University of Leipzig
Faculty of Physics and Earth Sciences

Diploma Thesis

Development of the Data Acquisition and Analysis Systems for a Portable Raman Lidar and a Doppler Wind Lidar

in partial fulfillment of the requirements for the degree of

Diplom-Physiker
(Graduate Physicist)

by
Peter Rhone
in June 2004

presented to
Prof. D. Michel and
Prof. J. Heintzenberg

It is better to be a hammer than an anvil.
- St.Dominic de Guzman

Contents

1 Introduction

The Earth's climate is principally affected by solar radiation. In the upper atmosphere, the yearly average of this radiation is 1370 W/m^2 through a surface whose normal points toward the Sun. A portion of this radiation is scattered and absorbed such that at midday only approximately 1000 W/m^2 reach the Earth's surface around latitude 50 °. This radiation heats the surface, which in turn heats the lowest regions of the atmosphere. A well-mixed aerosol and particle rich "**p**lanetary **b**oundary **l**ayer" (PBL) in the lower atmosphere is created by convection of the heated air from the Earth's surface. In the summer, the PBL extends from the ground up to 1.5–3.5 km. Depending on the reflectivity of the surface (albedo), radiation is reflected back into space. This radiation budget is influenced by various components of the atmosphere. One such component is CO_2, which decreases the amount of radiation reflected back into space and thus plays a role in global warming. In the year 2000, CO_2 increased the average radiation by +1.5 W/m^2 [1]. Atmospheric particles also affect the radiation budget, but are less well understood. Estimations of their effect upon the global energy budget range from –2 to +0.5W/m^2, depending upon the type, concentration, and size of the particles. Here both the direct and indirect aerosol effects play a role. The direct effect refers to the interaction of aerosol particles with solar radiation. The indirect effect, however, describes the role played by particles in the formation of clouds, and the subsequent effect the clouds have upon the radiation budget.

Natural sources of particles such as black carbon from forest fires, sea salt, and desert dust are always present. Additionally, particles are created from precursor gases such as SO_2. Volcanic eruptions can push such gases into the stratosphere, where sulphuric acid droplets form and can remain for years. Since the industrial revolution, man also emits significant quantities of particles and precursor gases into the atmosphere. The particles are carried to the upper PBL, where turbulent exchange processes may inject them into the free troposphere, from where they can be globally distributed. The free troposphere is responsible for global distribution since high-speed non-turbulent winds are normally present and the probability of precipitation is much lower than within the PBL.

In the study of the influence of particles on the global climate, knowledge about creation mechanisms, properties and concentration, and vertical transport is particularly important. The vertical transport determines the exchange between the PBL and the free troposphere. At the Leibniz Institute for Tropospheric Research (IFT), various measurement methods are implemented to obtain the aforementioned information. One branch of methods, optical remote sensing, encompasses measurements made with lidar (light detection and ranging) systems, solar photometers, and DOAS (differential optical absorption spectroscopy). The other branch includes in-situ methods such as measurements with nephelometers, CPC (condensation particle counter) and DMA (differential mobility analyzer).

This thesis is concerned with my contributions to the development of two lidar systems. The container-based multi-wavelength lidar [2] and a permanently installed Raman li-

dar [3] already present in the IFT allow measurements of particle distributions, water vapor, temperature profiles, and the depolarization properties of the atmosphere. The container based six-wavelength lidar with eleven detector channels has been successfully used in national and international measurement campaigns such as LACE 98 (Lindenberg Aerosol Characterization Experiment), ACE-2 (North Atlantic Regional Aerosol Characterization Experiment) and INDOEX (Indian Ocean Experiment) [4, 5, 6]. The permanently installed nine-channel Raman lidar delivers regular data for the European lidar network EARLINET (European Aerosol Research Lidar Network) [7]. Both of these systems are expensive to construct and operate. The container-based system is portable, but only with considerable effort. For these reasons, the first task in this work was the development of a small, portable, and inexpensive Raman lidar system. It should be possible with the system to determine the particle backscatter coefficient in the troposphere during the day, and at night both the particle backscatter coefficient and the particle extinction coefficient. This lidar system should also be automated to collect atmospheric data at preprogrammed intervals with minimal or no operator intervention. The development of the PBL should, weather permitting, be measured daily for statistical purposes. The measurements can be used, for example, to determine if a measurement with the larger systems is worthwhile.

The second task relates to the lidar measurements of the vertical wind speeds important to the exchange processes between the PBL and the free troposphere. The measurements are based on the relativistic Doppler effect, in which the frequency of light scattered from moving particles is shifted relative to the unscattered light. The Doppler frequency shift is proportional to the wind speed. Two detection methods can be used to determine the shift. The first is the edge technique [8], an optical method which makes use of steep edge filters with wavelength independent transmission tuned to the frequency under study. From the intensity of the transmitted light, the shift can be measured. The second method uses coherent detection in which the backscattered light from the atmosphere is mixed with a reference beam. The coherent mixing produces optical beating from whose frequency the wind speed can be calculated. My task was to develop the data acquisition and analysis software for a coherent 2-µm wind lidar at the IFT. The system is based on a system of the National Oceanic and Atmospheric Administration, Environmental Technology Laboratory in Boulder, Colorado [9]. This lidar, called HRDL for "high resolution Doppler lidar", also operates at 2 µm. Additionally, in this work I used knowledge gained from the experiences of Holger Linné with the heterodyne 1123-nm DIAL system (differential absorption lidar) of the Max Planck Institute for Meteorology in Hamburg. My thesis was developed in close co-operation with Ronny Engelmann, whose work can be found in his diploma thesis [10].

This thesis is organized into the following three main sections. The first section, chapter 2, introduces the theoretical basis necessary to understand my projects. The project section 3 details my work in the development of both the mini Raman lidar and the wind lidar system of the IFT. Experimental results are included. Following these sections, section 4 contains a summary of results and suggestions for future work.

2 Theory

This chapter introduces the theoretical basics of aerosol Raman lidar and Doppler wind lidar. I have limited the majority of content to the concepts and tools required for the understanding of later experiments and the interpretation of their results. The first section treats the physics behind the Raman lidar measurements performed with the portable Raman lidar of section 3.3, and includes all relevant formulae required to extract the backscatter and extinction coefficients from the acquired data. The second section introduces the principles required for the data acquisition and analysis system of the wind lidar of section 3.4. Emphasis is upon the concepts of digitization required for the retrieval of the measured signal, and the mathematics required to extract wind speed profiles from the digitized signal.

2.1 Raman Lidar

The portable Raman lidar system which was developed as part of this work detects the photons backscattered both elastically and at an inelastic wavelength. This dual channel arrangement allows the determination of particle extinction profiles without the assumptions that would be necessary in a backscatter lidar possessing only an elastic detection channel. The meanings of elastic and inelastic detection will be covered in the following sections in greater detail.

2.1.1 Basic Principles

In a basic lidar (**Li**ght **D**etection **a**nd **R**anging) system, as shown in Fig. 1, a pulsed laser emits radiation into the atmosphere. A portion of this laser radiation is backscattered by particles and molecules as the pulse propagates away from the system. The backscattered photons are collected by a telescope and passed to the detection optics, which may consist of a diaphragm in the focal point of the telescope, beam splitters, and selection filters for separating inelastically from elastically backscattered photons, optical filters to reduce the amount of light entering the photodetectors, and the photodetectors themselves. The time from pulse emission to detection is a measure of the distance from the system at which the backscattering occurred and provides lidar systems with the ability to make height resolved measurements. The electric signals generated by the photodetectors are passed to the data acquisition system for storage and analysis. Depending on system design, it is possible to measure atmospheric properties such as density, water vapor content, temperature, wind speed, cloud properties, minor constituents, and aerosols [11]. In this thesis, two types of lidar systems will be dealt with in greater detail: coherent Doppler lidar and aerosol Raman lidar.

The optical power returned to the lidar from height z contains both elastic and inelastic components. The elastically backscattered component of the received signal can be

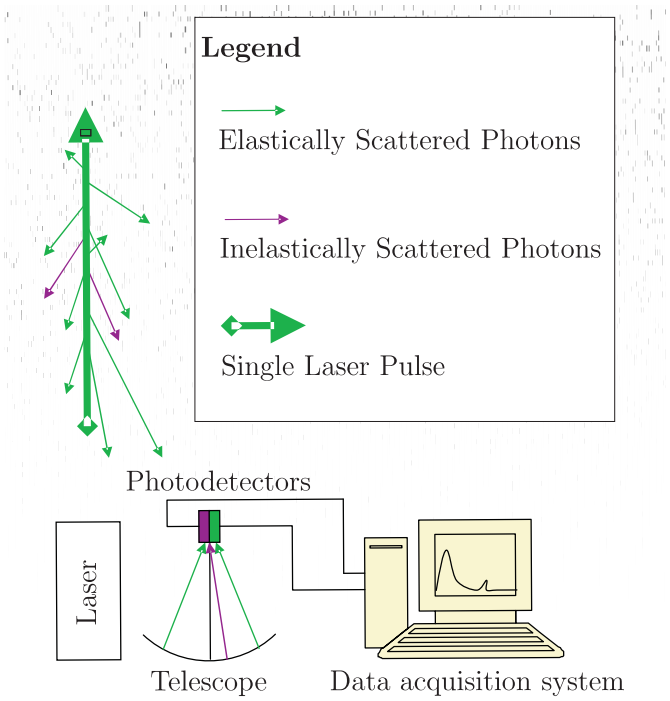

Figure 1: Simplified diagram of a two-channel aerosol Raman lidar.

calculated from the following equation:

$$P_{\lambda_0}(z) = P_0 \frac{c\tau}{2} \frac{A_T \eta_{\lambda_0} O(z)}{z^2} \beta(z) \exp\left[-2 \int_0^z \alpha_{\lambda_0}(z')\mathrm{d}z'\right] , \tag{2.1.1}$$

where

$P_{\lambda_0}(z)$ ... returned power at wavelength λ_0 from distance z [W],
$\beta(z)$ ... elastic 180° backscatter coefficient at z [$\mathrm{m^{-1}sr^{-1}}$],
$\alpha_{\lambda_0}(z)$ ... extinction coefficient at wavelength λ_0 at z [$\mathrm{m^{-1}}$],
P_0 ... average emitted power of the laser [W],
τ ... pulse length [s],
c ... speed of light [m/s],
A_T ... area of the telescope [$\mathrm{m^2}$],
η_{λ_0} ... transmission of the receiver optics at wavelength λ_0 [1],
$O(z)$ ... overlap function between emitter and receiver field of view [1].

The factor of two in the exponential term of Eq. (2.1.1) accounts for the transmission of the emitted light in the backscattering volume, as well as the transmission of the scattered light back through the same column of atmosphere on its return to the lidar.

The elastic backscatter $\beta(z)$ and extinction $\alpha_{\lambda_0}(z)$ coefficients consist of both molecular (M) and particulate (P) components such that $\beta(z) = \beta^M(z) + \beta^P(z)$ and $\alpha_{\lambda_0}(z) = \alpha^M_{\lambda_0}(z) + \alpha^P_{\lambda_0}(z)$. The molecular components, $\beta^M(z)$ and $\alpha^M_{\lambda_0}(z)$, can be calculated in principle from pressure determined with the barometric formula and temperature, but $\beta^P(z)$ and $\alpha^P_{\lambda_0}(z)$ cannot be calculated in a similar way. There are thus two unknowns in Eq. (2.1.1).

The function $O(z)$ describes the overlap between the laser beam, or the emitter's field of view, and the field of view of the receiver optics. In the Raman lidar used in this work, the sender and receiver optics are biaxial. The *minimum* distance from which a lidar backscatter signal can be measured occurs where the emitter's field of view enters the receiver's field of view. This distance $z = z_1$ is determined by the telescope's field of view, the divergence of the laser beam, the spatial separation and angle between the emitter and receiver optical axes, and overall system adjustment. From there, the overlap function increases to unity. Knowledge of the overlap function at every distance in the range of the measurement is important in signal averaging. If the overlap is zero, no information is contained in the acquired data. If the overlap is small, a greater number of shots must be averaged to obtain the same signal-to-noise ratio we would have measured had the overlap been unity.

