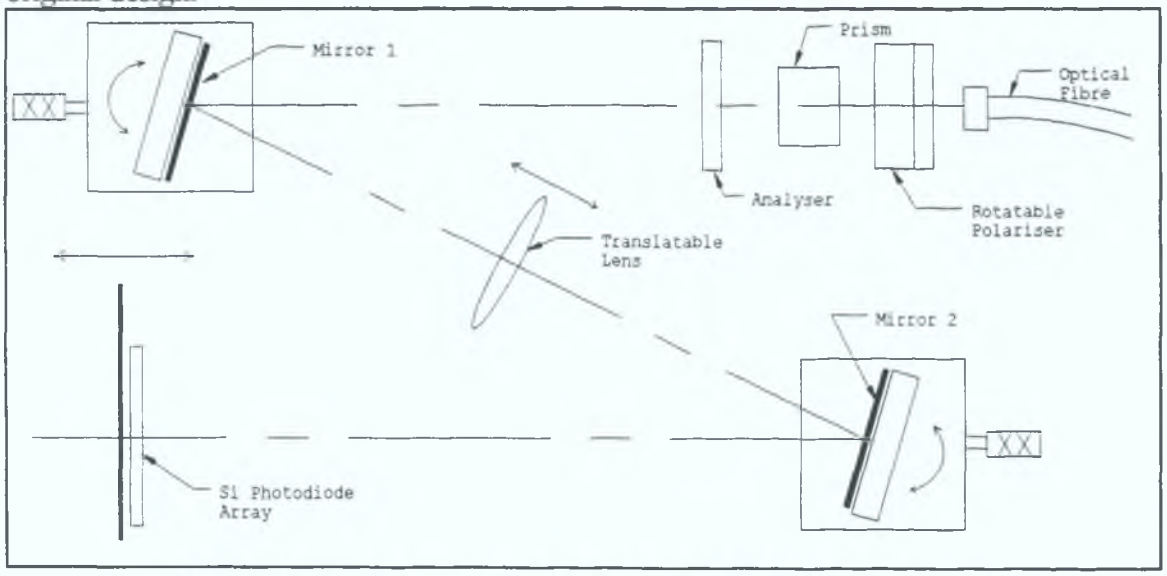


Fig. 3.2 Drawing of the original Savart-plate set-up, which had to be modified to accommodate lenses with shorter focal length.

The mirrors and polarizers in this interferometer were 2 inch in diameter each. The mirrors were held in position by kinematic mounts which were themselves attached to translation stages, providing easy alignment of the optics. Furthermore, the analyser was rotatable, so that an antiphase fringe pattern could be obtained for the purpose of background subtraction.

3.3 Electronics and Data Aquisition

The optical detection system, supplied by EG&G, consisted of a 1024-photodiode array (RL 1024S) and electronic aquisition and communication boards.

The photodiode array was mounted directly onto a small RC 1001 Satellite board, allowing it to be located compactly in the interferometer Each sensor element of the self scanning detector array has an aspect ratio of 100 1 or 2 5mm 25μm which makes it suitable for the detection of line fringes The spectral response ranges from 200nm to 1000nm, combined with a low dark current (~5pA at 25°C) and low noise due to small output capacitances

The satellite board was connected to an RC 1000 mother board which provided signal processing, syncronisation and instruction signals for aquisition by PC A master clock TTL signal and a sync output provided the time base to synchronise the read out of the video signal to the video channel

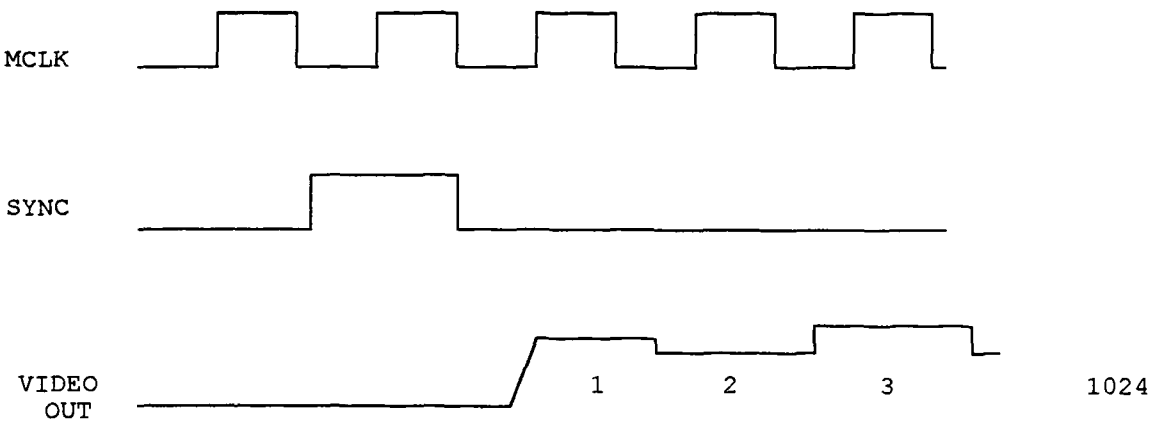


Fig. 3.3 Timing Diagram of the Signals used for Data Aquisition

The motherboard was interfaced with the PC by means of an MPIBM3 Bytronics expansion board featuring digital I/O lines as well as an ADC which was used to digitise the video signal The maximum sampling rate of the ADC was 40 kHz at 12 bit resolution

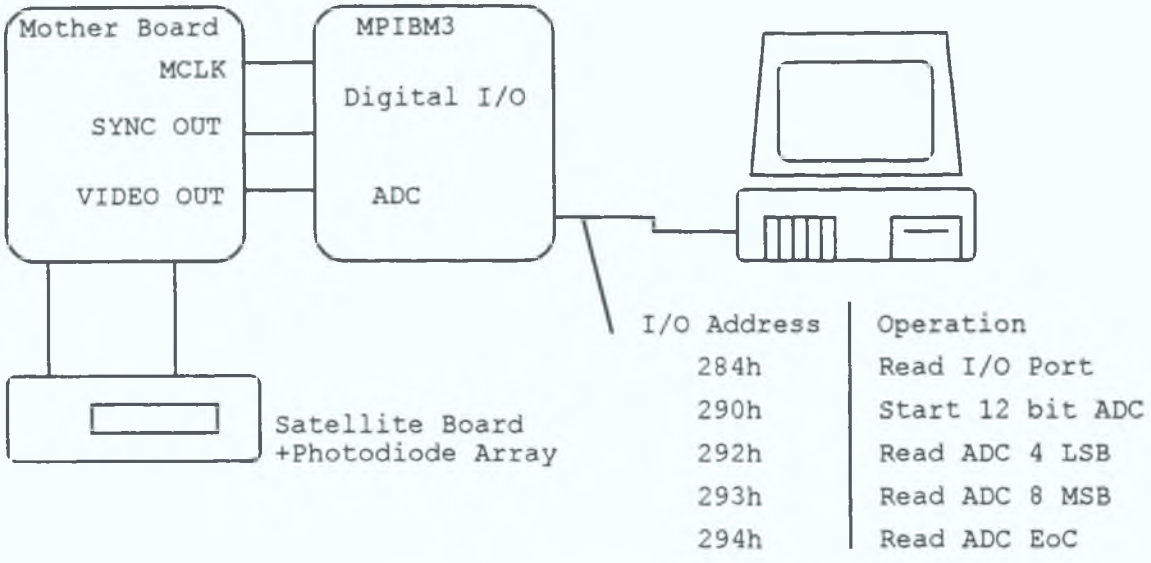


Fig. 3.4 Data Aquisition System

A routine was written in assembly language (Appendix [C]) for control of, and collect of data from the I/O card. A program was written in C (Appendix[B]), which performed the more mathematical routines such as the descrete Fourier transform. When executed, the routine returned the address of an one dimensional array containing the signal of each pixel of the detector to the main program. There, the background was subtracted, and after apodization and phase corrections were applied to the interferogram, the spectrum was reconstructed and displayed on the screen. The flow-chart overleaf shows the steps involved.

3.4 Summary

Although both instruments provided relatively easy optical alignment, the triangular interferometer proved to be more difficult to set up than the Savart-type interferometer. A laser beam was found to be useful for the adjustment of the first apparatus: The laser light was launched directly into the instrument without using the optical fibre. The mirrors, the beamsplitter and the lens were adjusted until the two beams of light formed in the interferometer coincided in the detector plane. Then, the lens was removed and the same procedure was applied again. The optical axes within the interferometer are aligned, if only one spot of light can be seen in both cases.

The laser was also used for the coarse adjustment of the Savart-plate interferometer. Initially, the lens and the polarisers were removed and the direction of polarisation of the laser was aligned at 45 degrees to the baseplate of the system. The Savart plate was now rotated until the two spots of laser light appeared on a

horizontal line in the detector plane Then the polarizers were put back into place and rotated until both spots had maximum and equal intensities

The final adjustment of both instruments was done during their operation simply by maximising the fringe contrast in the interferogram displayed on the computer screen

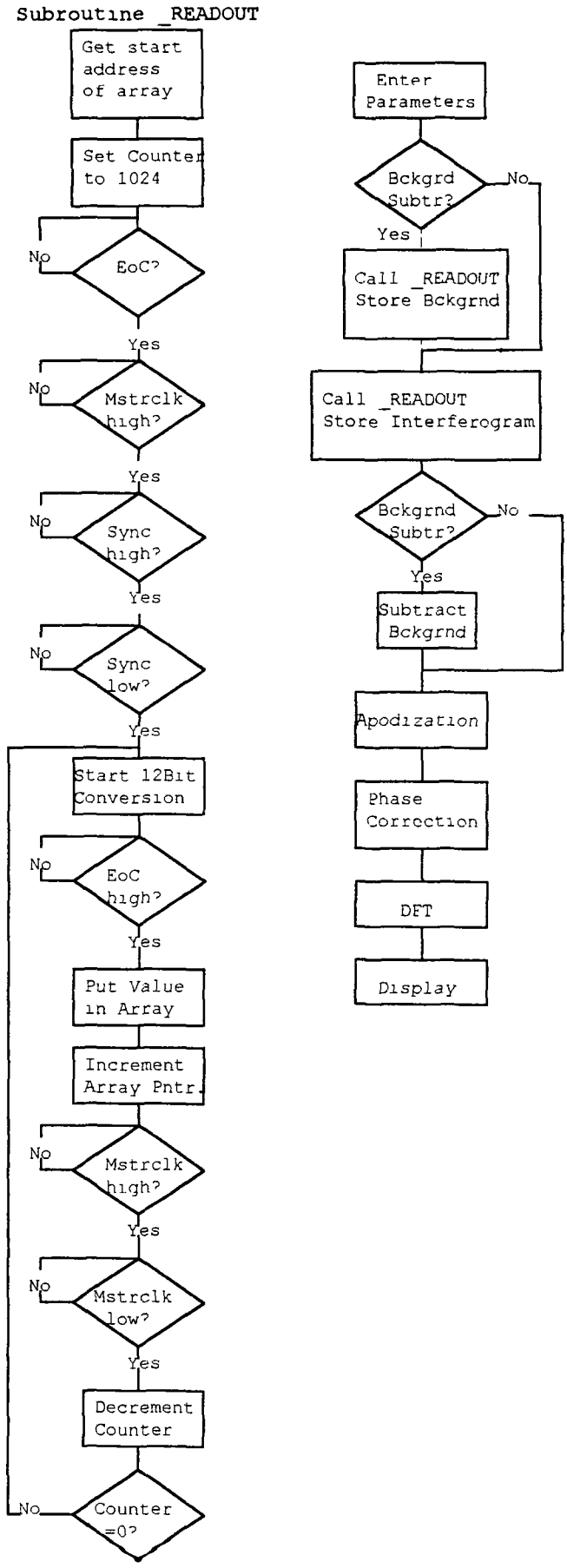


Fig. 3.5 Flow Chart of the Data Aquisition and Processing Program

Chapter 4

Results and Discussion

4.1 Introduction

Both the triangular and Savart type instruments were used to measure the emission spectra of low pressure mercury, cadmium and sodium lamps The spectra obtained are compared with the actual ones found in literature [e g The Handbook of Chemistry and Physics, 1995, CRC, Cleveland] Also, the light emitted from a red colour LED was measured since stationary FT spectrometers are better suited for wide band radiation spectra measurements Initially it was thought to use a He-Ne laser for the necessary wavelength calibration of the instruments However, due to the high temporal coherence of the laser light the interferograms were distorted by speckles produced by the fibre bundle In the following, the complete procedure of taking a spectrum is described

4.2 Starting the Measurement

Before the interferogram produced by a particular light source could be recorded the interferometer-detector system had to be aligned The software supported this alignment procedure by means of a continuous sampling option available from the 'sample' menu which is displayed on the screen after the computer has been turned on (figure 4 1) The menubar is always shown on top of the screen and the menus can be operated via the cursor keys unless entry of data is required A detailed description of each option follows below

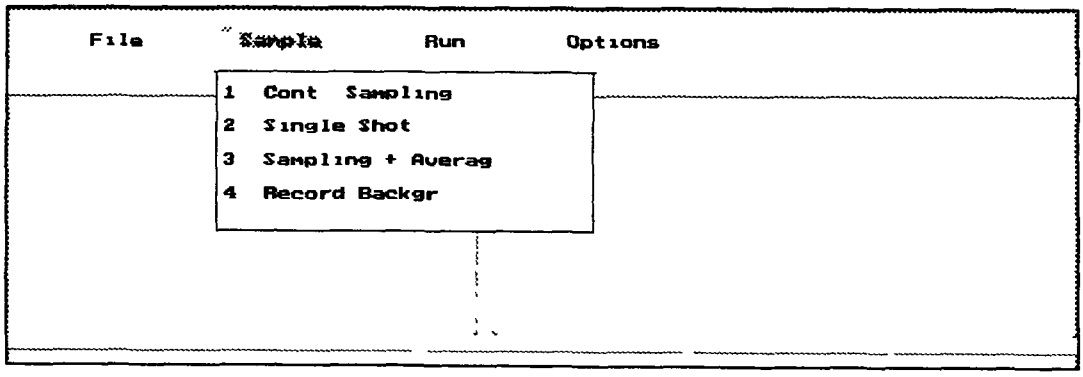


Fig. 4.1 The sample menu

In the 'continuous sampling' mode the detector is read out continuously and the light signal picked up by the detector is displayed The screen is refreshed between each cycle