The minimum distance z_1 for a lidar return signal in a biaxial lidar system can be calculated from [10]:

$$z_1 = \frac{d_0 - (d_T + d_L)/2}{\varepsilon + (\delta_T + \delta_L)/2}, \tag{2.1.2}$$

here

d_0 ... distance between the sender and receiver optical axes,
d_T ... diameter of the telescope mirror,
d_L ... diameter of the laser beam,
ε ... angle between the sender and receiver optical axes,
δ_T ... telescope's field of view,
δ_L ... divergence of the laser beam.

The overlap function $O(z)$ can be treated as unity at a certain height z_4 above the lidar system. This height is system dependent, but usually lies between 100 m and 1 km. This height can be calculated from [10]:

$$z_4 = \frac{d_0 + (d_T + d_L)/2}{\varepsilon + (\delta_T - \delta_L)/2}. \tag{2.1.3}$$

If the lidar is properly aligned, $O(z)$ should remain unity for all subsequent distances.

In Raman lidar systems, there is an inelastic or *Raman* signal in addition to the elastic component of the returned signal. In aerosol Raman lidar systems, inelastic scattering from N_2 and O_2 is used. The optical power of this signal is calculated from

$$P_{\lambda_R}(z) = P_0 \frac{c\tau}{2} \frac{A_T \eta_{\lambda_R} O(z)}{z^2} N_R(z) \frac{\mathrm{d}\sigma(\pi, \lambda_0)}{\mathrm{d}\Omega} \exp\left[- \int_0^z \left[\alpha_{\lambda_0}(z') + \alpha_{\lambda_R}(z') \right] \mathrm{d}z' \right]. \quad (2.1.4)$$

The additional symbols used here are

$P_{\lambda_R}(z)$ ... returned power at wavelength λ_R from distance z [W],
$\alpha_{\lambda_R}(z)$ ... extinction coefficient at the Raman shifted wavelength λ_R at z [m^{-1}],
η_{λ_R} ... transmission of the receiver optics at Raman wavelength λ_R [1],
$\frac{\mathrm{d}\sigma(\pi,\lambda_0)}{\mathrm{d}\Omega}$ ... Raman backscattering cross section [m^2/sr], and
$N_R(z)$ ... molecular number density of the Raman scattering gas [m^{-3}].

The Raman backscatter coefficient $N_R(z)\frac{\mathrm{d}\sigma(\pi,\lambda_0)}{\mathrm{d}\Omega}$ depends solely on the Raman backscattering cross section and the number density of the gas causing the scattering, i.e., no Raman scattering is observed from aerosols. This means that $\beta(z)^P$ is zero, and Eq. (2.1.4) contains only a single unknown, namely $\alpha^P(z)$. The exponential term contains two terms, the first represents the transmission of the laser beam on its outward journey, the second the transmission of the inelastically scattered light on its return to the receiver telescope.

Assuming a known gas concentration, the backscatter and extinction coefficients can be measured independently. The extinction coefficient is calculated from the inelastic backscatter equation (2.1.4) after putting it in the following form and assuming the overlap function to be unity:

$$\alpha^P_{\lambda_0}(z) = \frac{\frac{\mathrm{d}}{\mathrm{d}z} \ln \frac{N_R(z)}{z^2 P_{\lambda_R}(z)} - \alpha^M_{\lambda_0}(z) - \alpha^M_{\lambda_R}(z)}{1 + \left[\frac{\lambda_0}{\lambda_R}\right]^k} \,. \quad (2.1.5)$$

The k exponent in the denominator accounts for the assumption of a constant wavelength dependency in the extinction coefficient between the emitted and Raman shifted wavelengths. The value of k is often set to -1.

The backscatter coefficient, however, is calculated by determining the quotients of the elastic and inelastic signals at z and at a reference height z_0, that is $\frac{P_{\lambda_0}(z)}{P_{\lambda_0}(z_0)}$ and $\frac{P_{\lambda_R}(z)}{P_{\lambda_R}(z_0)}$,

and dividing these quotients. This results in a formula which can be solved for $\beta^P_{\lambda_0}(z)$:

$$\beta^P_{\lambda_0}(z) = [\beta^P_{\lambda_0}(z_0) + \beta^M_{\lambda_0}(z_0)] \frac{P_{\lambda_0}(z)}{P_{\lambda_R}(z)} \frac{P_{\lambda_R}(z_0) N_R(z)}{P_{\lambda_0}(z_0) N_R(z_0)} \frac{\exp\left[-\int_{z_0}^{z} \alpha_{\lambda_R}(z')\mathrm{d}z'\right]}{\exp\left[-\int_{z_0}^{z} \alpha_{\lambda_0}(z')\mathrm{d}z'\right]} - \beta^M_{\lambda_0}(z). \tag{2.1.6}$$

The reference height z_0 is chosen such that $\beta^M_{\lambda_0}(z) \gg \beta^P_{\lambda_0}(z)$, for example in the upper atmosphere where the aerosol concentration is negligible. Since the quotient of the overlap functions cancels in Eq. (2.1.6), it is possible to determine the backscatter coefficient for almost all distances greater than zero. The lowest possible distance limit for a useful extinction coefficient calculation is given by Eq. (2.1.3), since at distances less than z_4, the assumption that $O(z) = 1$ which was made in Eq. (2.1.5) is no longer correct. At distances less than z_1 where $O(z) = 0$ (no overlap), no conclusions can be drawn about the atmosphere. The same holds for the backscatter coefficient. Extinction determination below z_4 is only possible if the overlap effect can be corrected. In practice this is possible for heights for which $O(z) > 0.5$.

2.1.2 Elastic Backscattering

Elastic backscattering returns photons of wavelength approximately equal to that of the unscattered photons. Any differences significant to lidar measurements are due solely to Doppler induced frequency shifts. The mechanisms responsible for the backscattering are complex when the scattering centers have size dimensions comparable to the wavelength of the photons being scattered (Mie scattering) [11]. The responsible processes can be broken down into two groups: Rayleigh scattering from molecules and particles whose sizes are much smaller than the scattered photon's wavelength, and Mie scattering from larger particles. In Rayleigh scattering, the electric component of an electromagnetic wave induces a dipole moment in the molecules which it encounters. This induced dipole radiates its acquired energy elastically.

Mie scattering occurs from a wide range of particles. The difference in size between the smallest and largest particles spans many orders of magnitude. For isotropic dielectric spherical particles of much smaller size than the wavelength of the photons used in the investigation, the light is scattered in a way similar to Rayleigh scattering from molecules. Since any particle will be larger than a molecule, the backscattering cross section is correspondingly larger for small particles than for molecules.

If the size of the particle increases to the point where it is comparable to the wavelength of the photons it is scattering, Mie theory must be applied. The wavelength dependency of particle scattering ranges from about λ^{-4} for particles whose size dimensions are much smaller than the wavelength of the incident photon to λ^0 for large particles. It is derived by applying Maxwell's equations to a dielectric, isotropic, homogeneous

sphere and solving them for the far-field zone [12]. To a good approximation, Mie theory can also be applied to irregularly shaped atmospheric aerosol particles, as long as the particle dimensions do not greatly exceed the wavelength of the photons to be scattered. For large non-spherical particles, specific scattering theories such as the discrete dipole approximation or the T-matrix method have to be applied.

2.1.3 Inelastic Backscattering

Inelastic, or Raman, scattering returns light of a different frequency than the incident light. The prerequisite is a polarizable scatterer such as the N_2 molecules common in our atmosphere. The frequency shift is caused by a change in the vibrational and/or rotational state of the molecule from which the photon is scattered. If the molecule undergoes a vibrational state change, or a combined vibrational and rotational state change, the backscattered photon is shifted by a larger frequency than would have been the case had the molecule undergone a purely rotational state change. The larger the frequency shift, the easier it is to separate the shifted photons from the elastically backscattered photons. For this reason, the inelastic channel used in the Raman lidar of this work measures photons scattered by molecules in which a vibrational-rotational transition occurred. If the molecule ends in a higher energy state, the backscattered photon is redshifted and the the scattering process is called Stokes Raman scattering. If the opposite occurs, i.e., the scattered photon gains energy (blue shifted) and the molecule is left in a lower energy state, the process is called anti-Stokes scattering. At the temperatures found in the atmosphere, Stokes Raman scattering is more probable than anti-Stokes Raman scattering, and the measurable signal intensity is correspondingly greater by up to six orders of magnitude.

When comparing the scattering cross sections of inelastic backscattering to elastic backscattering, the inelastic cross section is a factor of about three orders of magnitude smaller. This results in a correspondingly weaker signal. The frequency shifts are characteristic of the molecule from which the scattering occurred, and can thus be measured to determine molecular types and concentrations present.

2.2 Doppler Wind Lidar

Doppler wind lidar makes use of the Doppler shift of incident light on molecules and particles carried by the wind to determine the component of the wind velocity along the line of sight of the lidar. In the following sections, I will describe the principles behind the Doppler shift, the experimental detection theory, and signal acquisition techniques.

2.2.1 Wind Induced Doppler Shift

The light emitted by the laser is elastically backscattered by particles moving with the wind. Backscattering can be broken down into a two parts. First, light from a stationary laser travels to a backscattering particle, and second, the scattered light returns from the particle to the receiver optics in the wind lidar system. To properly determine the frequency of the returning photon, it is therefore necessary to apply the relativistic Doppler equation twice. The result is double what would be expected from the simple case of a stationary emitter and mobile detector since no distinction is made in the relativistic equation between the velocity of the emitter and detector; only the relative velocity plays a role [13].

The general form of the relativistic Doppler equation is:

$$\frac{\nu_{obs}}{\nu_{rest}} = \frac{\sqrt{1-\frac{u^2}{{c_0}^2}}}{1+\frac{u}{c_0}\cos\Theta} = \frac{\sqrt{1-\frac{u^2}{{c_0}^2}}}{1+\frac{v_r}{c_0}}, \tag{2.2.1}$$

where

ν_{obs} ... measured photon frequency,
ν_{rest} ... photon frequency in the rest frame of the scatterer,
u ... velocity of the scatterer,
Θ ... angle between the observer and the scatterer's velocity vector,
v_r ... component of the scatterer's velocity onto the line of sight of the observer,
see Fig. 2.

For horizontal winds, $\Theta = 90°$, and the denominator of Eq. (2.2.1) is unity. The Doppler shift for a purely horizontal wind will therefore be *larger* than the shift for an upward vertical wind (u positive) of equal velocity magnitude. Compared to a downward wind (u negative), however, it will be *smaller*. At speeds much less than the speed of light, the numerator in the right hand side of Eq. (2.2.1) is approximately unity, and the denominator dominates. In such cases, horizontal wind speeds are no longer significant, and the only contributing factor is the radial speed of the wind.

Since we are more interested in the quantity $\Delta\nu = \nu_{obs} - \nu_{rest}$, which determines the frequency of the oscillations in heterodyne measurements, it is more convenient

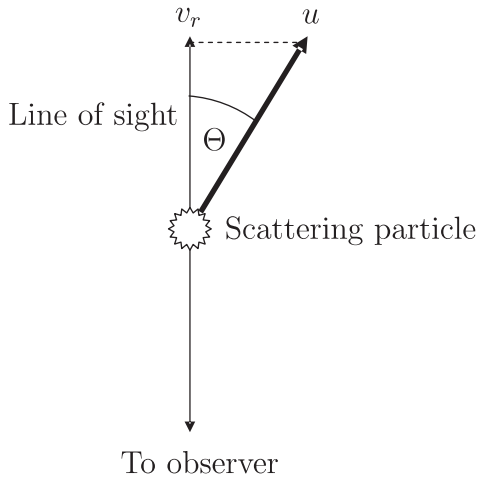

Figure 2: Terms used in the relativistic Doppler equation.

to simplify the above equation by first-order Taylor expansion of the numerator of Eq. (2.2.1) about $u^2 = 0$, $\sqrt{1 - \frac{u^2}{{c_0}^2}} \approx (1 - \frac{u^2}{2c_0^2})$. Introducing this Taylor expansion into Eq. (2.2.1) and rearranging the result for $\nu_{obs} - \nu_{rest}$ leads to:

$$\Delta\nu = \nu_{obs} - \nu_{rest} = \nu_{obs}\frac{u}{c_0}\cos\Theta - \nu_{rest}\frac{u^2}{2c_0^2}. \tag{2.2.2}$$

For everyday wind speeds, the cofactor $\frac{u^2}{2c_0^2}$ on the right of Eq. (2.2.2) is $\approx 10^{-16}$, which is much smaller than the cofactor $\frac{u}{c_0}$ of the first term on the right, which is $\approx 10^{-8}$. For this reason, the last term can be completely ignored for all values of $\cos\Theta$ not very close to zero. Setting $u\cos\Theta$ equal to the radial velocity v_r, dropping the last term of Eq. (2.2.2) and adding a factor of 2 to account for the reflection, we arrive at:

$$\Delta\nu = 2\nu_{obs}\frac{u}{c_0}\cos\Theta, \tag{2.2.3}$$

or more simply:

$$\Delta\nu = 2\frac{v_r}{\lambda_{obs}}. \tag{2.2.4}$$

The downdraft is blueshifted and the updraft is redshifted.

2.2.2 Coherent and Non-Coherent Detection

The measurement of the backscattered signal in a lidar system can be performed either coherently or non-coherently. Non-coherent measurement or *direct detection* involves the determination of a time resolved signal intensity. The light collected by the telescope is, after appropriate optical filtering, directed onto a photodetector. The intensity

of the measured signal (current or number of pulses counted) is related to the number of photons returned from a given height. The portable Raman lidar treated in this work performs non-coherent measurements. The *coherent* measurements used in wind lidar, however, require the mixing of the captured backscattered photons with photons from a *local oscillator* of fixed frequency. The measurement is no longer the simple quantification of signal intensity, but rather a frequency and/or phase analysis of the mixed signal. Coherent measurements can be either *heterodyne* or *homodyne*. The wind lidar makes use of heterodyne measurements, which can be used to determine the frequency difference between the backscattered Doppler shifted photons and the photons of the emitted pulse. This frequency difference is directly proportional to the speed of the wind, as shown in Eq. (2.2.4). Homodyne measurements can only be used to determine the phase difference between photons of the same frequency.

Homodyne and Heterodyne Detection In homodyne and heterodyne measurements, the received light is mixed with a portion of light from the local oscillator. For coherent mixing to occur, the two signals must have the same polarization direction. If the signals are perpendicularly polarized, a simple non-coherent addition of the signal intensities will be measured on the photodetector. The mathematical derivation for both types of detection is very similar, and application of the general formula to each specific case results in different relations. The general formula can be derived from the following simple observations (see Fig. 3):

The electric component of the electromagnetic field at the photodetector is composed of two unified components. One component comes from the received light:

$$E_r(t) = E_{r0}\cos(\omega_r t + \delta), \tag{2.2.5}$$

where

$E_r(t)$ ... electric field of the received photon field,
E_{r0} ... amplitude of the received field,
ω_r ... angular frequency of the received field,
δ ... phase offset of the received field relative to the local oscillator field,

and the other from the local oscillator:

$$E_L(t) = E_{L0}\cos(\omega_L t), \tag{2.2.6}$$

where L stands for local oscillator.

The photocurrent is proportional to the intensity $I(t)$, i.e., to the square of the sum of these fields:

$$I(t) \propto [E_r(t) + E_L(t)]^2 = [E_{r0}\cos(\omega_r t + \delta) + E_{L0}\cos(\omega_L t)]^2. \tag{2.2.7}$$

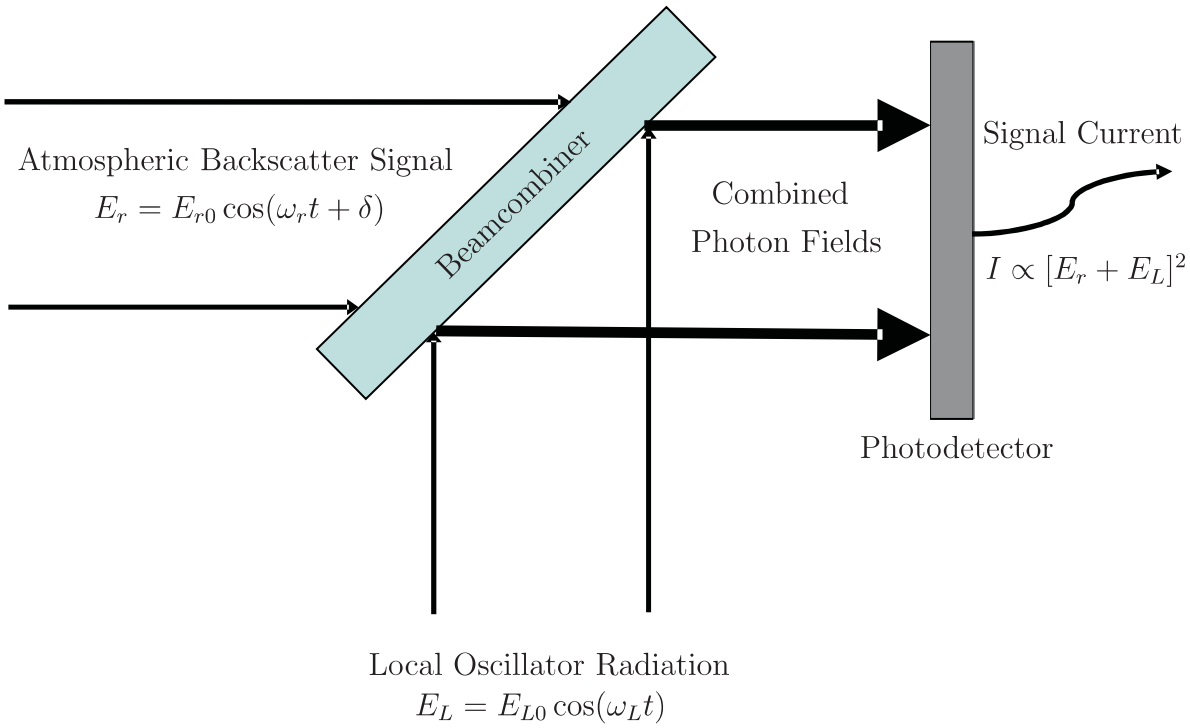

Figure 3: Principle of coherent detection.

Multiplying out the right hand side of (2.2.7) and using the trigonometric identities $\cos^2(a) = \frac{1+\cos(2a)}{2}$ and $\cos(a)\cos(b) = \frac{1}{2}[\cos(a-b) - \cos(a+b)]$ gives the general equation for the photocurrent of the detector:

$$I(t) \propto \frac{1}{2}E_{r0}^2[1 + \cos 2(\omega_r t + \delta)] + \frac{1}{2}E_{L0}^2(1 + \cos 2\omega_L t) + E_{r0}E_{L0}[\cos((\omega_r - \omega_L)t + \delta) + \cos((\omega_r + \omega_L)t + \delta)]. \quad (2.2.8)$$

In *homodyne measurements*, the received light is of the same frequency as the outgoing signal (and consequently also of the local oscillator), i.e., $\omega_r = \omega_L \equiv \omega_0$. Hence Eq. (2.2.8) simplifies to:

$$I_{homodyne}(t) \propto \frac{1}{2}E_{r0}^2[1 + \cos 2(\omega_0 t + \delta)] + \frac{1}{2}E_{L0}^2(1 + \cos 2\omega_0 t) + E_{r0}E_{L0}[\cos\delta + \cos(2\omega_0 t + \delta)]. \quad (2.2.9)$$

Since we are measuring light, the frequency ω_0 of the light lies many orders of magnitude above the maximum response frequency of available detectors $\omega_{response}$, and therefore only the average value of any time-dependent cosine functions oscillating at a frequency much larger than $\omega_{response}$ will be visible in the current. Since the average value of cosine over many periods is equal to zero, all time-dependent terms in Eq. (2.2.9) are

undetectable. The result for the detectable homodyne photocurrent $I_{homodyne}$ is thus:

$$I_{homodyne} \propto \frac{1}{2}(E_{r0}^2 + E_{L0}^2) + 2E_{r0}E_{L0}\cos\delta. \tag{2.2.10}$$

The photocurrent is therefore only proportional to the phase difference between the two fields interfering on the detector plus a DC offset voltage.

The situation is different in *heterodyne measurements*. The rapidly varying terms of Eq. (2.2.9) still average to zero, but the received light now has a different frequency than the outgoing signal. The term $2E_{r0}E_{L0}\cos((\omega_r - \omega_L)t + \delta)$ is no longer constant in time, but oscillates with the frequency $\Delta\omega = \omega_r - \omega_L$. The photocurrent $I_{heterodyne}(t)$ is thus time dependent:

$$I_{heterodyne}(t) \propto \frac{1}{2}(E_{r0}^2 + E_{L0}^2) + 2E_{r0}E_{L0}\cos(\Delta\omega t + \delta). \tag{2.2.11}$$

The heterodyne frequency $\Delta\omega$ can be measured if $\Delta\omega \leq \omega_{response}$. In the case of laser light Doppler shifted on particles moving with the wind, the heterodyne frequency is low enough to be measured by currently available technology. For example, Eq. (2.2.4) tells us that light of wavelength 2 µm reflected from a particle moving at 1 m/s is Doppler shifted by $\Delta\omega = 1$ MHz.

In the case of wind lidar, not only the magnitude of the wind velocity is desired, but also its direction. To accomplish this, the outgoing laser pulse is internally shifted by an additional amount $\Delta\omega_i$. This frequency shift is selected to be so large, that the sum $\Delta\omega_i + \Delta\omega_{Doppler}$ is always larger than zero. A stationary air volume results in a heterodyne signal of frequency $\Delta\omega_i$; downwinds increase this frequency and upwinds decrease it. Subtracting $\Delta\omega_i$ from the measured heterodyne frequency and setting the result to equal the $\Delta\nu$ of Eq. (2.2.4), negative speeds correspond to upwinds and positive speeds to downwinds.

2.2.3 Analyzing the Heterodyne Signal

The heterodyne measurement method of the wind lidar produces an analog signal from a photodetector that must be analyzed for its frequency components. The main frequency component is related to the wind speed. The analysis contains a number of stages. In chronological order, the stages are: filtering, digitization, windowing, zero-padding, and transformation into the frequency domain. I will treat each of these steps in this section in non-chronological but logical order, beginning with digitization.

Digitization of Analog Signals The digitization of an analog signal is performed in two stages, as shown in Fig. 4. The voltage of the signal is first sampled and held, after which the sampled voltage is assigned a number by the analog-to-digital converter

(ADC) corresponding to its value. This process is repeated at the digitization rate. The sample and hold stage assures that the input to the ADC doesn't change during

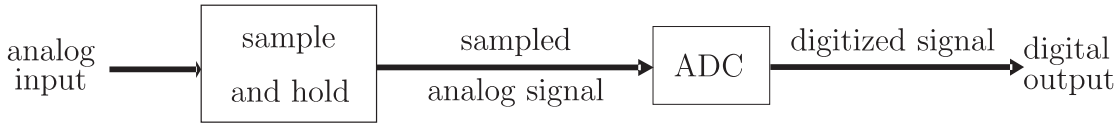

Figure 4: Digitization process.

conversion. In the ADC stage, the sampled voltage values are changed from continuous to discrete, or *quantized*. This introduces an error into the digitized signal, since a range of voltages will be digitized to the same value. The size of this voltage range depends on the number of bits used in the digitization, and is equal to the voltage range of the LSB (or least significant bit). Each sample in a digitized signal can thus have a maximum error of $\pm\frac{1}{2}$ LSB. A useful model of quantization error states that quantization results in the addition of random noise to the signal. This noise is spread uniformly from $-\frac{1}{2}$ LSB to $+\frac{1}{2}$ LSB, has a mean of zero and a standard deviation of $\frac{1}{\sqrt{12}}$ LSB [14]. Digitizing an analog signal with 8 bits thus automatically introduces a root mean square (rms) noise of $\frac{1}{2^8 \cdot \sqrt{12}} \approx \frac{1}{900}$. For a 1 V peak-to-peak signal, this would be ±1.13 mV. Treating the quantization error as random noise allows us to add it to noise already present in the analog signal. This is accomplished by adding the variances of the noise signals, i.e., $\sqrt{(signalnoise)^2 + (quantizationnoise)^2} = totalnoise$. Quantization error cannot be treated as random noise in slowly varying signals.