Figure 4 2 shows the interferogram produced by the light of the sodium lamp after the adjustment of the interferometer was completed

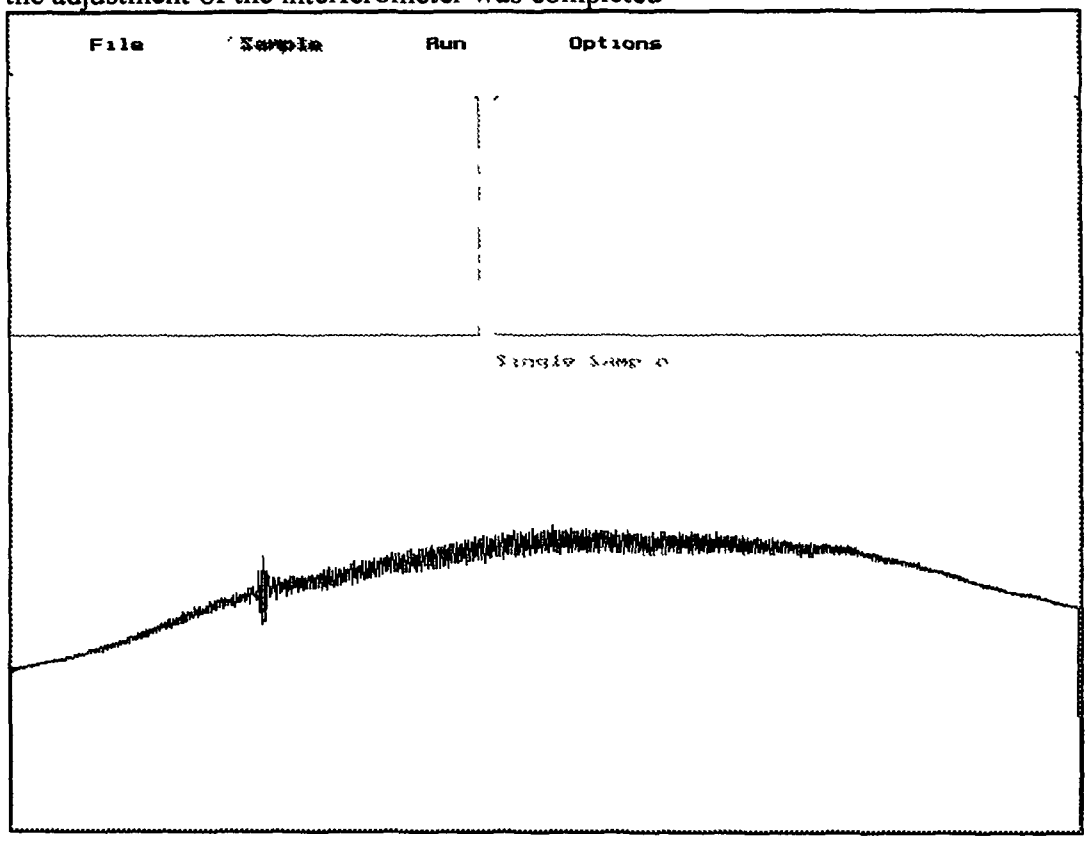


Fig. 4.2 Sodium interferogram

The position of zero path difference in the interferogram lies roughly in the centre of the left half of the screenbox and the fringe contrast was set to a maximum

In order to Fourier transform the raw data the optical path difference between two adjacent pixels in the detector array needed to be found An estimate for this value was calculated using equations 2 71 and 2 72 and entered in the options menu (fig 4 3) A calibration routine to obtain a more accurate figure for the optical path difference is described in the next section

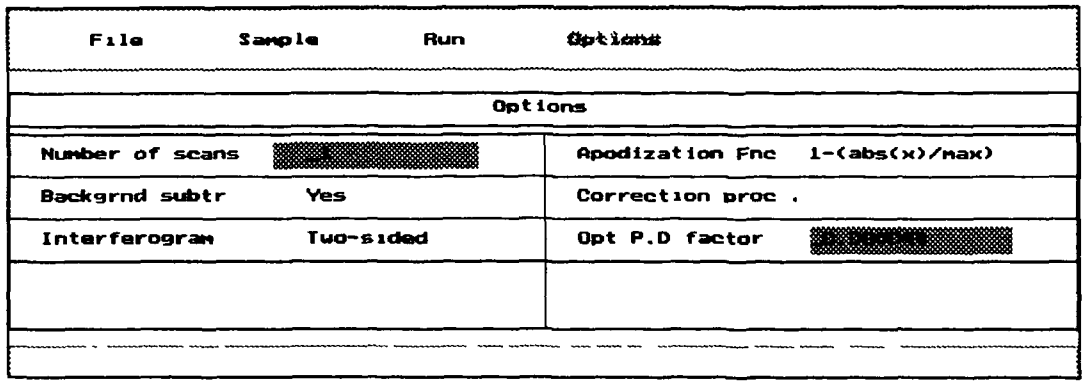


Fig. 4.3 The options menu

The instrument was now ready to measure a spectrum The 'run' option from the menubar starts the complete routine of recordmg the interferogram, processmg the raw data and finally displaying the spectrum on the screen Figure 4 4 overleaf shows a mercury spectrum obtained in one of the early test runs using the triangular interferometer

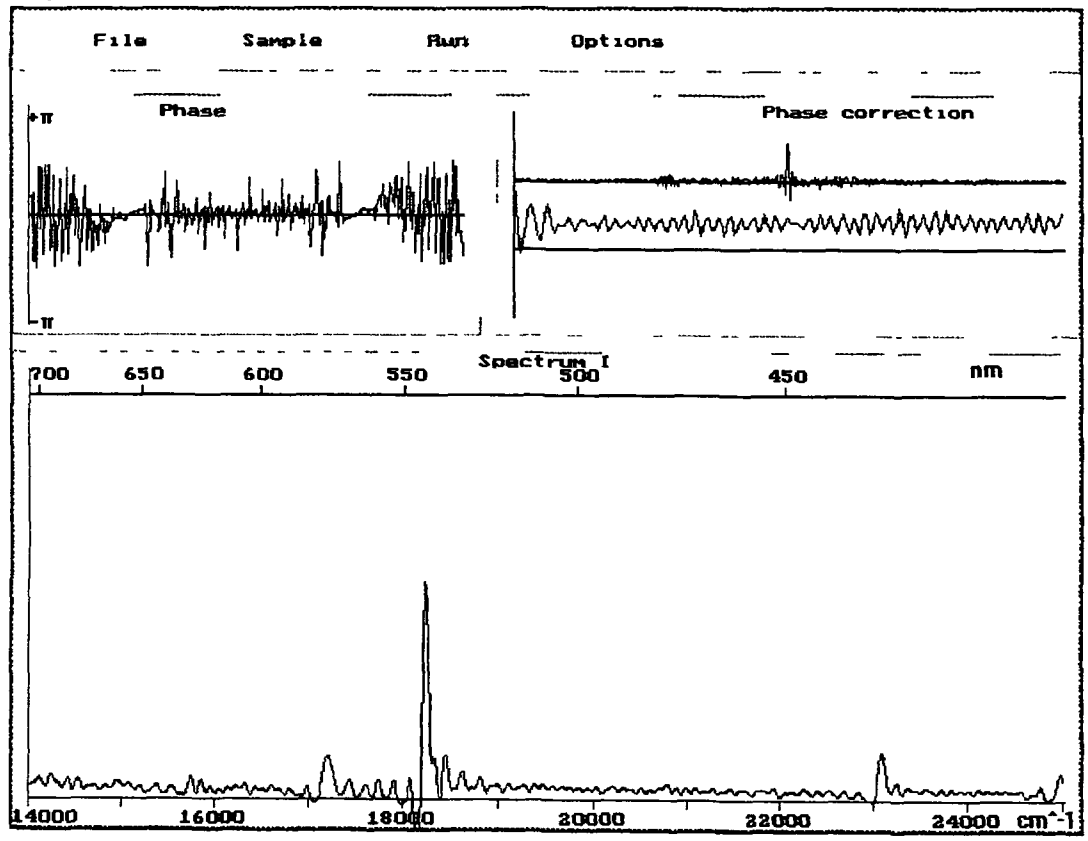


Fig. 4.4 Mercury spectrum

The two screen boxes on top of the spectrum contain the phase function and the phase correction function according to equations 2 47 and 2 46 The second graph in the screen box labelled "phase correction" is a enlarged view of the corrected

interferogram with the peak of the zeroth fringe intersecting the y-axis Again, a more detailed description of the effects of phase correction will be given later

4.2.1 Calibration

The mercury spectrum in figure 4 4 exhibits not only a rather poor SNR and symmetry, it is also slightly shifted along the wavelength axis For instance, the mercury green line (546 nm) in the centre of the graph should be at 18314 cm^{-1} in wavenumber terms but is, in fact, at ~18230 cm^{-1} or 548 5 nm This means the corresponding optical path difference (OPD) between two adjacent pixels in the detector array, originally calculated with equations 2 71 and 2 72, was slightly off the true value The true value was found by iteratively measuring the sodium spectrum Between each measurement the OPD was altered in the options menu until the sodium D-line appeared in the correct position at 588 5 nm or 16950 cm^{-1} Figure 4 5 shows the sodium spectrum obtained after wavelength calibration

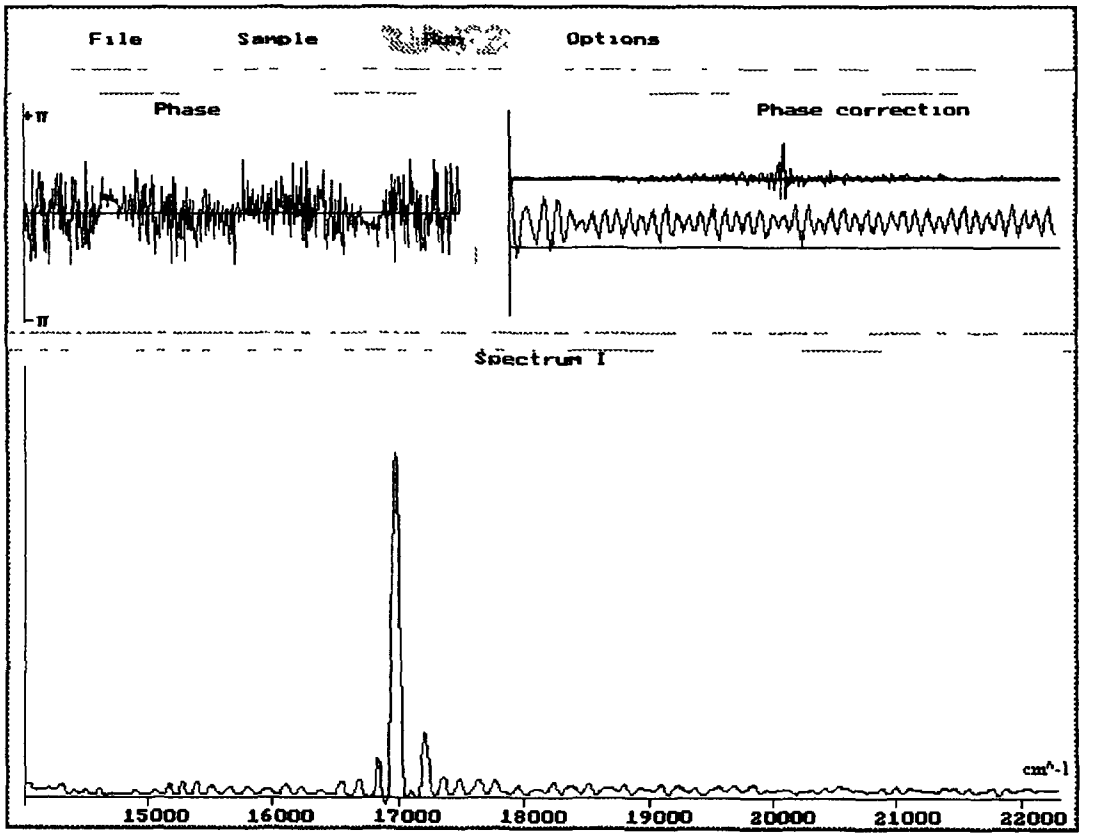


Fig 4.5 Sodium spectrum after wavelength calibration

Since the optical path difference can be adjusted in steps of tenth of nanometres a spectral line can also be shifted in intevals of the same order of magnitude

A problem that arose after calibration can be seen in the next graph (fig 4 6) It shows a mercury spectrum recorded after wavelength calibration with the sodium lamp

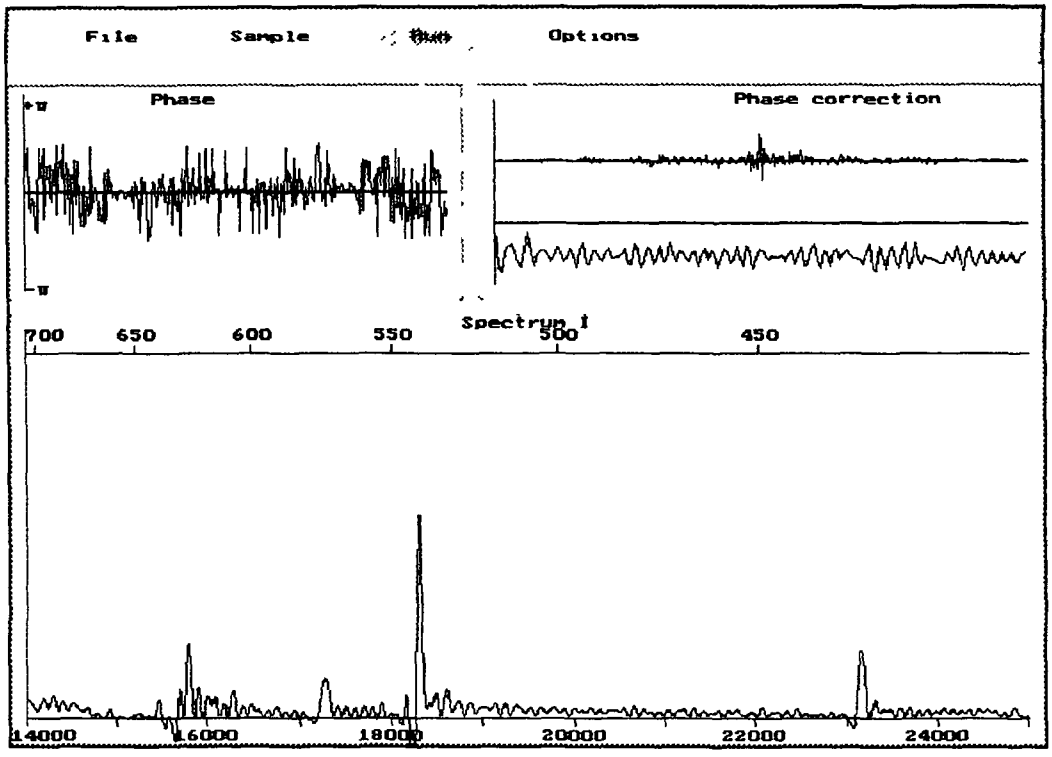


Fig. 4.6 Mercury spectrum

The lines that are well away from the sodium line are somewhat off the correct position For example, the 435 nm line appears at ~23050 cm^{-1} or 434 nm Since the lines far away from the calibration wavelength have not broadened it is unlikely that the wavelength-shift as a function of the distance from the calibration wavelength is due to geometrical abberation effects of the optical devices Rather, it is caused by a variation of the distance between the two virtual sources in the interferometer as a function of wavelength Therefore, the effect is probably due to chromatic dispersion in the beamsplitter and collimating lens It can be shown that chromatic dispersion in the beamsplitter does in fact affect the spatial frequency of the interferogram dependend on the colour light used Consider figure 4 7 overleaf It shows light rays as they pass through the beamsplitter in the triangular interferometer

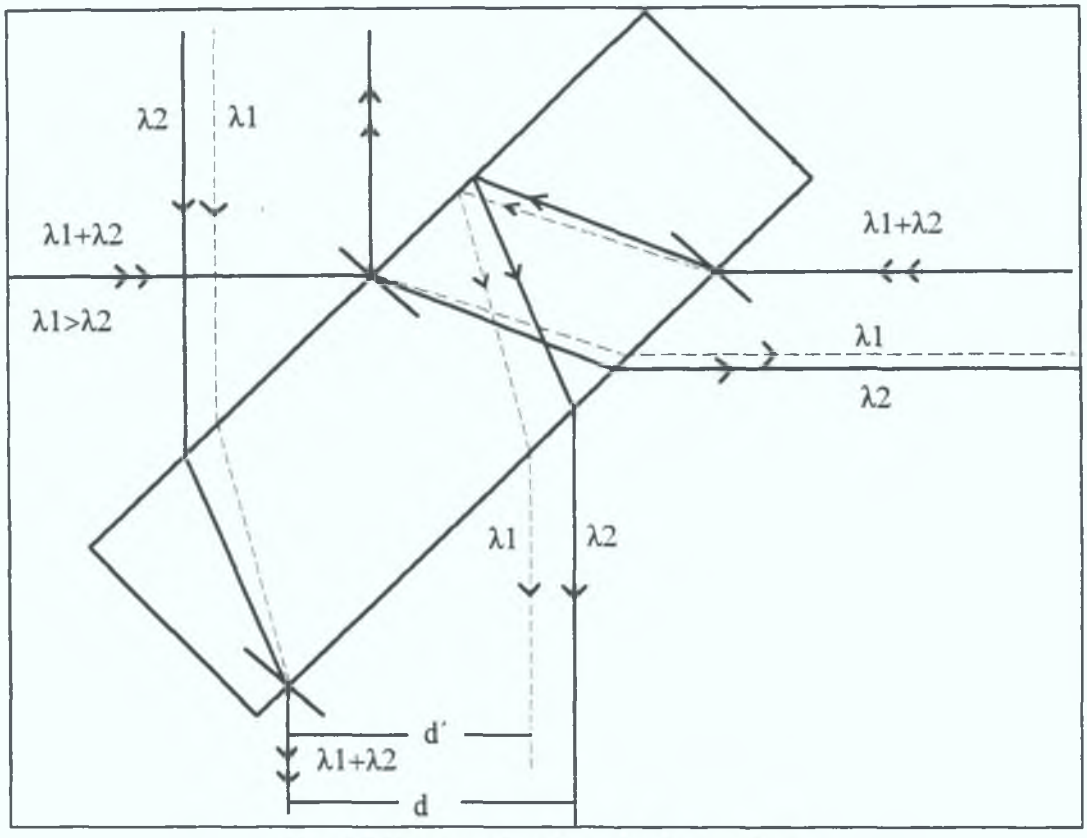


Fig. 4.7 Rays passing through a beamsplitter

The solid line represents the path taken by a ray of light at the calibration wavelength λ2. It emerges from the beamsplitter spatially seperated by a small distance d. The dashed line is light of a different, longer, wavelength λ1. If the beamsplitter exhibits normal dispersion it means that this light is seperated from its original path on refraction from the beamsplitter. The portion that travels anti-clockwise in the interferometer leaves the beamsplitter following the path taken by the light of calibration wavelength λ2. The other part, however, is deviated so that the beam seperation is smaller for greater wavelengths and vice versa. As a result, the red lines are shifted to a longer wavelength whereas lines of a shorter wavelength move towards the centre of the graph. In order to make corrections for this effect the dispersion curve n(λ) needs to be known. It would be used in the discrete Fourier transform to find the true value for the optical path difference as a function of wavelength.

4.2.2 Phase Correction

The phase correction described in chapter 2 as well as the apodization functions were applied to the interferograms before the discrete Fourier transform. Although there was a residual asymmetry in most of the spectra obtained, the phase correction was working effectively. In order to illustrate this, a cadmium spectrum

was taken with (figure 4.8) and without (figure 4.9) phase correction using the Savart plate interferometer.

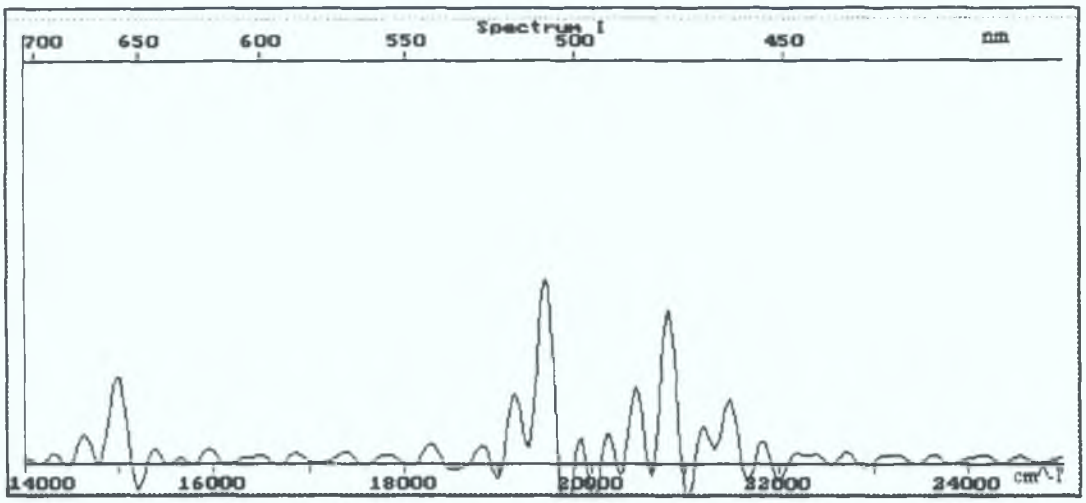


Fig. 4.8 Cadmium spectrum without phase correction

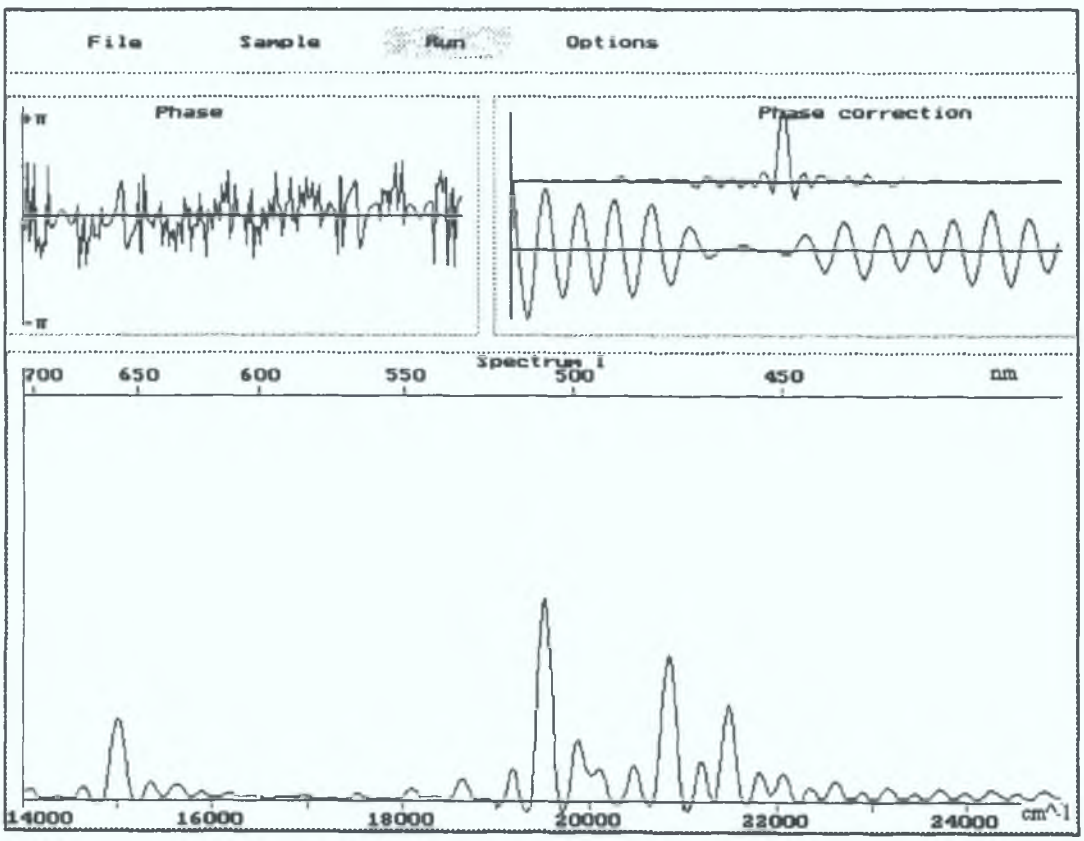


Fig. 4.9 Cadmium spectrum with phase correction

In comparison with graph 4.9 the first spectrum exhibits significant undershoots as well as a slight shift in wavelength. This is in correspondence with the theory which states that the spectrum reconstructed from an unsymmetrical interferogram loses its symmetry itself.