The sampling theorem (or Nyquist theorem) dictates that proper sampling of a signal is only possible if the frequency components of the signal lie below the Nyquist Frequency ν_{Nq}, which is half the sampling rate. If a signal is properly sampled, it can be exactly reconstructed from the sampled data. Any signal frequency components above ν_{Nq} are *aliased* to frequencies between 0 and ν_{Nq}. It is mathematically impossible to determine unambiguous information about the true frequency based solely on the aliased frequency. The presence of aliased frequencies in a measured signal can imperil sought after information. For example, the amplitudes of true signal components can be increased if they lie atop an aliased frequency, aliased frequencies can be mistaken for true signal components, and the SNR is decreased. For these reasons it is crucial to filter out any frequency components greater than one-half the sampling frequency *before* digitization, especially if these aliased frequencies are strong enough to contest with the true signal components under study. Aliasing can also shift the phase by inverting the signal (180°). This will not be further considered here, since in our use of the FFT, phase information is ignored.

Filtering Frequencies above half the sampling frequency should be prevented from entering the ADC, since they will be aliased and thus indiscernible from non-aliased

signal components. The simple scheme shown in Fig. 4 can be modified by placing a low-pass filter before the sample-and-hold stage of the digitization scheme, see Fig. 5.

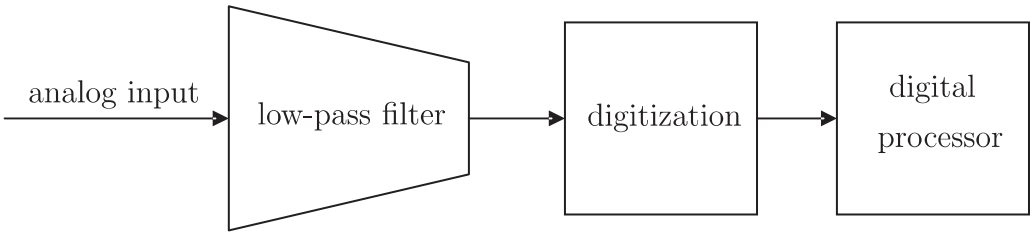

Figure 5: The analog low-pass filter is also called an antialias filter, since it prevents frequency components exceeding the Nyquist frequency from entering the ADC.

The most commonly used analog filters are Chebyshev, elliptic, Butterworth, and Bessel [14]. Each of these filters optimizes a different performance parameter, namely roll-off, passband flatness, and step response. Roll-off refers to the sharpness of the change from zero to infinite attenuation after the cutoff frequency. This parameter is important to antialias filters because all frequencies above the cutoff should be blocked whereas all others should be passed without attenuation. The Chebyshev filter is best in this respect. When selecting the cutoff frequency, the roll-off must be taken into consideration. If the cutoff frequency is set to match the Nyquist frequency, frequencies within a certain range beyond the cutoff will be aliased back into the signal due to the finite roll-off of the analog filter. The cutoff frequency, which is defined to be the point at which the amplitude is reduced to $1/\sqrt{2}$, should thus be set lower than the Nyquist frequency such that sufficient attenuation is reached for all signal components greater than the Nyquist frequency. Frequencies beyond the Nyquist frequency still within the tail of the filter roll-off will be aliased back into the filter passband. Due to the combined effects of roll-off and aliasing, the general rule of thumb is that the frequency band between 0.4 and 0.5 of the Nyquist frequency should be ignored [14].

The flatness of the attenuation curve within the passband is called passband ripple. Butterworth filters provides the steepest roll-off possible while keeping the passband flat [14]. The steeper roll-off achieved by the Chebyshev filters is attained at the expense of passband flatness. A common value for the passband ripple of Chebyshev filters is 0.5 dB.

The final parameter is the step response and refers to the filter's response when the input rapidly changes. This parameter is of little interest in this work and will thus be ignored.

In the frequency analysis of wind lidar signals, the information is encoded in the frequency domain, and the phase information is irrelevant. For this reason, a Chebyshev, elliptic, or Butterworth filter would be most appropriate. Of these, the Chebyshev or elliptic are probably best since the roll-off is steepest. The greater ripple in the passband as compared to the Butterworth filter should have no negative effect on the

interesting aspects of the acquired data, since exact amplitude analysis of the frequency spectrum is not performed.

Determining the Frequency Components of a Sampled Signal After the signal has been sampled, the frequency components must be found. This can be accomplished with a discrete Fourier transform (DFT). This transform converts the time-domain signal into the frequency domain. If the input to the DFT is an array of N time-domain points, the output is an array of N complex numbers representing the frequency components. Only the first $N/2+1$ output array elements contain relevant information. The remaining half of the array contains duplicate values mirrored onto the negative frequency axis and can thus be ignored. This transform entails finding the projection of the observed signal onto an orthogonal basis set [15]. This set consists of sines and cosines with periods equal to an integer submultiple of the observation interval NT, and is defined for continuous signals as:

$$\left.\begin{array}{l}\cos\left[\frac{2\pi}{NT}kt\right]\\ \sin\left[\frac{2\pi}{NT}kt\right]\end{array}\right\}\begin{array}{l}k=0,1,...,N-1,N,N+1,...\\ 0\leq t\leq NT.\end{array} \tag{2.2.12}$$

For sampled signals, the basis set is identical but the frequency domain sequences are obtained from $\frac{N}{2}+1$ samples of the continuous spanning set:

$$\left.\begin{array}{l}\cos\left[\frac{2\pi}{NT}knT\right]=\cos\left[\frac{2\pi}{N}kn\right]\\ \sin\left[\frac{2\pi}{NT}knT\right]=\cos\left[\frac{2\pi}{N}kn\right]\end{array}\right\}\begin{array}{l}k=0,1,...,\frac{N}{2}\\ n=0,1,...,N-1.\end{array} \tag{2.2.13}$$

The DFT can be calculated in three different ways. The first is by solving a set of simultaneous equations. This method is inefficient and will not be further discussed in this work. The second method makes use of correlation, and the third method is called the Fast Fourier Transform (FFT). If the input to the DFT contains less than thirty-two points, correlation is sometimes used (although an FFT is approximately ten times faster at calculating the same 32-point DFT), otherwise the FFT is the method of choice.

The calculation of the DFT by the correlation method is accomplished by solving the following equations:

$$\begin{aligned} Re\{X[k]\} &= +\sum_{i=0}^{N-1} x[i]\cos\left(\frac{2\pi ki}{N}\right),\\ Im\{X[k]\} &= -\sum_{i=0}^{N-1} x[i]\sin\left(\frac{2\pi ki}{N}\right), \end{aligned} \tag{2.2.14}$$

where

$Re\{X[k]\}$ ... real part of the k^{th} frequency domain sample,
$Im\{X[k]\}$ ... imaginary part of the k^{th} frequency domain sample,
$x[i]$ ... the i^{th} sample of the time domain signal being analyzed,
k ... index of the frequency domain sample and runs from 0 to $\frac{N}{2}$.

Thus each sample in the frequency domain is found by multiplying the complete time domain signal by the sought after sine or cosine wave, and summing the resulting points. The negative sign preceding the summation in the imaginary component is included to make the real DFT consistent with the complex DFT.

There are a number of algorithms available to calculate the FFT. Although the first algorithm was introduced by Karl Friedrich Gauss (1777-1855), it was largely forgotten. The application of digital computers to digital signal processing led to increased demand for an efficient algorithm to calculate the DFT. For this reason, Cooley and Tukey are generally given credit for the discovery of the FFT since they introduced their algorithm at an opportune time. Their algorithm is described in their paper [16]. The basic idea can be summarized as follows:

1. Decompose the time domain signal of length N into N time domain signals by reordering the input elements and treating each element as a distinct time domain signal. The reordering is accomplished by bit reversing the element indicies[1] (LSB→MSB, e.g., sample # 1 becomes sample # 8 since 0001→1000, and sample # 3 becomes sample # 12 since 0011→1100).

2. Calculate the corresponding N frequency spectra of each of the N time domain signals. Since the frequency spectrum of a 1-point time domain signal is equal to itself, no work is required here.

3. Combine the resulting spectra to form a single spectrum. The combination is done in the reverse order of the time domain decomposition. This is accomplished via a butterfly calculation, which is the basic computation element of the FFT. It takes two complex numbers, z_1 and z_2, and forms the quantities $z_1 + z_2 e^{-i2\pi k/N}$ and $z_1 - z_2 e^{-i2\pi k/N}$.

Several variations on the Cooley-Tukey algorithm can further improve the performance of an FFT computation by 20–40% [14].

When a computer calculates a DFT of length N by correlation, two loops running N times must be executed. Thus, the number of operations required is $\propto N^2$. If the DFT is calculated using the FFT method, there are $\log_2 N$ stages, and in each

[1] Signal length must be a power of two for bit reversal to function

stage there are $N/2$ butterfly computations. The decomposition of the time-domain signal requires virtually no time. The number of computations is thus $\propto N \log_2 N$, and the FFT is always more efficient than an equivalent DFT calculated by correlation if the constant of proportionalities required to determine the execution times are not significantly different. A further advantage of the FFT over correlation calculations results from the reduced number of calculations required. Fewer calculations means fewer truncations and thus greater precision. The FFT algorithm is thus more precise and faster than the same DFT calculated by correlation [14].

Windowing As can be seen in Fig. 6, simply calculating the DFT on an unwindowed signal can lead to the high floor present in the lowest frame. To achieve the improved results shown in the third row of Fig. 6, two modifications can be applied to the sampled signal. The first is windowing, or apodization, and is treated here. The second is zero-padding, and is discussed below. The results of performing these steps before calculating the DFT on the same sampled signal can also be seen in the third row of Fig. 6.

A periodic infinite signal is necessarily truncated when acquired since the acquisition time is finite. This truncation can be mathematically described as the multiplication of the signal with an apodization or windowing function. For example, a rectangular apodization function of length 128 allows 128 points to retain their original values while setting all other points of the signal to zero. A Blackman-Harris window of the same length additionally shapes the 128 non-zero points, as shown in the second panel of Fig. 6. Since multiplication in the time domain corresponds to convolution in the frequency domain, and a perfect sinusoid in the time domain transforms to a delta function in the frequency domain [14], the spectrum of a windowed sinusoid is the spectrum of the window centered around the frequency of the sinusoid.

Frequencies from the continuum of possibilities between 0 and half the sampling rate which coincide with one of the $\frac{N}{2}+1$ basis vectors in the frequency domain [see equation (2.2.13)] will have a non-zero projection on the corresponding basis vector only. All other frequencies, however, exhibit *spectral leakage*, i.e., non-zero projections onto *all* basis vectors [15]. Spectral leakage results from the aperiodicity in the observation window of signal components with frequencies differing from those of the basis set. Since the DFT considers the sampled input signal to be periodic in the observation window [15], the periodic extension of the signal introduces discontinuities at the boundaries of the observation. These discontinuities have non-zero projections onto the entire basis set. Window functions help minimize this spectral leakage by reducing the order of the observation window boundary discontinuities. This is achieved by smoothing the sampled signal so that it not only approaches zero at its edges, but also has derivatives at the beginning and end that match (zero is a good choice here).