The greater linewidth in the two graphs above is due to the reduced beam separation of the Savart plate described in chapter 3 The relative size of the side lobes has also increased to the extend that they could easily be mistaken for spectral features

The two graphs labelled "phase" and "phase correction" indicate if the asymmetry in the original interferogram is too strong so it cannot be reliably corrected If the phase spectrum $\phi(\sigma)$ varies only slowly over the range of wavelengths present in the light source, the low resolution used for its determination is sufficient If the phase spectrum shows sharp peaks and discontinuities it can be assumed that the information contained in $\phi(\sigma)$ is not complete

4.2.3 Apodization

The software also provided a choice of three apodization functions The following graphs shows each one of them applied to the sodium spectrum

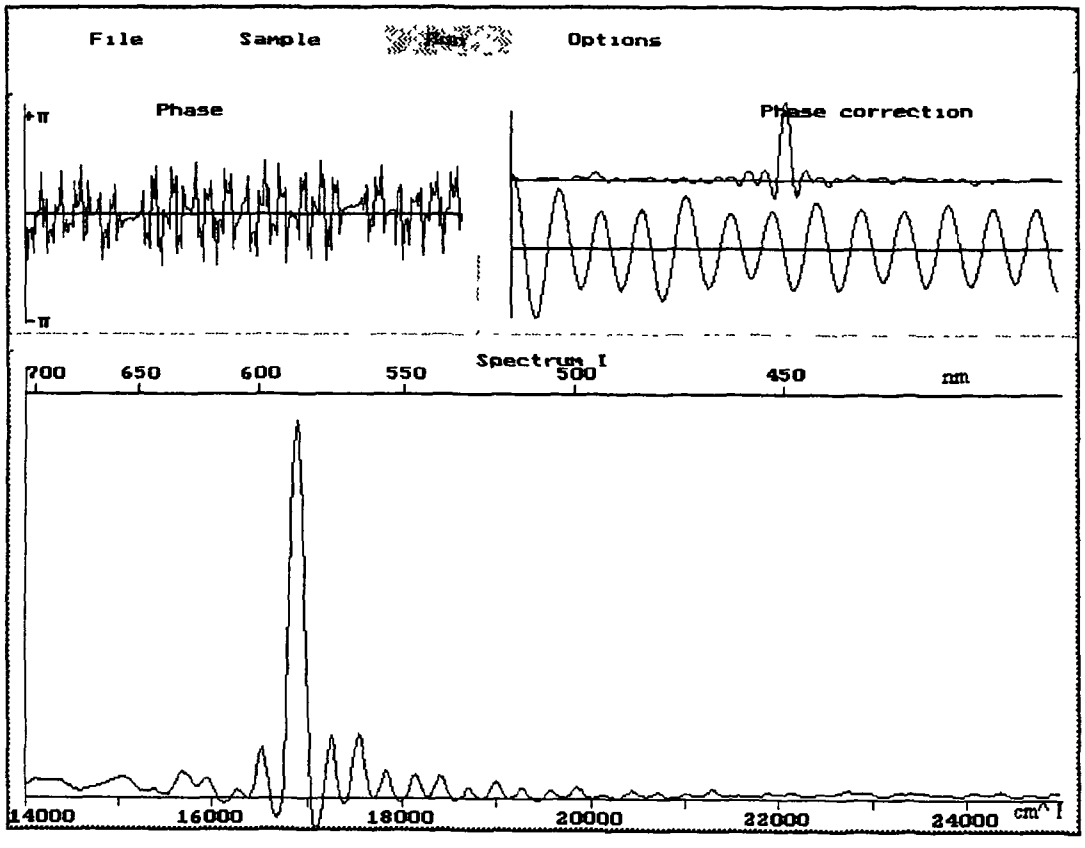


Fig 4.10 Sodium spectrum apodized with eq 4 1

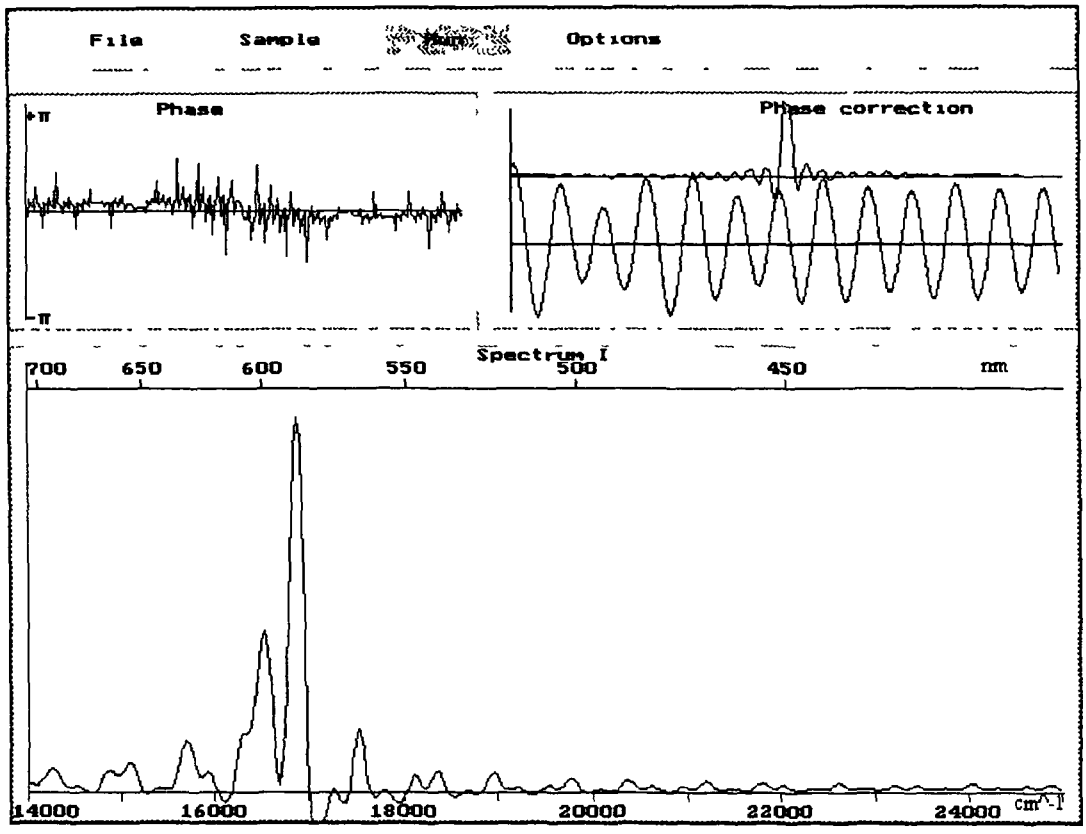


Fig 4.11 Sodium spectrum apodized with eq 4 2

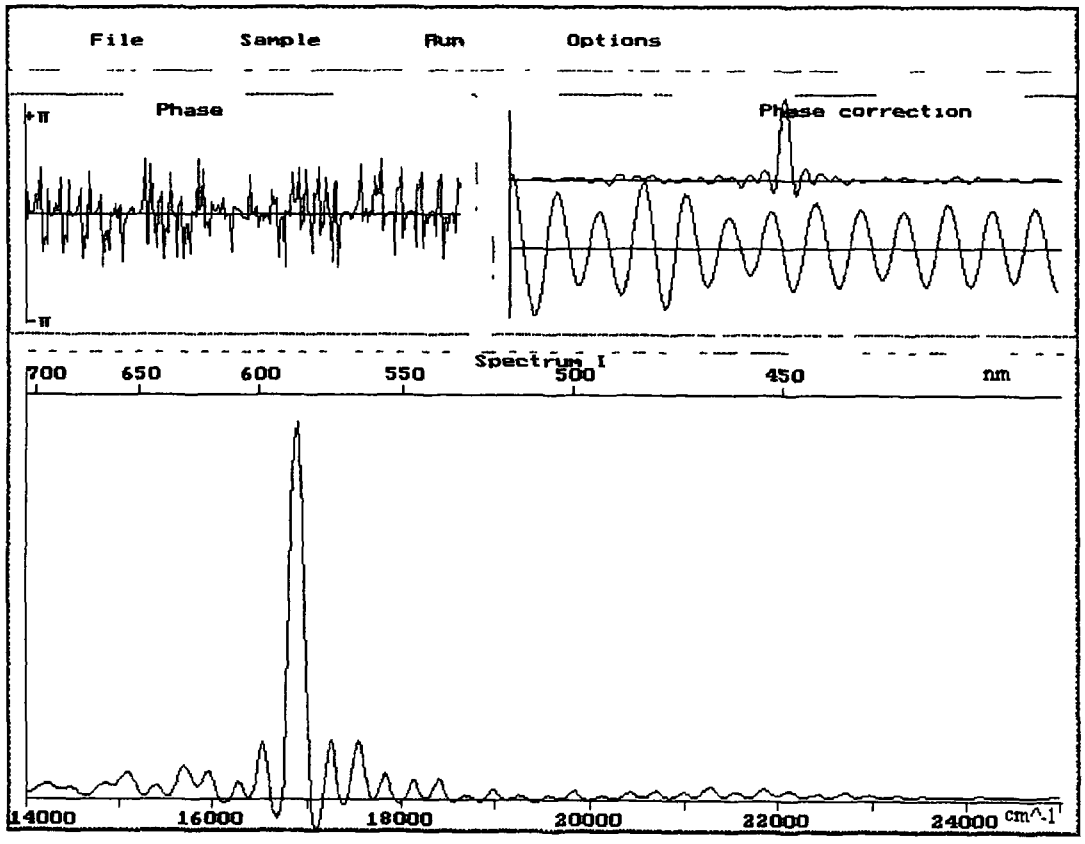


Fig 4.12 Sodium spectrum apodized with eq. 4.3

There follows a list of the apodization functions available from the 'options' menu:

- $f(x) = \left|1 - \frac{x}{D}\right|$ (4.1)
- $f(x) = \left|1 - \left(\frac{x}{D}\right)^2\right|$ (4.2)
- $f(x) = \left(1 - \left(\frac{x}{D}\right)^2\right)^2$ (4.3)

The apodization functions 4.1 and 4.3 gave the best results for both instruments. The first function produced a slightly smaller instrumental linewidth, whereas the latter provided a slightly better sidelobe suppression. Both equations, 4.1 and 4.3, were used in the tests.

4.2.4 Double Sided Interferograms

The software also featured a double sided interferogram mode. It dispenses with the phase correction and hence decreases the computing time considerably, at the expense of a lower resolution. In this mode, the point of zero path difference in the interferogram lies about in the centre of the photodiode array. Again, the 'continuous sampling' option can be used to perform the necessary alignment procedures. The next graph shows the interferogram produced by the mercury source and the triangular interferometer after linear apodization.

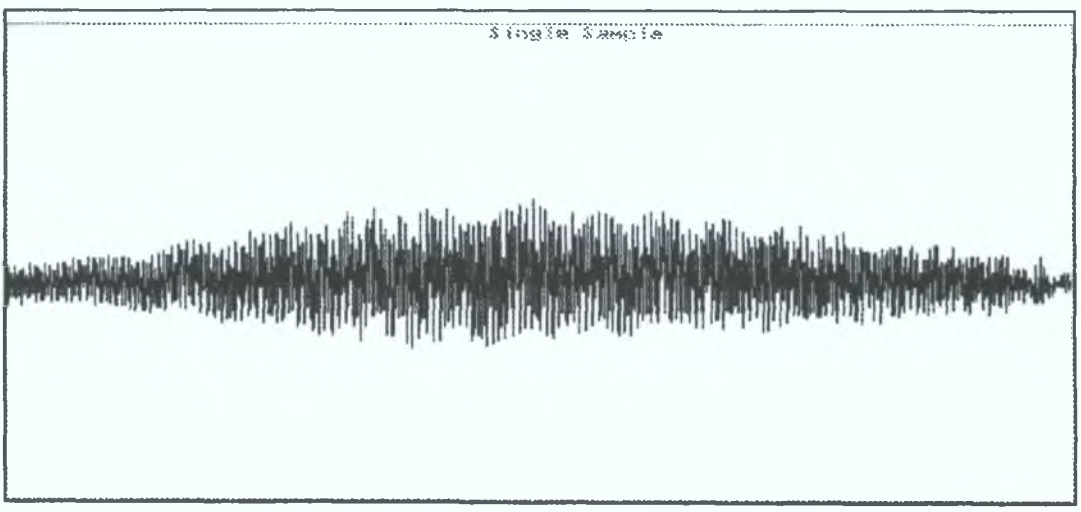


Fig. 4.13 Double sided interferogram from the mercury source after apodization

The diagram below (figure 4.14) is the spectrum reconstructed from the interferogram shown above. It features much lower resolution and has more

distinguished side lobes than the spectrum obtained from the single sided interferogram A short description of the processing of double sided interferograms is given in appendix A

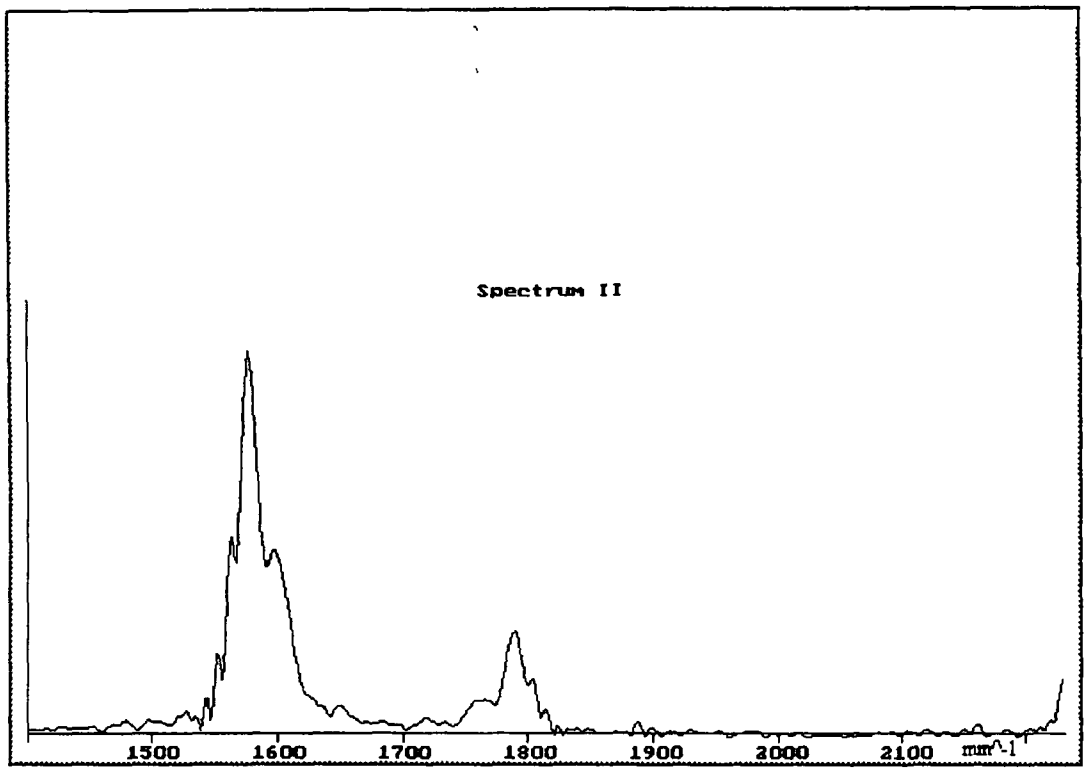


Fig. 4.14 Mercury spectrum reconstructed
from a double sided interferogram

4.2.5 Background Subtraction

A non-interference background radiation signal is superimposed on any interferogram produced by a Fourier transform spectrometer (see also chapter 2) Furthermore, this background signal is usually not uniform but rather distorted by the imperfections of the instrument optics Hence, it generates additional noise and diminishes the dynamic range of the spectrometer

The software stores the previously recorded background signal and subtracts it mathematically from the original interferograms In case of the Savart plate spectrometer the efficiency of this method can be enhanced Figure 4 15 shows an interferogram from the mercury lamp with the axis of the analyzer antiparallel to that of the polarizer

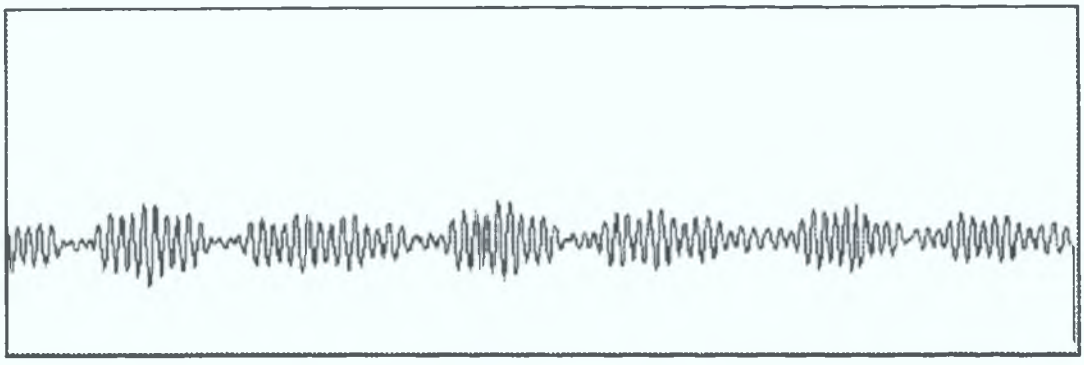


Fig. 4.15 Anti-phase interferogram of the mercury source

This arrangement exhibits a negative peak at the zero path difference position while the normal arrangement shows a positive peak. There is a non interference background distribution in both cases. This background radiation can now be removed by subtracting the anti-phase interferogram from the in-phase one. The interference component becomes twice as large as a result of this subtraction.

4.3 Discussion

The Savart plate instrument provided easier optical alignment than the triangular spectrometer. It also has better mechanical stability and it usually does not require readjustment during operation -properties that are very important for a small portable instrument. Furthermore, it has a signal to noise ratio at least twice as large due to the more efficient background subtraction. The next graphs are the remaining spectra taken from a cadmium and the sodium lamp and also from a red colour LED.

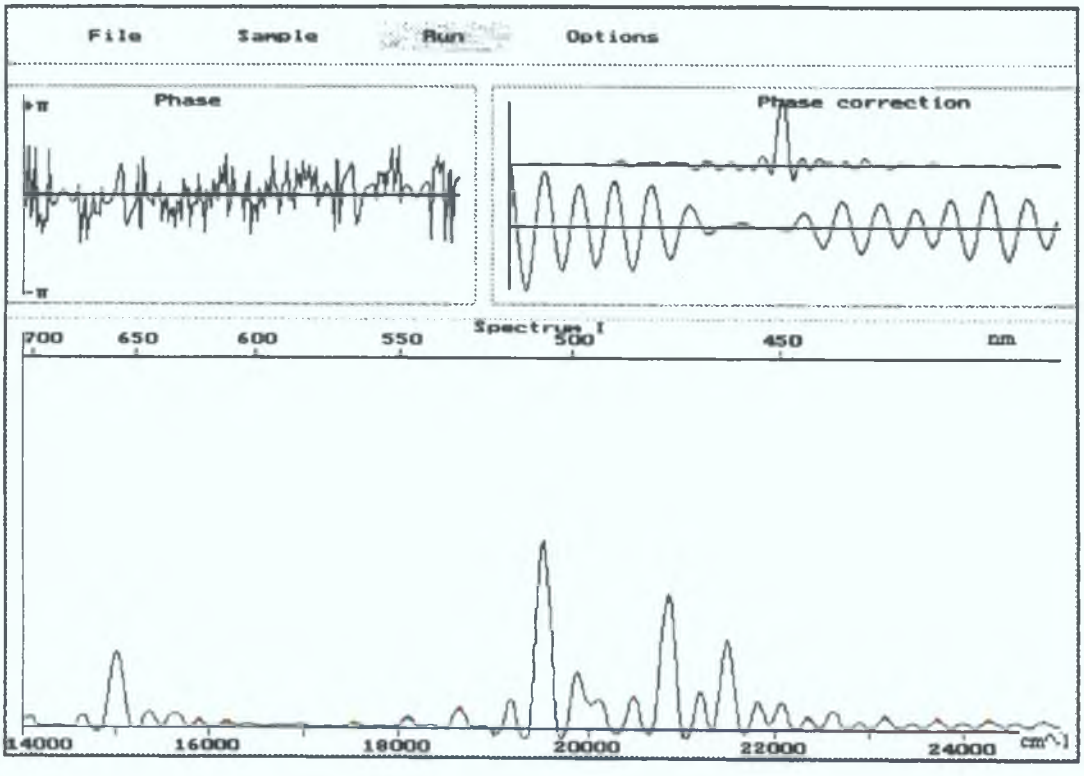


Fig. 4.16 Spectrum of the cadmium lamp

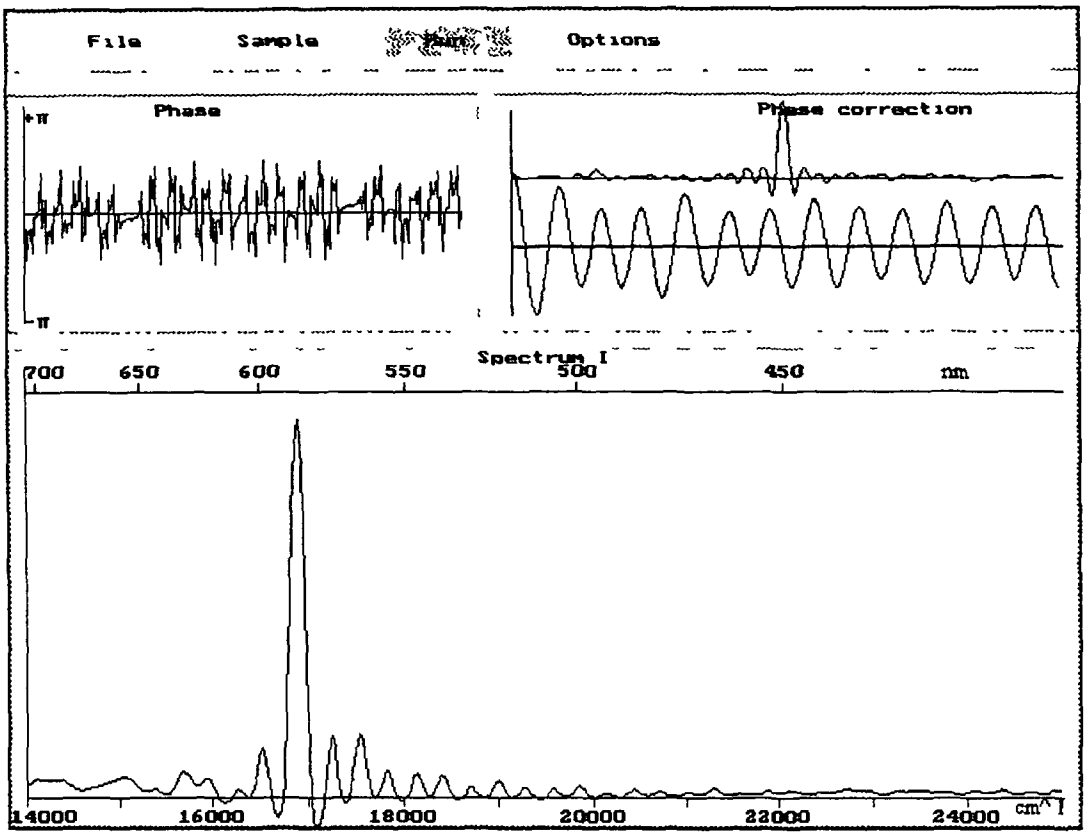


Fig. 4.17 Spectrum of the sodium lamp

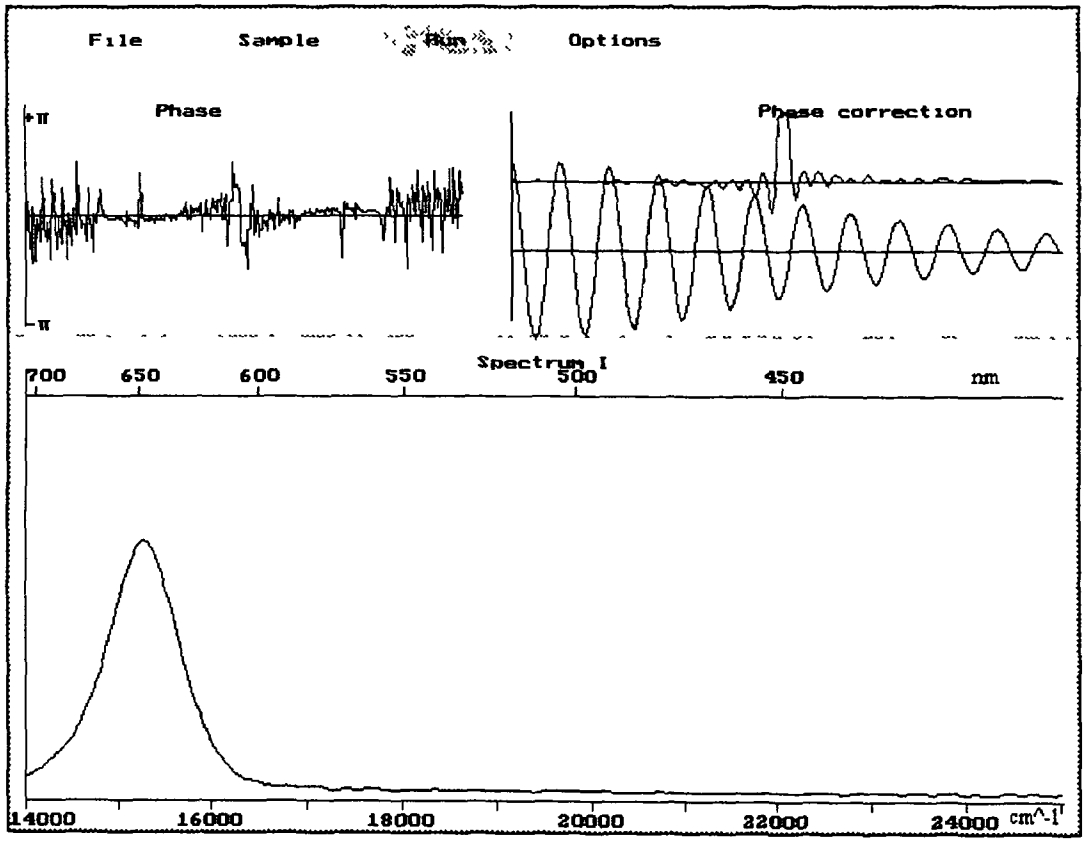


Fig. 4.18 Spectrum of a red colour LED

The large linewith is due to the reduced beam seperation of the Savart plate explained in chapter 3 which also leads to an increased size of the side lobes They have increased to the extent that they could easily be mistaken for spectral features In a practical instrument a resolution this low would not be sufficient and would have to be at least as large as that of the triangular spectrometer