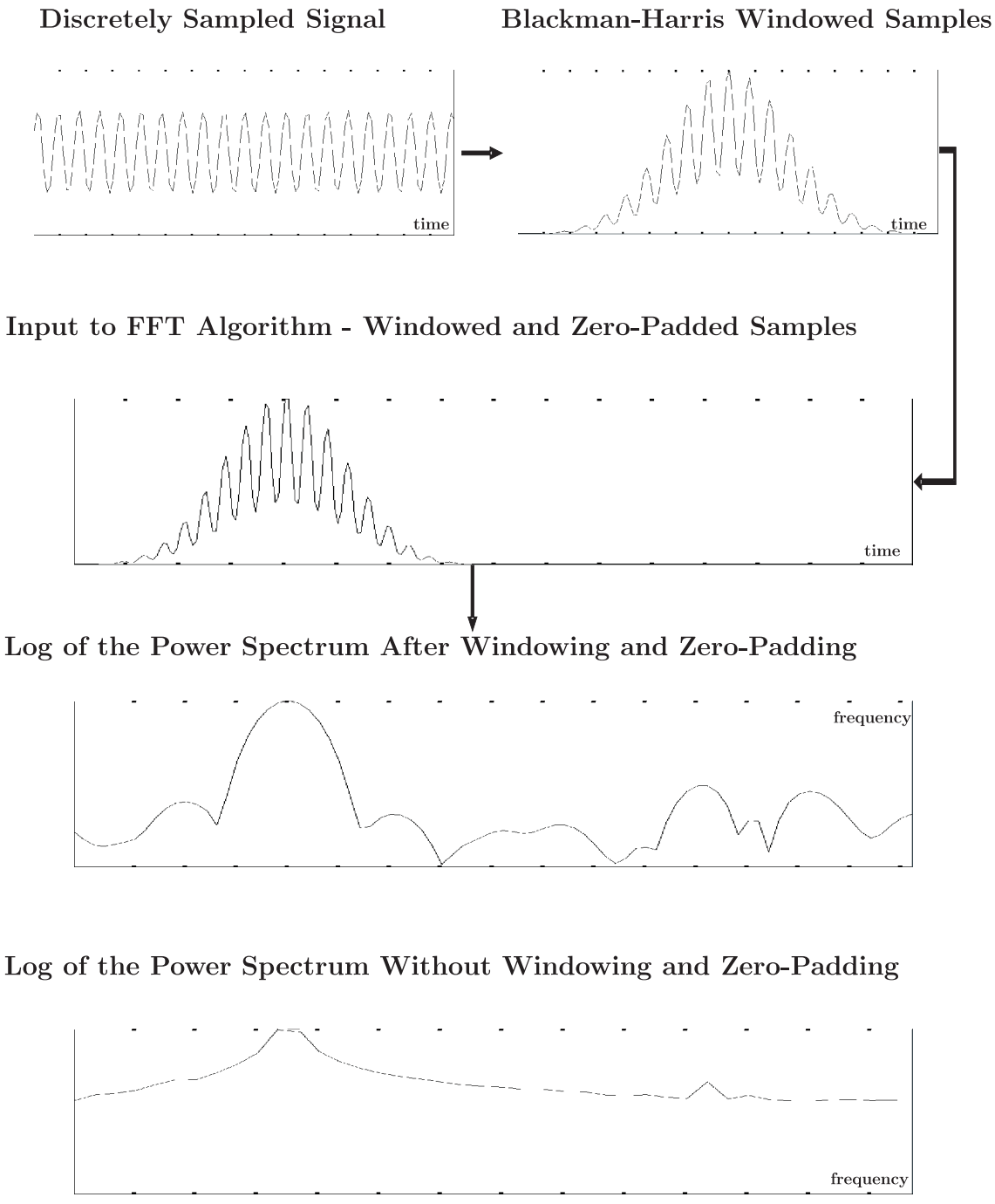

Figure 6: Stages in frequency determination.

One such window is the Minimum 4-Term Blackman Harris window, defined for the DFT as:

$$w(n) = a_0 - a_1 \cos\left(\frac{2\pi}{N}n\right) + a_2 \cos\left(\frac{2\pi}{N}2n\right) - a_3 \cos\left(\frac{2\pi}{N}3n\right), \tag{2.2.15}$$

where

$n = 0, 1, 2, ..., N-1$,
$a_0 = 0.40217$,
$a_1 = 0.49703$,
$a_2 = 0.09392$,
$a_3 = 0.00183$.

The upper graph in Fig. 7 shows the shape of a 64-point Blackman-Harris window in the time domain. The DFT of this window function multiplied by the 64-point sample of a sinusoid sampled at four times its frequency is shown in the lower graph.

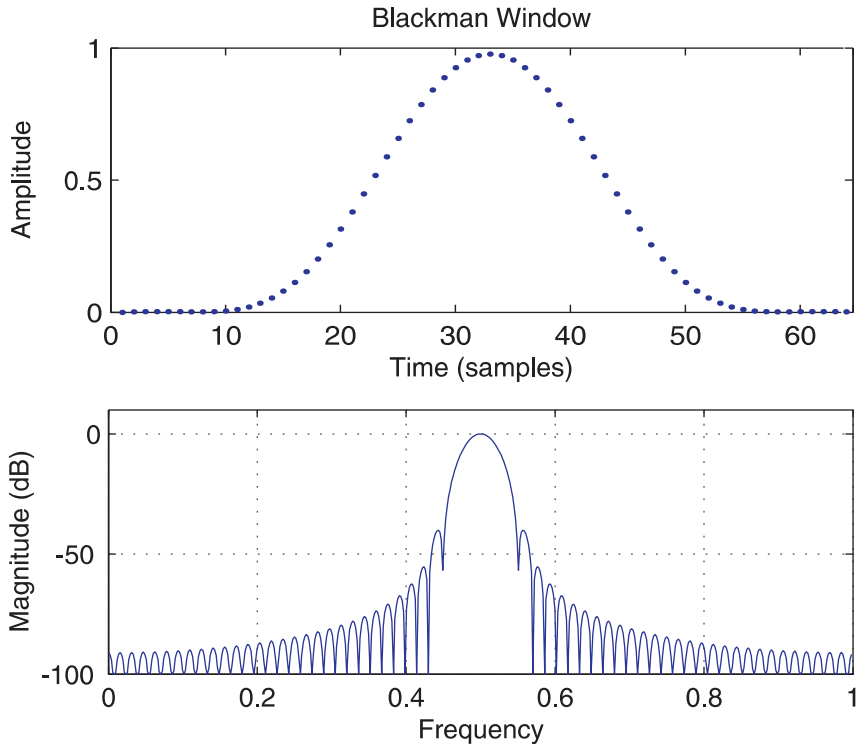

Figure 7: The Blackman-Harris window and the windowed transform of a sinusoid with a frequency equal to 0.25 the sampling rate. The magnitude of the main lobe has been normalized to 0 dB.

Windowing necessarily affects the frequency domain by broadening the peak and adding side lobes. The amplitude and extent of the side lobes as well as the width of the peak

can be optimized through selection of an appropriate apodization function. Generally, there is a trade off between peak width (or resolution) and side lobe amplitude (or spectral leakage), i.e., a window function which has a narrow peak has large side lobes.

When selecting an appropriate window, there are a number of parameters that must be considered. Of particular interest in this work are the bandwidth increase resultant from the window and spectral leakage.

The minimum spectral resolution Δf of an N point DFT sampled at f_s is found from [15]

$$\Delta f = \beta \left(\frac{f_s}{N} \right). \tag{2.2.16}$$

The factor β reflects the bandwidth increase due to the window function. For common windows, it ranges from 1 to 2.08.

A good window should have a low peak sidelobe level, and the sidelobes should fall off rapidly. This helps reduce the bias in the amplitude and position of a spectral line caused by the contribution of a given spectral component at frequency ω_0 to another spectral component at ω_1. The contribution is determined by measuring the gain at ω_1 of the window centered at ω_0 [15].

Zero Padding After the signal has been digitized and a window function has been applied, zeros can be padded as shown in the second row of Fig. 6 before performing the DFT. The advantages gained by this step are discussed here.

A sampled input signal to a DFT can contain *any* frequencies between zero and half the sampling frequency. It is therefore necessary that the transform be continuous over this range. Thus even though the time domain contains only discrete sampled values of the signal, the frequency domain transform of the signal samples is continuous. An N point DFT of a signal produces $\frac{N}{2} + 1$ *samples* of this continuous frequency domain curve. Increasing the length N of the DFT input increases the number of samples of the transform. This results in improved resolution in the frequency domain. One way to increase the length of the DFT is to pad the sampled input signal with zeros by adding zeros to the end of the array. The resolution in the frequency domain can theoretically be infinite, if the input is padded with an infinite number of zeros. Padding an infinite number of zeros forces the time domain signal to be aperiodic, and simultaneously reduces the spacing between the samples of the transform in the frequency domain to zero, thus producing a continuous curve. This is the same as calculating a discrete time Fourier transform (DTFT), which transforms a discrete aperiodic time domain signal into a continuous curve in the frequency domain [14]. Aside from increased resolution in the frequency domain, a further reason to pad the input array with zeros is to make the array length a power of two. The Cooley-Tukey FFT algorithm requires an input array whose length is a power of two.

Zero padding is a form of spectral interpolation which adds no false information to the signal. For this reason it is preferred over a spline fit to frequency domain samples. The decision as to the most appropriate interpolation should be based on execution speed. To reduce the sample spacing in the frequency domain to the desired level, the length of the DFT must be increased. It may be faster in certain circumstances to keep the DFT as small as possible and interpolate in the frequency domain using a different method such as those discussed in section 3.4.4.

All these mathematical methods for the extraction of the frequency components from a sampled signal are implemented in the wind lidar. Test results in section 3.4.5 quantify the improvements thus attained.

2.2.4 Signal to Noise Ratio

The **S**ignal to **N**oise **R**atio or SNR is a measurement of detection sensitivity [17]. It is defined for functions with a single peak as the ratio between the maximum signal voltage S_{max} and the root-mean-square of the noise voltage $\sqrt{\overline{n^2(t)}}$:

$$SNR = \frac{S_{max}}{\sqrt{\overline{n^2(t)}}}. \tag{2.2.17}$$

This equation is only valid if the noise is frequency independent (white) in the spectral range used to detect the signal.

Since the calculation of the standard deviation is a slow operation, it is possible to find an average value $\overline{R}$ of $\frac{n_{pp}}{\sqrt{\overline{n^2(t)}}}$, where n_{pp} is the peak-to-peak value of the voltage. One can then use this average value to approximate the SNR:

$$SNR \approx \frac{\overline{R} S_{max}}{n_{pp}}. \tag{2.2.18}$$

In the determination of $\overline{R}$, it is necessary to calculate the root-mean-square of the noise. The integral

$$\overline{n^2(t)} = \frac{1}{T}\int_0^T [n(t) - \overline{n(t)}]^2 dt \tag{2.2.19}$$

must be solved numerically. Here, $\overline{n(t)}$ refers to the DC component of the signal, and is calculated from the integral

$$\overline{n(t)} = \frac{1}{T}\int_0^T n(t) dt. \tag{2.2.20}$$

3 Projects

This chapter describes my contributions to the portable Raman lidar "Polly" and the wind lidar system under development as of June 2004 at the Leibniz Institute of Tropospheric Research in Leipzig. The tasks assigned to me in both cases were primarily related to the data acquisition systems and analysis software. As such, this chapter begins with a brief introduction to the software design concepts I made use of in this work, followed in section 3.2 by general computer selection criteria. My contributions to the portable Raman lidar system are then covered in section 3.3, and to the wind lidar system in section 3.4.

3.1 General Software Design Concepts

There are a number of important points to be considered during the development of the control and analysis software for complex systems. Since system hardware may be modified or replaced by new hardware with different control requirements, it is essential that the program code be easily rewritable to accommodate new hardware. Years after initial program development it may be necessary or advantageous to implement more efficient algorithms or program additional software functionality. This could be a difficult task if the program code is difficult to understand. The code itself should therefore be structured to simplify comprehension, but at the same time be reliable and perform its tasks efficiently. With these software design goals in mind, object oriented programming and the state machine structure present themselves as appropriate for the portable Raman lidar and Doppler wind lidar systems considered in this work.

3.1.1 Object Oriented Programming

Object oriented design is a convenient approach to modeling systems that interact with the world in real time [18]. In this style of programming, the trick lies in the correct identification of relevant objects, the organization of these objects into a hierarchal *class* structure, the definition of the attributes which determine the features of the classes in the problem context, and the addition of methods to the classes which perform specific tasks. Object oriented programming languages simplify the object identification process by providing three key features: encapsulation, inheritance, and polymorphism [19].