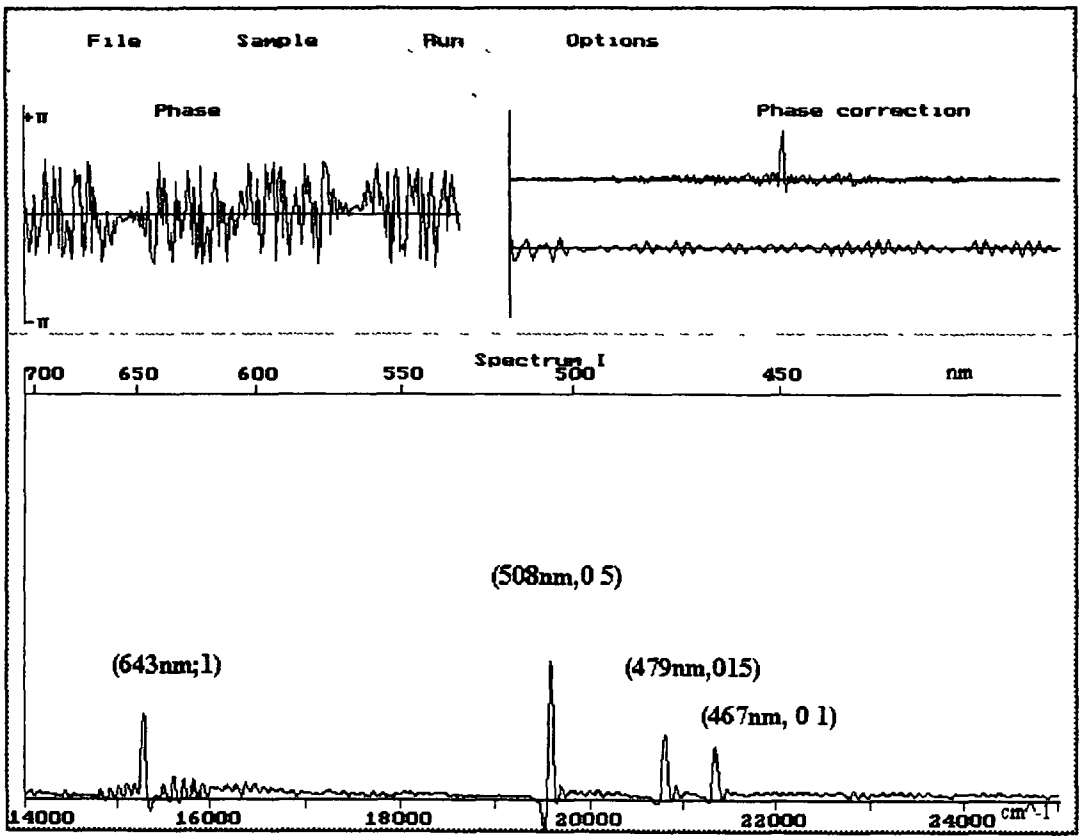


Fig. 4.19 Spectrum of the cadmium lamp

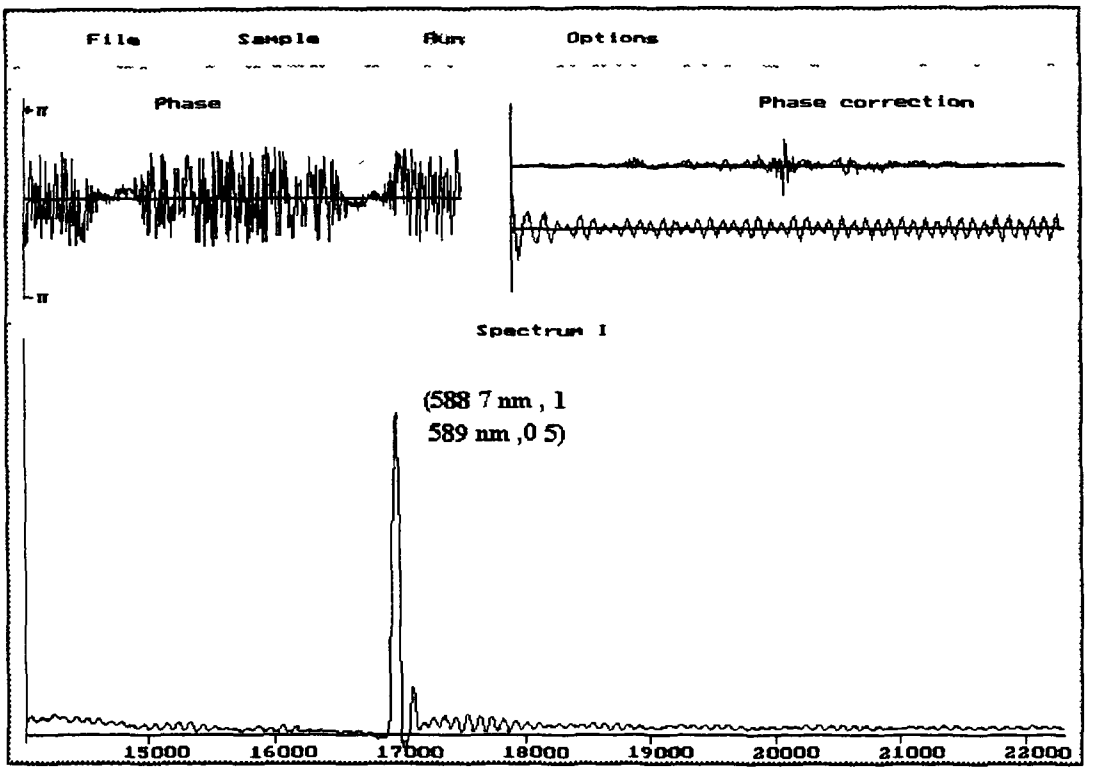


Fig4.20 Spectrum of the sodium lamp

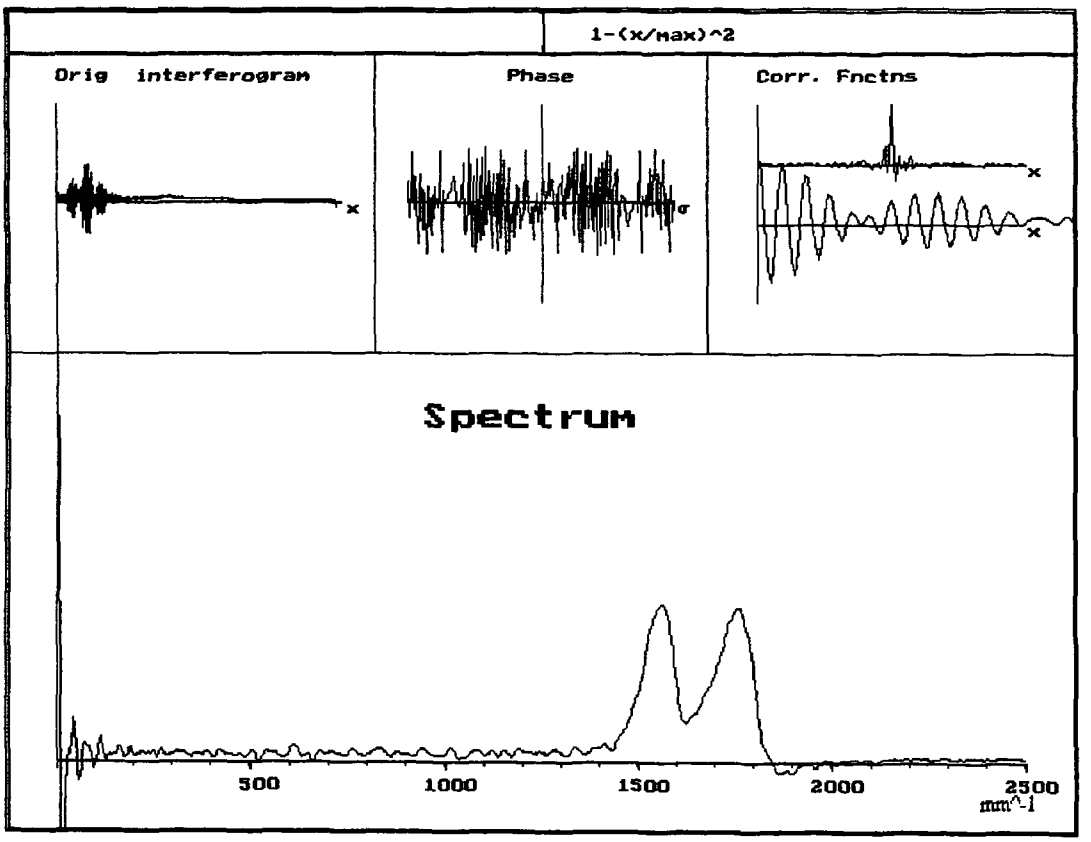


Fig. 4.21 Spectrum of a tri-colour LED It emits light in two regions roughly from 530nm to600nm and from 600nm to 680nm

The seperation between the two virtual sources was 2mm, hence the theoretical linewidth at half maximum was 1 9nm at 540nm This agrees closely with the measured value of $(2 \pm 0\ 5)$nm which was taken from a plot of the mercury spectrum

The costs are roughly the same for both instruments The requirements for a high flatness and a large size of the beamsplitter and the Savart plate result in a high price for these components The beamsplitter should also have a nearly constant reflection to transmission ratio over the wavelength range of interest Compared to diffraction grating instruments, however, both designs are low in cost as long as low resolution is sufficient Unlike grating instruments, the resolution of stationary Fourier transform spectrometers can only be improved by increasing the number of elements in the detector, for a given spectral range Good quality linear photodiode arrays having more than 4096 elements are still expensive, although the prices for these devices are expected to decline in the future

Chapter 5

Conclusion

Two different types of stationary FT-spectrometers were successfully built and tested. Both instruments exhibited a high optical throughput as there was no need for entrance or exit slits. The relatively low resolution of the Savart- plate instrument could easily be improved by using a different lens - crystal combination. This could also lead to a design which is largely reduced in size. In fact, a group of researchers from the University of St. Andrews (M.J. Padgett, A.R. Harvey, 1995), Scotland, have developed an instrument of the size of a golf ball using a customised Savart - plate and no collimating lens. A Scottish company has already commercialised this new design.

In general, the Savart plate set-up is superior to the triangular one in terms of robustness, ease of alignment and size. The problem of finding the position of zero path difference in the interferogram was solved by means of a bifurcated optical fibre which was used to couple a small portion of white light into the probe beam. This way, in combination with the appropriate phase correction procedure, single sided interferograms may be used to reconstruct the spectrum, yielding a higher resolution.

The dynamic range of both instruments is determined by that of the photodiode array which was approximately 40 dB (EG&G Reticon, RL 1024S Instruction Manual), while the typical value for detectors used in Fourier transform spectroscopy is 90 dB. The dynamic range of the photodiode array may be used more efficiently by the technique of chirping (L.Mertz, 1965). By inserting a dispersive element into the optical path behind the collimating lens, the phase of the central fringes varies with the wavelength. Thus the energy of the interferograms is widely distributed spatially.

In order to make the two instruments more useful for "every-day" spectroscopic measurements some features would have to be added to the software package. The implementation of options for saving the raw data and spectra are necessary. Also, a routine would have to be written that compensates for the spectral response of the detector and the transmission and reflection functions of the optical components

employed Comparison of the measured intensities with the actual ones in the last chapter shows that the recorded intensities are highly wavelength dependent

Finally, it should be mentioned here that the use of fixed optics Fourier transform spectrometers is not limited to the visible spectrum of light M Hashimoto *et al* (1992) developed an instrument for mid-IR measurements based on the Savart-plate set-up The polarisers were aluminium wire grids on a CaF_2 base and the lenses were also made of CaF_2 The detector was a 4096 element Schottky barrier type of platinum silicide with a spectral range covering ~1 3 to 5 μ m The entire set up could be mounted in a cryostat to reduce noise and background radiation

In the XUV to soft x-ray region, the required high quality of the optical components in an interferometer often impose large difficulties on the designer of such a system Interferometry requires a much better waveform than does imagery A french group (J Svatos, D Joyeux, D Phallipou, F Polack, 1993) therefore designed a wavefront division interferometer based on Fresnel's mirrors to produce an interferogram from x-rays The two mirrors made of uncoated silica were tilted at an angle of 2'14'' with respect to each other and the grazing incidence angle on the mirrors was set to 3° to 6° to ensure good reflectivity For their experiment the researchers used wideband synchrotron radiation to produce an interferogram which they recorded by means of a high resolution photographic plate Later, the photographic record of the fringe pattern was digitised and Fourier transformed

Clearly, it would be desirable to dispense with the last two steps and to employ an array detector to pick up the signal CCD's are very good detectors of ionising radiation but unfortunately the MOS shift register structures of most CCD's are sensitive to radiation damage Sol M Gruner describes in his article (1989) a number of solutions to this problem He reports that inorganic phosphors, such as rare-earth oxysulfides are typically used for the conversion of x-rays into light, because these materials have a high x-ray stopping power and a high efficiency Then, the visible light is coupled to a CCD by either lenses or fibreoptics where the latter have much better efficiencies Alternatively, phosphor grains can be settled directly into the deeply etched out cores of a fibre optic plate

Appendix A

If the interferogram is centred on the detector, one can make allowance for the asymmetry in the interferogram without using the phase correction algorithms described in chapter 2

Consider equation 2 42, its inverse Fourier transform yields

$$\int_{-\infty}^{\infty} F_{asy}(x)\exp(-2\pi i\sigma x)\,dx = B(\sigma)\exp(-\phi(\sigma)) = p(\sigma) - iq(\sigma)$$

where

$$p(s) = \int_{-\infty}^{\infty} F_{asy}(x)\cos(2\pi\sigma x)\,dx$$

and

$$q(x) = \int_{-\infty}^{\infty} F_{asy}(x)\sin(2\pi\sigma x)\,dx$$

The required spectrum is therefore given by

$$B_e(\sigma) = \left([p(\sigma)]^2 + [q(\sigma)]^2\right)^{\frac{1}{2}}$$

with no need to know $\phi(\sigma)$

Appendix B

```
#include <stdio.h>
#include <conio.h>
#include <graphics.h>
#include <stdlib.h>
#include <math.h>

# define  CLIP_ON 1


  extern void arrayplot( );
  extern void readout();
  void options();
  float nummerumwand();


  int  intgrm1[1025], intgrm2[1025];
  float  acc[1025],intgrm3[1025],intgrm4[1100];
  char menu_code='0';
  char y,intfgrm_arr_full='n';
  unsigned short int number_of_scans=1, apodization=0, correction_
func=1;
  char backgnd_sub='n', interferogr='t';
  float factor=0.0, max1;
  char menupoint1[6], menupoint2[15], menupoint3[5], menupoint4[7]
;
  short int menu_pos;



  void  setgraphics(void)     {
        int  g_driver, g_mode, errorcode;

        clrscr();
        detectgraph(&g_driver, &g_mode);
        initgraph(&g_driver, &g_mode, "..\\bgi");
        errorcode = graphresult();
         if(errorcode != grOk) {
          printf("Graphic error: %s",grapherrormsg(errorcode));
          getch();
                                }
         return;
                              }


  int windowsetup(void)       {

        clearviewport();
        setcolor(3);
        rectangle(0, 50, 280, 190);
        rectangle(290, 50, 639, 190);
        rectangle(0, 200, 639, 479);
        setfillstyle(1, 1);
```

```
        bar3d(0, 0, 639, 36, 0, 0);
        setcolor(14);
        outtextxy(50, 15,menupoint1);
        outtextxy(140, 15,menupoint2);
        outtextxy(250, 15,menupoint3),
        outtextxy(335, 15,menupoint4);
      return;
                              }

  short int  menucontrol(void)      {
        char menu_open,b;



    /*  setfillstyle(1, 4);     */
    /*  bar(25, 10, 100, 27);   */
        setcolor(14);
        outtextxy(50, 15,menupoint1);
          while(((b=getch()) != '\r')&&(b != 80))   {
           switch(b)    {
              case 75:if (menu_pos == 0)   {
                        setfillstyle(1, 1);
                        bar( 25, 10, 100, 27);
                        outtextxy(50, 15,menupoint1);
                        setfillstyle(1, 4);
                        bar( 325, 10, 400, 27);
                        outtextxy( 335, 15,menupoint4);
                        menu_pos = 3;
                                              }
                      else                    {

                        setfillstyle(1, 1);
                        bar(((menu_pos*100)+25), 10, ((menu_pos*1
00)+100), 27);
                         switch(menu_pos)    {


                              case 1 :    outtextxy(140, 15,menu
point2);
                                          setfillstyle(1, 4);
                                          bar(25, 10, 100, 27);
                                          outtextxy(50, 15,menu
point1);
                                          break;

                              case 2 :    outtextxy(250, 15,menu
point3);
                                          setfillstyle(1, 4);
                                          bar(125, 10, 200, 27)
;
                                          outtextxy(140, 15,men
upoint2);
```

```
                                                    break;

                                    case 3 :    outtextxy(335, 15,menu
point4);
                                                    setfillstyle(1,4);
                                                    bar(225, 10, 300, 27)
;
                                                    outtextxy(250, 15,men
upoint3);
                                                    break;
                                                    }

                                menu_pos--;

                                                  }      /*"else" closed*
/
                                 break;
                        case 77 :if(menu_pos==3)      {

                                setfillstyle(1, 1);
                                bar(325, 10, 400, 27);
                                outtextxy(335, 15,menupoint4);
                                setfillstyle(1, 4);
                                bar(25, 10, 100, 27);
                                outtextxy(50, 15,menupoint1);
                                menu_pos=0;
                                                    }

                               else                 {
                                setfillstyle(1, 1);
                                bar(((menu_pos*100)+25), 10,((menu_pos*10
0)+100), 27);
                                 switch(menu_pos)    {
                                   case 0:  outtextxy(50, 15,menupoint1);
                                            setfillstyle(1, 4);
                                            bar(125, 10, 200, 27);
                                            outtextxy(140,15,menupoint2);
                                            break;

                                   case 1:  outtextxy(140, 15,menupoint2)
;
                                            setfillstyle(1, 4);
                                            bar(225,10, 300, 27);
                                            outtextxy( 250, 15,menupoint3
);
                                            break;

                                   case 2:  outtextxy(250, 15,menupoint3)
;
                                            setfillstyle(1, 4);
                                            bar(325, 10, 400, 27);
                                            outtextxy(335, 15,menupoint4)
```

```
;
                                        break;

                                                }       /*"switch2"
 closed*/
                                    menu_pos++;
                                         }          /*"else2" closed*
/
                                    break;

                                  }     }                   /*"switch"
 + "while" closed*/

         return(menu_pos);
                                  }
  char menu0(void)                {


            void *popmenu;
            unsigned int size=0;
                     char menu_num;
                     int left=25, top=36, right=150, bottom=100;

            size=imagesize(left,top,right,bottom);
            popmenu = malloc(size);
            getimage(left, top, right, bottom, popmenu);
            setfillstyle(1, 1);
            bar3d(left, top, right, bottom, 0, 0);
             outtextxy(30, 45,"1. Exit");
             while(((menu_num=getch())!='\r')&&(menu_num!='1')&&(m
enu_num!=27));

            putimage(left, top, popmenu,COPY_PUT);


            free(popmenu);
                      return(menu_num);
                                }


  char menu1(void)              {
```

```
            void *popmenu;
            unsigned int size;
            char menu_num='0';
                     int left=125, top=36, right=350, bottom=130;

            size=imagesize(left,top,right,bottom);
            popmenu = malloc(size);
            getimage(left, top, right, bottom, popmenu);
            setfillstyle(1, 1);
            bar3d(left, top, right, bottom, 0, 0);
             outtextxy(130, 45,"1. Cont. Sampling");
             outtextxy(130, 65,"2. Single Shot");
             outtextxy(130, 85,"3. Sampling + Averag.");
             outtextxy(130, 105,"4. Record Backgr.");

              while(((menu_num=getch()) != ('\r'))&&(menu_num!='1'
)&&(menu_num!='2')
              &&(menu_num!='3')&&(menu_num!='4')&&(menu_num!=27));
            putimage(left, top, popmenu,COPY_PUT);


            free(popmenu);
            return(menu_num);
                                    }



  void errormssge(void)                    {

       void *popmenu;
       unsigned int size=0;
       unsigned int left=220, top=190, right=420, bottom=290;


            size=imagesize(left,top,right,bottom);
            popmenu = malloc(size);
            getimage(left, top, right, bottom, popmenu);
            setfillstyle(1, 4);
            bar3d(left, top, right, bottom, 0, 0);
            outtextxy(270, 220,"No Signals!");
            outtextxy(260, 240,"Press any key!");
             getch();
            putimage(left, top, popmenu,COPY_PUT);

             free(popmenu);
             return;
                                     }
```

```
short int signaltest(void)              {
          short int flaga=0, flagb=0, flagc=0, flagaa=0;
          unsigned int n;

           outportb(647, 144);
            for(n=0; n<=20000; n++) {
              if(inportb(644)&1)
               flaga=1;              }

            for(n=0; n<=20000; n++) {
              if('(inportb(644)&1))
               flagb=1;              }

            for(n=0; n<=20000; n++) {
              if(inportb(644)&4)
               flagc=1;              }

           if(flaga && flagb && flagc){
               flagaa=1;               }
            return(flagaa);

                                      }

void setoptwin(void)                  {

     unsigned int left=0, top=50, right=639, bottom=190;
     unsigned short int n=6;
     char a;


           setcolor(14);
           setfillstyle(1, 1);
           bar3d(left, top, right, bottom, 0,0);
           outtextxy(290, 55,"Options");
           moveto(0, 70);
           lineto(639, 70);
           moveto(0, 75);
           lineto(639, 75);
           moveto(0, 100);
           lineto(639,100);
           moveto(0, 125);
           lineto(639, 125);
           moveto(0, 150);
           lineto(639, 150);
           moveto(320, 75);
           lineto(320, 190);
            outtextxy(20, 83,"Number of scans:");
            outtextxy(20, 108,"Backgrnd subtr.:");
            outtextxy(20, 133,"Interferogram :");
            outtextxy(340, 83,"Apodization Fnc.:");
            outtextxy(340, 108,"Correction proc.:");
            outtextxy(340, 133,"Opt.P.D factor :");
```

```
            options();
             a=getch();

            while(a!=27)       {



            if(a==9) {
                              if(n==6)
                                    n=1;
                              else
                                    n++;

       switch(n)                {

           case 1:      setfillstyle(1, 1);
                        bar(480, 130, 600, 145);
                        setfillstyle(1, 3);
                        bar(160, 80, 280, 95);
                        if((a=getch())=='\r')
                        number_of_scans=(int)nummerumwand(180, 83)
;
                        else   {
                        setfillstyle(1, 1);
                        bar(160, 80, 280, 95);}
                        options();
                        break;

           case 2:      setfillstyle(1, 3);
                        bar(160, 105, 280, 120);
                        options();
                        while(((a=getch())!='\r')&&(a!=9)&&(a!=27)
)      {

                        if (((a==72)&&(backgnd_sub=='y'))||
                              ((a==80)&&(backgnd_sub=='y')))
                                          backgnd_sub='n';
                        else {
                           if (((a==72)&&(backgnd_sub=='n'))||
                              ((a==80)&&(backgnd_sub=='n')))
                                          backgnd_sub='y'; }

                             bar(160, 105, 280, 120);
                             options();


   }
                                          break;

                             case 3:      setfillstyle(1, 1);
                                          bar(160, 105, 280, 120);
                                          setfillstyle(1, 3);
```

```
                                        bar(160, 130, 280, 145);
                                        options();

                                        while((a=getch())!='\r')&
&(a!=9)&&(a!=27))   {
                                         if((a==72)&&(interferogr
==t'))||
                                            ((a==80)&&(interferogr
==t')))
                                                              interf
erogr='o';
                                         else    {
                                           if((a==72)&&(interfero
gr=='o'))||
                                              ((a==80)&&(interfero
gr=='o')))
                                                              interf
erogr='t';}
                                        bar(160, 130, 280, 145);
                                        options();
                                                               }
                                        break;

                                case 4: setfillstyle(1,1);
                                        bar(160, 130, 280, 145);
                                        setfillstyle(1, 3);
                                        bar(480, 80, 600, 95);
                                        options();
                                        while((a=getch())!='\r')&
&(a!=9)&&(a!=27))   {
                                         if(a==80)  {
                                            if(apodization==3)
                                                    apodization=0;
                                            else
                                                    apodization++;
                                         bar(480, 80, 600, 95);
                                          options();
                                                    }
                                                      }
                                        break;

                                case 5: setfillstyle(1, 1);
                                        bar(480, 80, 600, 95);
                                        setfillstyle(1,3);
                                        bar(480, 105, 600, 120);
                                        options();
                                        a=getch();
                                        break;

                                case 6: setfillstyle(1, 1);
                                        bar(480, 105, 600, 120);
                                        setfillstyle(1, 3);
```

```
                                        bar(480, 130, 600, 145);
                                        options();
                                        if((a=getch())=='\r')
                                        factor=nummerumwand(480, 1
33);
                                        a=getch();
                                        break;

                                           }}
                                  else
                                      a=getch();

                                    options();
                                           }


                  return;                  }

  float nummerumwand(int startx, int starty)      {
        float result=0.0;
        char c;
        int nn=0,decpnt=0;

        moveto(startx, starty);
        outtext("_");
        while((nn<=12)&&((c=getch())'='\r'))        {
             switch(c)       {
                case '0':   outtext("0");
                            result=result*10;
                            nn++;
                            break;
                case '1':   outtext("1");
                            result=(result*10)+1;
                            nn++;
                            break;
                case '2':   outtext("2");
                            result=(result*10)+2;
                            nn++;
                            break;
                case '3':   outtext("3");
                            result=(result*10)+3;
                            nn++;
                            break;
                case '4':   outtext("4");
                            result=(result*10)+4;
                            nn++;
                            break;
                case '5':   outtext("5");
                            result=(result*10)+5;
                            nn++;
                            break;
                case '6':   outtext("6");
```

```
                        result=(result*10)+6;
                        nn++;
                        break;
                case '7':   outtext("7");
                        result=(result*10)+7;
                        nn++;
                        break;
                case '8':   outtext("8");
                        result=(result*10)+8;
                        nn++;
                        break;
                case '9':   outtext("9");
                        result=(result*10)+9;
                        nn++;
                        break;
                case '.':   if(decpnt==0)  {
                        outtext(".");
                        decpnt=nn;     }
                        break;
                case '\b':  if(nn!=decpnt)   {
                        result=floor(result/10);
                        nn--;
                                        }
                        else            {
                          decpnt=0;     }
                          moverel(-8,0);
                          bar(getx(), gety(), getx()+8, gety()
+12);
                          break;

                        } }
                if((decpnt!=nn)&&(decpnt!=0))
                    result=result/(pow10(nn-decpnt));
                    return(result);
                                }


  void options(void)                    {
       char apo1[17], apo2[20], apo3[20];

          sprintf(apo1,"1-(abs(x)/max)");
          sprintf(apo2,"1-(x/max)^2");
          sprintf(apo3,"(1-(x/max)^2)^2");

                  switch(backgnd_sub)  {
                     case 'n':   outtextxy(180, 108, "No");
                                  break;
                     case 'y':   outtextxy(180, 108, "Yes");
                                  break;
                                      }
                  switch(interferogr)  {
                     case 't':   outtextxy(180, 133,"Two-sided");
```

```
                                    break;
                        case 'o':   outtextxy(180, 133,"One-sided");
                                    break;
                                         }

                    switch(apodization)  {
                        case 1:     outtextxy(480, 83, apo1);
                                    break;
                        case 2:     outtextxy(480, 83, apo2);
                                    break;
                        case 3:     outtextxy(480, 83, apo3);
                                    break;
                                         }
                     return;
                                                          }

void plot(aa)
     int aa[1024];     {
     unsigned int n, xx;
     float y;

         moveto( 0, 475);
         setcolor(14);
         for(n=0; n<=1023; n++)    {
               xx=n*0.626;
               y=475-aa[n]*0.06;
              lineto(xx, y);
                                   }
           return;
           }


    void plotint(bb)

     int bb[1024];              {
     unsigned int n, xx;
     float y;

         moveto( 0, 475);
         setcolor(14);
         for(n=0; n<=1023; n++)    {
               xx=n*0.626;
               y=350-bb[n]*0.06;
              lineto(xx, y);
                                   }
           return;
           }

void apo(float maxx)  {
    float n;

    if(apodization==1)  {
```

```
          for(n=0; n<maxx; n++)                    {
             intgrm3[n]=intgrm3[n]*(1-((maxx-n)/maxx));   }
          for(n=maxx; n<2*maxx; n++)                {
             intgrm3[n]=intgrm3[n]*(1-(n-maxx)/(maxx));   }
                          }
      if(apodization==2)   {
          for(n=0; n<maxx; n++)                      {
             intgrm3[n]=intgrm3[n]*(1-pow(((maxx-n)/maxx),2)); }
          for(n=maxx; n<=2*maxx; n++)                   {
             intgrm3[n]=intgrm3[n]*(1-pow(((n-maxx)/(maxx)),2)); }
                          }
      if(apodization==3)   {
          for(n=0; n<=maxx; n++)                     {
             intgrm3[n]=intgrm3[n]*pow((1-pow(((maxx-n)/maxx),2)),
2); }
          for(n=maxx; n<2*maxx; n++)                  {
             intgrm3[n]=intgrm3[n]*pow((1-pow(((n-maxx)/(maxx)),2)
),2); }
                    }
                return;
                }

  void twosidedft(void)          {
       float psig=0.0, qsig=0.0;
       float deltasigma,o,n1,m,a;
       int n,nn=0;

         setfillstyle(1, 0);
        setcolor(3);
         bar3d(0, 200, 639, 479, 0, 0);
        setcolor(14);
        outtextxy(280,202,"Spectrum II ");
        moveto(10, 210);
        lineto(10, 462);
        lineto(630, 462);
         for(n=0; n<=8; n++)    {
          moveto(((n*74.69)+10),462);
          linerel(0, 5);      }

         outtextxy(70,470,"15000");
         outtextxy(145,470,"16000");
         outtextxy(220,470,"17000");
         outtextxy(295,470,"18000");
         outtextxy(370, 470,"19000");
         outtextxy(445,470,"20000");
         outtextxy(520,470,"21000");
         outtextxy(595,470,"22000");
         moveto(10,462);

        deltasigma=0.2/(max1*factor);     /*=unapodized resolution
 */
        o=(0.563636*deltasigma);
```

```
          for(n1=1400; n1<=2500; n1=n1+deltasigma)    {
           qsig=0.0;
           psig=0.0;
            for(m=0; m<=2*max1; m++)      {
             qsig=qsig+(intgrm3[m]*20*sin(2*M_PI*n1*(m-max1)*factor
));
             psig=psig+(intgrm3[m]*20*cos(2*M_PI*n1*(m-max1)*factor
));
                                          }
            qsig=factor*qsig;
            psig=factor*psig;
           a=sqrt((psig*psig)+(qsig*qsig));
            lineto(10+(o*nn),(462-a));
            if(kbhit()!=0)  break;
            nn++;
                                         }

           getch();
           return;                 }

  void phasecorr(void)         {
        float m,n;
        double  psig, qsig,deltasig,pcorr,fcorr,px,pxx;
        float phasecorr[402],phase[900],pyy;
        char a;


        moveto(10,56);
        lineto(10, 184);
        moveto(10, 120);
        lineto(270, 120);
        outtextxy(10, 60,"+ã");
        outtextxy(10, 180, "-ã");


          moveto(300, 60);
          lineto(300, 180);
          moveto(630, 140);
          lineto(300,140);
          moveto(630, 100);
          lineto(300,100);


       outtextxy(90, 56,"Phase");
       outtextxy(450, 56, "Phase correction");
       moveto(10, 120);

        deltasig=2500/max1;
          for(n=0; n<=2*max1; n++)  {
           qsig=0.0;
           psig=0.0;
```

```
          a='1';
         for(m=0; m<=2*max1; m++)    {
           psig=psig+(intgrm3[m]*cos(2*M_PI*(m-max1)*(n-max1)*fac
tor*deltasig));
           qsig=qsig+(intgrm3[m]*sin(2*M_PI*(m-max1)*(n-max1)*fac
tor*deltasig));
                                      }
           if(psig!='0')              {
            if(psig<0)
             a='0';
            psig=fabs(psig);
           phase[n]=(atan2((factor*qsig),(factor*psig)))*(-1);
                  }
           else
            phase[n]=0;
            if(a=='0')  {
             a='1';
             phase[n]=phase[n]*(-1);
                        }
         px=10+((260*n)/(2*max1));
         if(kbhit()!=0) break;
         lineto(px, 120-(phase[n]*20));
                        }

        moveto(300, 100);

          for(n=0; n<=300; n++)    {
            pcorr=0.0;
             for(m=0; m<=2*max1; m++)  {
             pcorr=pcorr+(cos(phase[m]+(2*M_PI*(n-150)*(m-max1)*f
actor*deltasig)));
                                       }
            phasecorr[n]=pcorr*(1-(fabs(n-150)/150));
           pyy=100-(phasecorr[n]*0.2);
           if(pyy>50)
          lineto((300+1.1*n),pyy);
                                  }

        moveto(300,140);
         for(n=max1; n<923; n++)    {
          fcorr=0.0;
           for(m=0; m<=300; m++)      {
           fcorr=fcorr+(intgrm3[(n-150)+m]*phasecorr[(300-m)]);
                                      }
          intgrm4[(n-max1)]=fcorr*factor*40;
          pxx=(300+(2*(n-max1)));

          if(kbhit()!=0) break;
          if(pxx<630)
         lineto((pxx),140-intgrm4[(n-max1)]*0.025);
                                  }
```

```
        return,
                        }

void onesidedft(void)           {
     float deltasigma,n1,psig,o;
     float m;
     int nn=0,n;

     deltasigma=0.1/((922-max1)*factor);
     o=(0.56364*deltasigma);
     moveto(10, 462);

       setfillstyle(1, 0);
      setcolor(3);
       bar3d(0, 200, 639, 479, 0, 0);
      setcolor(14);
      outtextxy(280,202,"Spectrum I ");
      moveto(10, 210);
      lineto(10, 462);
      lineto(630, 462);
      moveto(10, 225);
      lineto(630, 225);
       moveto(16, 225);
       lineto(16, 220);
       moveto(78, 225);
       lineto(78, 220);
       moveto(150, 225);
       lineto(150, 220);
       moveto(236, 225);
       lineto(236, 220);
       moveto(338, 225);
       lineto(338, 220);
       moveto(463, 225);
       lineto(463, 220);


       for(n=0; n<=11; n++)    {
        moveto(((n*56.363)+10),462);
        linerel(0, 5);      }


       outtextxy(1,470,"14000");
       outtextxy(102,470,"16000");
       outtextxy(215,470,"18000");
       outtextxy(328,470,"20000");
       outtextxy(440, 470,"22000");
       outtextxy(553,470,"24000");
       outtextxy(11, 210,"700");
       outtextxy(68, 210,"650");
       outtextxy(140, 210,"600");
       outtextxy(226, 210,"550");
       outtextxy(328, 210,"500");
```

```
        outtextxy(453, 210,"450");

        moveto(10, 462);

        if(apodization=='1')    {
         for(n=0; n<=(922-max1); n++)
          intgrm4[n]=intgrm4[n]*(1-(n/(922-max1)));
                                }

        if(apodization=='2')    {
         for(n=0; n<=(922-max1);n++)
          intgrm4[n]=intgrm4[n]*(1-pow((n/(922-max1)),2));
                                 }

        if(apodization=='3')      {
          for(n=0; n<=(922-max1); n++)
           intgrm4[n]=intgrm4[n]*pow((1-pow((n/(1022-max1)),2)),2
);
                                   }

       for(n1=1400; n1<=2500; n1=n1+deltasigma)  {
        psig=0.0;
         for(m=0; m<=(922-max1); m++)
          psig=psig+(intgrm4[m]*cos(2*M_PI*n1*m*factor));
          psig=psig*factor*0.4;
          if(psig<0)
          psig=(psig/3);
         lineto((10+(o*nn)),(460-psig*30));
          if(kbhit()!=0) break;
         nn++;
                                                    }
         return;
       }



  void main() {
   int n, shift;
   float max2, buff;
   float z;

    for(n=0; n<=1024; n++)
        intgrm1[n]=0;

   sprintf(menupoint1,"File");
   sprintf(menupoint2,"Sample");
   sprintf(menupoint3,"Run");
   sprintf(menupoint4,"Options");
   setgraphics();
   windowsetup();
    while('((menu_code=='1')&&(menucontrol()==0))) {
      switch(menucontrol())    {
```

```
            case 0: menu_code = menu0();
                    break;

            case 1: menu_code = menu1();
                     switch(menu_code)   {

                         case '1':   if(signaltest()==0)
                                       errormssge();
                                     else          {
                                      setfillstyle(1,0);
                                      setcolor(3);
                                      while(kbhit()==0) {
                                      outtextxy(280, 202, "Cont. S
ampl.");
                                      readout(&intgrm1);

                                      plot(intgrm1);
                                      for(n=0;n<=30000;n++);
                                      setcolor(3);
                                      bar3d(0, 200, 639, 479, 0, 0
);
                                                        }           }
                                  break;

                         case '2':     if(signaltest()==0)
                                         errormssge();
                                       else          {
                                          setcolor(3);
                                          setfillstyle(1, 0);
                                          outtextxy(290, 202,"Sing
le Sample");
                                          readout(&intgrm1);

                                          plot(intgrm1);
                                          getch();
                                          windowsetup();}
                                       break;

                         case '3':     if(signaltest()==0)
                                        errormssge();
                                       else          {
                                          setcolor(3);
                                          setfillstyle(1, 0);
                                           readout(&intgrm1);
                                          while(kbhit()==0)  {
                                           outtextxy(280, 202,"Aver
aging");
                                           readout(&intgrm2);
                                            for(n=0; n<=1022; n++)
 {
                                              intgrm1[n]=(intgrm1[n]
```

```
+intgrm2[n])/2;

 }
                                          plot(intgrm1);
                                         setcolor(3);
                                         bar3d(0, 200, 639, 479,
0, 0);
                                                          }    }
                        case '4':      if(signaltest()==0)
                                          errormssge();
                                       else                {
                                       setcolor (3);
                                       outtextxy(280, 202, "Backgr
ound");
                                       setfillstyle(1, 0);
                                        readout(&intgrm2);
                                        plot(intgrm2);
                                        getch();
                                       setcolor(3);
                                       bar3d(0, 200, 639, 479, 0,
0);
                                                          }

                                       break;

                                               }

                                 break;
                 case 2:    if (signaltest==0)
                                 errormssge();
                            else          {
                             for(n=0; n<=1023; n++)
                                acc[n]=0;
                              z=0;
                              while(z<floor(number_of_scans) && (z<20))
 {
                                readout(&intgrm1);
                                for(n=0; n<=1023; n++) {
                                   acc[n]=acc[n]+intgrm1[n];}
                                   z++;

                                                        }
                                for(n=0; n<=1023; n++)
                                  intgrm1[n]=(int)(acc[n]/(z));


                             if(backgnd_sub=='y') {
                               for(n=0; n<=1023; n++) {
                                intgrm3[n]=(intgrm1[n]-intgrm2[n])*0.5;}
                                                  }
                             else    {
                               for(n=0;n<=1023; n++)
```

```
                        intgrm3[n]=intgrm1[n];
                                          }
                      buff=intgrm3[0];
                       for(n=1; n<=1023; n++)  {
                         if(intgrm3[n]>buff) {
                           buff=intgrm3[n];
                           max1=n;           }
                                             }

                        if((apodization'=0)&&(interferogr=='t'))
{
                            apo(max1);       }

                         for(n=0; n<=1023; n++)
                          intgrm1[n]=floor(intgrm3[n]);
                         plotint(intgrm1);
                      /*          printf("max:%f",max1);    */
                         getch();

                        if(interferogr=='t'&& (max1<512))
                          twosidedft();
                        else {
                         if(max1<250)  {
                          phasecorr();
                          onesidedft();}
                             }
                                                    }
                     break;
          case 3:    setoptwin();
                     windowsetup();
                          break;


                           }
                        }

closegraph();

             }
```