Encapsulation is the process of breaking a complex program into self-contained modules or classes. By encapsulating program classes, only the objects and methods of the class which are explicitly defined as public are visible outside the class. This increases the reuseability of encapsulated modules by hiding the complexity of their implementation. The process of hiding details and data structure from other objects

is called *data hiding*. It is good policy to hide all but the most essential details of a class. Encapsulation thus reduces the complexity of large programs.

Inheritance incorporates the concept of *derived classes*. Its two primary advantages are the ability to design hierarchies which define relations between types, and the capability of inheriting features from classes higher in the hierarchy. Derived classes simply inherit the components of a parent or base class and extend its functionality. This is consistent with the DRY[1] principle [20] of programming.

Polymorphism allows a class to have the same behavior but implement it in different ways. This concept find application in the design of a class which communicates with a data acquisition card. If the card is replaced with a different one in the future, it would be highly advantageous if the only changes necessary to the program could be found in the class controlling the card, the public methods being accessed from external program modules in an identical fashion.

The class structures used in object oriented design are generally represented in a *class diagram* using the **U**nified **M**odelling **L**anguage UML. Class diagrams describe the types of object in the system and their interrelations. With UML, a class is drawn as a box divided vertically into three sections. The top section contains the name of the class, the middle section the attributes, and the bottom section the methods or operations. Each attribute or method is preceded by a symbol which represents its visibility, i.e., –, +, #, ∼ for private, public, protected, and package, respectively. This makes the class encapsulation apparent. Inheritance between classes is represented by arrows. Interclass dependencies are indicated by a dotted arrow connecting the dependent classes together. Dependencies should be few, since changing a class upon which others depend requires changes to all the dependent classes. Minimizing dependencies maximizes the maintainability of the program code.

The primary advantages to object oriented programming are comprehensibility, reuseability, and extensibility. The class approach imposes a structure as program size increases. Ideally, object oriented systems can be assembled from preexisting classes, and new requirements can be met by adding functionality to derived classes only. In this way, design time is reduced and perhaps most importantly, reliability is increased through the use of thoroughly tested base classes.

3.1.2 State Machines

Designing a program to operate as a state machine can offer a number of advantages over a traditional top-down structure. The resulting program can be faster, more modular, less coupled, and easier to maintain [21]. It is easy to avoid duplication, and intent and implementation are decoupled. This decoupling simplifies program maintenance. The advantages of state machines are most apparent in systems which

[1] **D**on't **R**epeat **Y**ourself

handle a large number of input events, and have many different states which depend on the input events.

A state machine is a structure consisting of unique states. The states include an initial and/or default and one or more final states. When a final state is reached, the program either exits or returns to the default state. States are connected by transitions named generally after the events with which they correspond. Events thus *cause* transitions between states, where the new state is determined by both the current state and the event. State machines can be represented by state diagrams in which states are drawn as circles and transitions as labeled arrows.

A program design modeled as a state machine does not exclude an object oriented approach. The combination through the use of classes allows a neat interface to the system being modeled [22].

3.1.3 Code Optimizations

If the program needs to perform time critical tasks, then the selection of a programming language and program flow can be of critical importance. C++ is known to be an efficient language [18], but optimizing compilers are available for most CPUs which use processor-specific extensions to make the executable code as fast as possible. It is, however, still necessary to be careful during the coding of time critical routines, especially if these routines are called frequently. In a complicated program which controls both data acquisition and analysis, it is probable that there are subroutines which need a long time to execute due to interaction with slow hardware such as hard discs and PCI-bus data acquisition cards. If a large number of calculations need to be performed on the acquired data once it has been moved into the system RAM, it would inefficient to wait for the slow subroutines to complete before performing data manipulations and/or calculations on already acquired data. To program efficiently, it is therefore necessary to make a careful time analysis of the program functions. A proper analysis includes not only the time required per function, but the number of calls to the function in question. Once this analysis has been completed (often with the help of a profiler program which runs in parallel to the program to be analyzed and conducts the time analysis in real time), the program execution should be *threaded* to allow slow routines to run in parallel with the main program. Through the use of threads slow operations which require little CPU time such as saving data to the hard drive and retrieving acquired data from the acquisition card can be performed in parallel with other routines such as polling the user interface for button clicks, updating the visual output, and evaluating the acquired data.

When designing a time-critical multi-threaded program, it is essential to assign an appropriate priority to each thread before launching it. The priority of a thread determines the CPU time slice assigned to the thread. The higher the priority, the more time the CPU spends executing the thread before moving on. Threads which update the visual output, for example, should usually be classified as low priority, whereas data

retrieval from the acquisition card, data analysis, and data storage may in many cases be given a high priority. A further consideration that should be made when assigning priorities is the amount of CPU time *needed* by the thread in question to operate at maximum efficiency. A thread which simply waits for an event need not be assigned a high priority since the CPU idles during the time assigned to the inactive thread.

3.1.4 Graphical User Interface Design

A well designed *Graphical User Interface*, or GUI, is a useful tool in scientific programs. It requires careful design considerations since it is often the only convenient interface between the program and the user. By making ample use of messages to alert users when necessary, and checkboxes and buttons which set the program and measurement states, the GUI reduces the need of the user to resort to program source code comments. It also simplifies the setting of measurement parameters, outputs, and data analysis by eliminating the need for an initialization file. The interface should be as easy to use as possible by abstracting the user from all but the most necessary details of the underlying implementation. It should only allow meaningful parameter values, and should provide error and warning messages which contain useful information.

The GUI elements of the program interface can be classified according to the source of the action, see Tab. 1, or according to the actions taken, see Tab. 2 [23]. Each of

Table 1: GUI Element Classification By Source

Program Initiated GUIs	User Initiated GUIs
Help, Warning, and Error Dialog Boxes	Display and/or Plot
Message Boxes	Start/Pause
Inputs and Queries	Terminate Running Loop/Exit
	Reassign-While-Running

Table 2: GUI Element Classification By Action

Informational	Interactive Control	Assignment
Help, Warning, and Error Dialog Boxes	Start/Pause	Specification of Cases
Message Boxes	Terminate Running Loop/Exit	Modify Default Values at Initialization
Visualization	Reassign Variables While Running	Reassign Variables While Running

the elements have properties which must be considered before implementation. Dialog

boxes, for example, generally suspend program operation until they are acknowledged. If need be, this default behavior can be overcome. Error messages should suspend program execution, whereas warning messages may in some cases allow the program to continue functioning.

3.2 General Remarks on Computer Selection

Since the computer is at the heart of any **d**ata **acq**uisition system (DAQ), its characteristics are of critical importance to the project as a whole. It must, at the very least, be able to retrieve the data from the DAQ card before the data is overwritten. In running experiments, the acquisition system must also be able to save the information to a long term storage medium at the same average rate as the acquisition. If these conditions are not met, the data retrieved from the card will be corrupt, and/or data will be lost when the computer's RAM fills up. Generally, however, the computer additionally performs data analysis on the raw acquired data and saves only the results of the analysis to a permanent medium. Although such analysis increases the load on the computers **c**entral **p**rocessing **u**nit (CPU), it may reduce the amount of data to be saved, since analyzed data can be many times smaller than the raw data from which it was derived.

If we assume that the card located on a *shared bus* is capable of delivering data as fast as the computer can request it, then the maximum data transfer rate R_{max} in MB/s from the card memory to the CPU is determined by the bit width of the shared bus, w_b and the bus frequency f_b in MHz by

$$R_{max} = \frac{1}{8} f_b w_b. \tag{3.2.1}$$

The minimum transfer time T_{min} in seconds can therefore be calculated as a function of the number of bytes B to be transfered:

$$T_{min}(B) = \frac{B}{10^6 R_{max}}. \tag{3.2.2}$$

In reality, however, this value is rarely sustained or even met, since other devices on the shared bus demand a portion of bus frequency cycles for themselves. One such bus is the **I**ndustry **S**tandard **A**rchitecture (ISA) bus. This bus is 16 bits wide since IBM introduced the PC-AT in 1984. It runs at a non-guaranteed frequency between 8 MHz and 10 MHz. If the shared bus uses **p**eripheral **c**omponent **i**nterface (PCI) architecture, however, it is possible to attain the theoretical maximum for two primary reasons. The first is **d**irect **m**emory **a**ccess (DMA), which refers to the ability of bus devices to access system memory without interference from the CPU. The second reason is the ability of PCI computers to support more than one bus, which makes it possible to place the data acquisition card on a dedicated bus to prevent time sharing with other devices

[24]. If a large amount of data is being acquired in real time, a PCI dual bus system is a very practical choice. The PCI bus used in the wind lidar system is 64 bits wide and operates at 133 MHz.

For the saving of data in time critical systems, the choice of a storage medium depends primarily on the volume of data to be stored. Generally, the choice is between saving data to an **i**ntegrated **d**rive **e**lectronics (IDE) device connected to an **AT a**ttachment interface (ATA) (where "AT" is short for the IBM model AT computer), or saving to a **s**mall **c**omputer **s**ystem **i**nterface (SCSI) device or device array. ATA is a standard method for connecting storage devices to a computer initially developed for the IBM AT computer. Since its initial development, the standard has undergone several revisions. The current revision is called ATA-5 or Ultra ATA/66. SCSI is a component interface high speed bus allowing multiple devices to be connected in an array. If the SCSI device array consists of hard drives, it can be setup to operate as a **r**edundant **a**rray of **i**ndependent **d**iscs (RAID). Such a configuration can increase read/write performance to levels unattainable by an IDE device. Both IDE/ATA and SCSI are used to connect storage devices such as hard drives, CD/DVD-RW drives, and tape drives.

Individual IDE devices, using the latest ATA-5 revision, support DMA transfers and a theoretical maximum transfer rate of 66.67 MB/s [25]. Assuming the computer can deliver data to the ATA interface at this rate, the true theoretical maximum transfer rate is limited by device characteristics such as the angular frequency of the hard disc drive. SCSI has several benefits over IDE, particularly in speed. It supports transfers of up to 160 MB/s [26]. RAID can be used to increase hard disk throughput by treating multiple discs as a single entity and writing or reading from several of them simultaneously. This is referred to as *striping*, and effectively increases the read/write performance of a single disc. The best performance is achieved when each of the drives used in the striping is connected to a separate controller [27]. There are several RAID setups possible. Some of them have the added advantage of mirroring data. This protects information should an individual hard drive fail [28].

The CPU should be rated well above the minimum requirements for both the operating system and the DAQ card. Some points to consider are the number of threads being simultaneously executed in the acquisition and analysis program, and the complexity of the algorithms used in the analysis. When several computation intensive threads need to execute concurrently, a dual processor system should be considered due to its ability to execute two threads simultaneously. If each of the two processors has simultaneous multi-threading technology (named Hyperthreading in Intel processors), then four threads can be simultaneously executed. Data analysis algorithms often require many floating point operations. To reduce the truncation error accumulation from each floating point operation, double precision 64-bit floating point numbers[1] are often used. The Intel Xeon and Pentium IV processors, for example, have 128-bit

[1] a double precision 64-bit number can range from negative 1.79769313486232e308 to positive 1.79769313486232e308

precision floating point registers, which means that two double-precision floating-point numbers can be operated upon simultaneously by a **s**ingle **i**nstruction **m**ultiple **d**ata (SIMD) instruction, thus increasing floating point performance by a factor of about 8 over a processor with 32-bit floating point registers running at the same frequency.

3.3 Portable Raman Lidar 'Polly'

In this section I will describe the development and testing of a **po**rtab**l**e **l**idar s**y**stem (Polly). The focus of my work was the data acquisition system. After a brief overview of the complete system, the components and methods directly relevant to data acquisition will be covered in detail. A comprehensive description of the optical setup can be found in [10].