References

(1) T. H. Barnes, "Photodiode Array Spectrometer with Improved Dynamic Range", Appl. Opt., **24,** 3702, (1985)

(2) T. H. Barnes, T.Eiju, K. Matsuda, "Heterodyned Photodiode Array Fourier Transform Spectrometer", Appl. Opt., **25**, 1864, (1986)

(3) M.L. Forman, W. H. Steel and G. A. Vanasse, J. Opt. Amer. **56**, 59, (1966)

(4) Sol M. Gruner, "CCD and vidicon x-ray detectors: Theory and practice", Rev.Sci.Instrum., **60**, 1545, (1989)

(5) M. Hashimoto, S. Kawata, "Multichannel Fourier Transform infrared Spectrometer", Appl. Opt., **31**, 6096, (1992)

(6) P. Jaquinot, "The Luminousity of Spectrometers with Prisms, Gratings or Fabry-Perot Etalons", J.Opt.Soc. Amer., **44**, 761, (1954)

(7) M.-L. Junttila, J. Kaupinen and E. Ikonen, "Performance Limits of Stationary Fourier Transform Spectrometers", Opt. Soc. Amer., **9**, 1457, (1991)

(8) M.-J. Junttila, "Stationary Fourier Transform Spectrometer", Appl. Opt., **31**, 4106, (1992)

(9) S.C. Leon, "Broad Source Fringe Formation with a Fresnel Biprism and a Mach-Zehnder Interferometer", Appl. Opt., **26**, 5259, (1987)

(10) L. Mertz, "Transformations in Optics", (Wiley, New York, 1965), 27-33

(11) T.Okamoto, S. Kawata and S. Minami, "Fourier Transform Spectrometer with a Self-Scanning Photodiode Array", Appl. Opt. **23**, 269 (1984)

(12) T. Okamoto, S. Kawata, S. Minami, "Optical Method for Resolution Enhancement in Photodiode Array Fourier Transform Spectroscopy", Appl. Opt., **24**, 4221, (1985)

(13) M J Padgett, A. R Harvey, "A Static Fourier Transform Spectrometer Based on Wollaston Prisms", Rev Sci Instrum , **66**, 2807, (1995)

(14) J Svatos, D Joyeux, D Phalippou, F Polack, "Soft-x-ray interferometer for measuring the refractive index of materials", Optics Letters, **18**, 1367, (1993)