3.3.1 Concept

The Leibniz Institute for Tropospheric Research (IFT) required a portable and inexpensive (< 100 k€) Raman lidar system capable of the semi-automatic determination of particle backscatter (β^P) and extinction coefficients (α^P) at 532 nm. A Raman lidar as opposed to a single-channel elastic aerosol lidar was desired for its ability to provide the particle extinction profiles $\alpha_\lambda^P(z)$ without assumptions about the extinction-to-backscatter ratio (or lidar ratio) [29], $S_\lambda^P(z) \equiv \frac{\alpha_\lambda^P(z)}{\beta_\lambda^P(z)}$. When assumed constant, this ratio can be a source of up to 100% error [30]. To meet cost requirements, design choices were based on standard components and available hardware. The laser produces green light at 532 nm. At this wavelength, atmospheric transmission is high, but both aerosol and molecular backscattering can still be measured. In addition to the elastic signal detection channel at 532 nm, we included a second inelastic channel at 607 nm for the simultaneous extinction and backscatter determination with the Raman method. This wavelength corresponds to a Stokes vibration-rotation line in the spectrum of the 532 nm light inelastically scattered from nitrogen. The 607 nm spectral line in particular has the largest intensity possible for a line that is still easy to separate from the elastic signal with an inexpensive beam splitter [31]. A laser repetition rate of 15 Hz was selected to allow sufficient signal detection during short measurement intervals of $\approx 10 - 30$ s required for the measurement of turbulent processes in the lower atmosphere.

3.3.2 Setup

The complete portable Raman lidar system Polly is shown in Figure 8. As indicated, it is installed in an outdoor cabinet which will normally be located on the roof of the IFT. The small size and low power consumption of the complete system simplifies measurement campaigns.

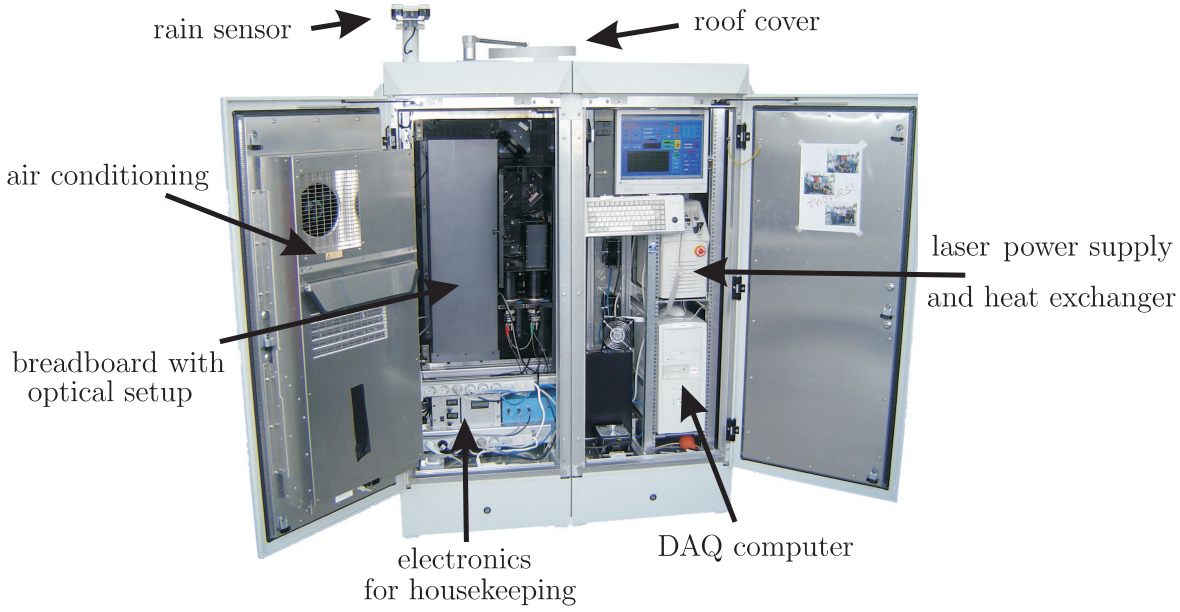

Figure 8: Complete Polly system as of June 2004.

Figure 9 shows the optical layout. The laser (BigSky model CFR200) is a frequency doubled Nd:YAG laser (Nd:YAG = yttrium aluminum garnet doped with neodymium). It emits 120 mJ pulses at 532 nm wavelength with a 15 Hz repetition rate. The beam profile is expanded by a factor of 8 to reduce the divergence below 0.5 mrad. The light is emitted along an axis adjusted by two mirrors to be as close as possible to the receiver optical axis. This helps ensure a complete overlap at low heights.

A Newtonian telescope of 20 cm diameter and 80 cm focal length f_T focuses the captured backscattered light onto a diaphragm with diameter D_D variable between 1 and 3 mm. The receiver field of view $\frac{D_D}{f_T}$ is thus variable between 1.25 and 3.75 mrad. The lens following the diaphragm collimates the light, after which it is deflected to a beam splitter which transmits the 532 nm elastically scattered light and reflects the 607 nm inelastically scattered light. An electronically adjustable cascade of neutral density filters is used to keep the light intensity entering the photomultipliers below the saturation level of the electronics and data acquisition system. Important for data analysis is the relationship between the emitter and receiver optical axes, and the resultant height z_4 calculated from equation (2.1.3) at which the overlap is unity. As the focus of this work is the data acquisition system, no further information regarding the optical setup will be given. For more detailed description of Polly's optical setup, please refer to [10].

Two **p**hoto **m**ultiplier **t**ubes (PMTs, Hamamatsu R5600P) operating in pulse mode are used for detection. They each have a diameter of 8 mm, quantum efficiencies of 7.5% and 1.2% at 532 nm and 607 nm, respectively, and maximum count rates exceeding

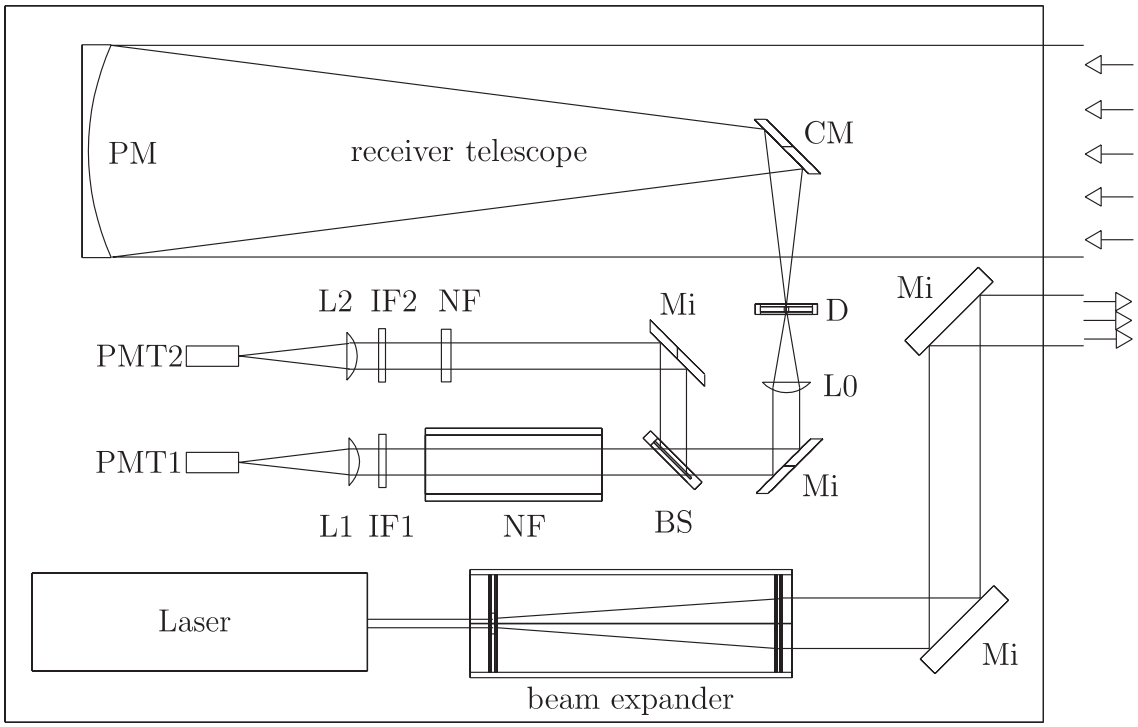

Figure 9: Optical setup of the mini Raman lidar. PM - primary mirror, CM - collecting mirror, D - diaphragm, Mi - mirror, L - lens, BS - beam splitter, NF - neutral-density filter, IF - interference filter, PMT - photo multiplier tube.

10 Mcounts/s. The pulses are amplified and discriminated before being counted by the dual-input multichannel scaler acquisition card (FastComTec 7882). Each time bin (or *channel* of the multichannel scaler) is 250 ns wide. This corresponds to a height resolution of 37.5 m. More detail can be found in subsection 3.3.3.

3.3.3 Electronics

The data acquisition electronics of Polly consists of the two PMTs with follow-up charge integrating pulse amplifiers and discriminators. These devices together produce a pulse for each detected photon. The pulses are counted by a multichannel-scaler card and analyzed by the acquisition program running on the same computer that contains the acquisition card.

Photomultipliers make use of the photoelectric effect, in which a photon ejects an electron from a photosensitive alkali metal, the photocathode [32]. The ejected photon is accelerated onto successive dynodes, ionizing additional electrons at each surface. This *electron multiplication* effectively amplifies the initial photocurrent signal with very low noise [32]. A charge integrator is used to find the total charge contained in the pulse. The discriminator only then produces an output pulse when the charge

contained in the integrated pulse exceeds the threshold corresponding to the detection of a photon. This step is necessary to avoid counting pulses originating from noise within the photomultiplier, normally produced by thermal generation of free electrons at one of the anodes. The rate of counts resulting from internal noise is called the dark counting rate, and our photomultipliers have a dark counting rate of about six counts per second at an operating voltage of −800 V. At room temperature and −800 V, the maximum difference between photon counting rate and dark counting rate was found [10]. Since the arrival of photons at the detectors does not occur at regular intervals, but the intervals are randomly distributed in duration according to Poisson statistics, a number of photons will be rejected during the detection dead time equal to the discriminator's pulse width t_p, which, in our case, has a full width at half maximum (FWHM) of 4±1 ns. This relates the true rate of arrival of photons at the detectors N_T to the measured counting rate N_M, thus setting an upper limit on the allowable counting rate which can be selected according to a chosen maximum error between the true rate of photon arrival and the measured rate. The allowable counting rate can be determined from [33]:

$$N_M = N_T \mathrm{e}^{-t_p N_T}. \tag{3.3.1}$$

If a 5% error is selected, the maximum allowable counting rate N_M is 12 MHz.

Dual-Input Multichannel-Scaler Acquisition Card and its Use in Improving the SNR PMT1 and PMT2 are connected to a two-channel multichannel-scaler data acquisition card. Multichannel scaler cards are time-based counters and signal averagers. Signal averaging is accomplished by the FastComTec 7882 by dividing the time scale after a trigger into a number of 250 ns time bins corresponding to the desired maximum distance of the measurement. Each 250 ns time bin represents a 37.5 m region of the atmosphere, found by calculating the distance traversed by light in 250 ns and halving the result to account for the return path followed by the photons. Since the laser has a repetition rate of 15 Hz, averaging over 10 s would mean summing the corresponding bin counts of 150 shots or one averaging cycle (in this case, one averaging cycle encompasses 150 *sweeps*). When a trigger is detected, the multichannel scaler "sweeps" through the bins, dwelling in each one for 250 ns before advancing. The number of pulses counted as the card dwells in a particular bin is stored in that bin. Each time a new trigger is detected (i.e., the laser is fired), the multichannel scaler returns to the first height bin and sweeps through all height bins as before, adding the new counts for the current height bin to the already there. If the current sweep is the first sweep of an averaging cycle, however, the pulse count for each height bins is reset to the value counted.

By summing sweeps, the true signals from the atmosphere will increase in intensity relative to the random background noise. The fluctuation of a random signal is proportional to $\sqrt{n_b}$, where n_b is the average number of background pulses added to each bin per sweep. After N sweeps, the fluctuation in each bin will be $\sqrt{N n_b}$, whereas

the channels containing a return signal n_s will have an additional Nn_s counts. The ratio between the signal and the background increases by $\frac{N}{\sqrt{N}}$, i.e., the SNR improves with $\sqrt{N}$, and the signal emerges from the background. The random background noise serves only to increase the baseline value.

The FastComTec 7882 multichannel-scaler bins are 32-bit wide. Since our height resolution is 37.5 m, and we have a maximum region of interest of approximately 20 km, we use approximately 2 kB of memory for each input channel. This is well below the available limit of 128 kB total memory. The card is connected to an ISA bus slot, see section 3.2. Assuming the minimum guaranteed clock frequency of 8 MHz, equation 3.2.2 tells us that the transfer time to the PC is approximately 250 µs for the 4 kB of the two inputs combined. The time available between shots for the transfer is an ample 67000 µs.

The computer itself contains a 1.2 GHz Intel Pentium III CPU. Since no time critical calculations need to be performed during data acquisition, this system was deemed sufficient for the tasks assigned to it.

3.3.4 Program Structure and Design

The data acquisition and analysis program was written primarily using National Instruments LabView™ version 6.1. The acquisition card manufacturer, FastComTec, provided the necessary LabView compatible card driver software in the form of a DLL (dynamic link library) for a Microsoft Windows based computer. LabView is a graphical development environment in which program development resembles flowchart design. The resulting simplicity of program development was a reason for basing the control program on LabView. Future modifications to the source code are straightforward even for people with little or no programming experience. A further advantage associated with LabView is the ability to publish the user interface (or *VI* for **v**irtual **i**nstrument) on the Web without modifying the source code, thus permitting control and interaction through a standard Web browser. As Polly will normally be inaccessible to direct control (located on the roof of the IFT), the ability to control data acquisition from a remote location is essential.

A flow chart indicating the data acquisition algorithm is shown in Fig. 10. A measurement cycle comprising a number of sweeps or laser pulses determined by the accumulation period is begun. During this period, the acquisition card accumulates the values of successive shots as detailed in section 3.3.2. When the cycle ends, the accumulated sweep values are saved to the hard drive for later analysis. This process of sweep accumulation and storage is repeated for the duration of the experiment.

For each measurement cycle consisting of a number of shots (typically around 450 shots or 30 s), the height bin counts are accumulated and displayed on-screen. This accumulated pulse count is shown as a function of distance, and the display is updated after every shot. In this way, one can visually adjust the cycle length until the features of

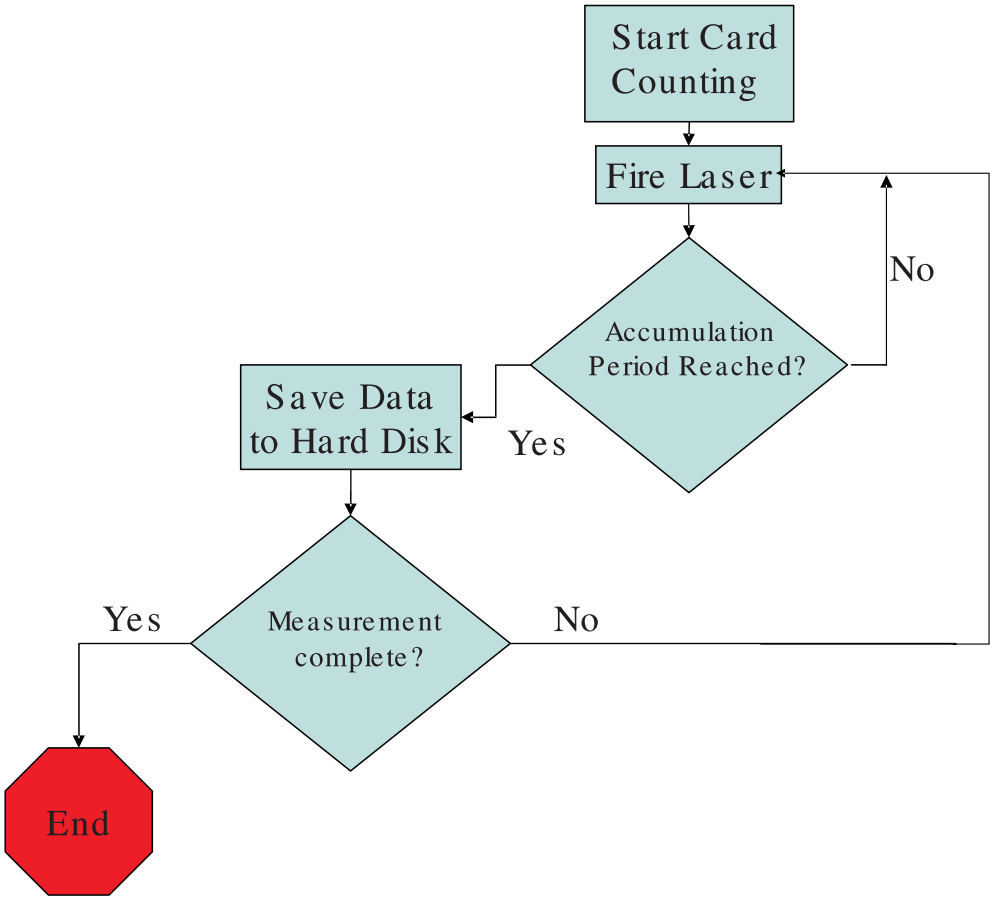

Figure 10: Data acquisition flow chart.

interest appear in the accumulated data. After each measurement cycle, the accumulated shot count is saved. From the saved data, a time history of the measurement is displayed. The height is shown on the vertical axis and the time on the horizontal axis. The accumulated shot count is indicated by the color of the corresponding pixel. For an example of such a time history, see Fig. 11.

In addition to the data acquisition program, there are a number of sub-programs responsible for system control and housekeeping. These sub-programs control the laser (power level, Q-switch, and flash lamps), track the number of laser shots since the last flash lamps replacement, monitor temperature, set the neutral filter density, control the lid on the top of the cabinet, and determine sunrise and sunset times for the current location. Due to the large number of sub-programs, and possibility for future additions, the program source was structured as a state machine (see section 3.1.2).

The flashlamp shot counter program resides on an external microprocessor, an Atmel Mega 8. The shot count is stored on-chip in flash memory, and read into the computer before a measurement is begun. If it is time for flashlamp replacement, the system

operator is given ample warning.

The data acquisition program calculates the peak counting rate in mega-counts per second. This calculation allows the automatic determination of the appropriate neutral filter value required to keep the peak counting rate below 10 Mcounts/s. Based upon this value, the acquisition program calculates the corresponding arrangement of filters in the filter cascade, and this arrangement is set by the external filter cascade control electronics.

A figure showing the user interface of Polly's control program can be found in Appendix A. Attention was given to separating the elements relating to unique aspects of the Polly system into grouped regions, deactivating entry fields that cannot be modified during measurements, and emphasizing controls of particular importance. If the flashlamp shot count exceeds their useful lifetime, a warning message is displayed which prevents the system from being operated.

Sunrise and Sunset Calculations The weak signal strength of Polly's inelastic channel requires the reduction of optical background noise. Since the principle source of optical background is the Sun, the greatest reduction is attained by performing measurements when the Sun is below the horizon. Since Polly is automated, it must have a way of knowing when it can begin measurements. A light detector would be one possibility, but a more elegant solution is to calculate the sunrise and sunset times as a function of date and location. A GPS receiver system can be added to keep date, time, and location variables accurate regardless of location. It is not generally necessary for the calculated times to be more accurate then some number of minutes since the transition from light to dark is gradual. Although sunrise and sunset are officially defined when the center of the Sun is $50'$ below the horizon (zenith$\approx$90.83°), if true nighttime conditions are desired, twilight must be considered. Twilight is the time before and after sunset in which photons scattered from the upper atmosphere, which is directly illuminated by the Sun, illuminate the surface of the Earth. In the astronomical definition, twilight begins when the Sun is 18° below the horizon in the east, and ends when it is 18° below the horizon in the west. For the Raman measurements made by Polly, the astronomical definition is stricter than necessary. The more lenient nautical or civil definitions can be used, which end twilight when the center of the Sun is 12° or 6° below the horizon, respectively.

The sunrise/sunset times can be calculated from the following algorithm [34], [35]:

1. Determine the day of the year T, where January 1^{st} = Day #1, etc.

2. Calculate the equation of time. This equation accounts for the Earth's tilt and elliptical orbit around the Sun by returning the time offset in hours between true solar time and average time based solely upon Earth's rotation period. It can range between $\approx \pm 16$ min. The value of the equation can be estimated to within

1 min of the truth from a numerical fit:

$$\begin{aligned} eqtime &= -0.1752\sin(0.033430T + 05474) \\ &\quad -0.1340\sin(0.018234T - 0.1939). \end{aligned} \tag{3.3.2}$$

3. Determine the solar declination δ in radians (the Sun's "*latitude*" on the celestial sphere). It can be approximated by the harmonic function

$$\delta = 0.40954\sin(0.0172(T - 79.35)). \tag{3.3.3}$$

4. The time Δt in hours for the Sun's apparent position to move from the zenith to an altitude h, where $h = 0$ is the horizon, is calculated from:

$$\Delta t = \pm\frac{12}{\pi}\arccos\left(\frac{\sin(h) - \sin(\delta)\sin(lat)}{\cos(lat)\cos(\delta)}\right). \tag{3.3.4}$$

   Here lat is latitude of the location in question. For sunrise/sunset, $h = -0.833°$, but $-6°$, $-12°$, or $-18°$ can also be used to account for twilight conditions, see above. The positive value of equation (3.3.4) refers to sunset, the negative to sunrise.

5. The sunrise and sunset or twilight times are then calculated in UTC hours from:

$$Sunrise/Sunset = 12 \pm \Delta t - eqtime - \frac{long}{15}, \tag{3.3.5}$$

   here $long$ refers to the longitude of the location in question.

This formalism is implemented in Polly and produces accurate results. The acquisition program will use the generated times to autonomously conduct nighttime experiments.

3.3.5 Experimental Tests and Results

In order to demonstrate the capabilities of Polly, a few measurement examples are shown below. Figure 11 shows the evening development of the **p**lanetary **b**oundary **l**ayer (PBL), the lowest layer of the troposphere, in the evening of March 29th, 2004. This region contains a large number of particles and thus returns a strong backscattered signal. As rising thermals from the Earth's surface decrease with the setting Sun, the thickness of the PBL decreased with time from about 1.8 to 1.4 km. The opposite effect can be observed in the morning, when an increasing temperature is accompanied by an increasing PBL thickness. Turbulent processes near the top of the PBL are of particular interest to meteorologists since they determine the interaction between the particle-rich air of the PBL and the relatively clean air of the free troposphere. Aerosols transferred into the free troposphere, where the horizontal wind speeds are generally high, can be transported around the world within days. The measurement of PBL turbulent processes can be accomplished with a wind lidar system such as the one discussed in the next section.

Polly can measure signals far above the PBL. In Fig. 12, the formation of cirrus clouds in the free troposphere between 5.5 and 8 km on April 22nd, 2004, can be seen. The signatures indicate falling ice crystals. Between 3.5 and 4.5 km, altocumulus clouds were occasionally present. The PBL extended up to 2 km at the beginning of the measurement.

A calculation of the backscatter and extinction coefficients by the Raman method, along with the range-corrected elastic channel data used in the calculation, can be seen in Figs. 13 and 14. Both of these calculations use data acquired in the night of August 30th, 2003. In Fig. 13, PBL aerosol layers and lofted aerosol layers can be seen up to a height of 4 km. These layers probably resulted from the forest fires at that time in southern Europe. Extinction and backscatter values were determined from a 1 h time average. The backscatter coefficient can be calculated with a resolution of 300 m and an error of < 5 % throughout the PBL (thin line). For an estimate of extinction profiles and the extinction-to-backscatter ratio with $< 25\%$, a smoothing of the profiles with 600 and 1200 m below and above 1400 m, respectively, was necessary. The extinction profile is shown above 1100 m only, were the effects of the incomplete overlap are negligible. The lidar ratio was about 30 sr below 2 km and around 70 sr above, indicating different types of aerosols in these height regions.

A cirrus cloud between 7.5 and 9 km that was present for 25 min can be seen in Fig. 14. With smoothing lengths of 900 and 1200 m below and above 8500 m, respectively, it was possible to determine the lidar ratio profile of this cirrus cloud. Typical values for larger ice crystals around 15 sr were found in the upper part of the cloud above 8500 m, whereas larger values around 40sr in the lower part indicate small, evaporating crystals.

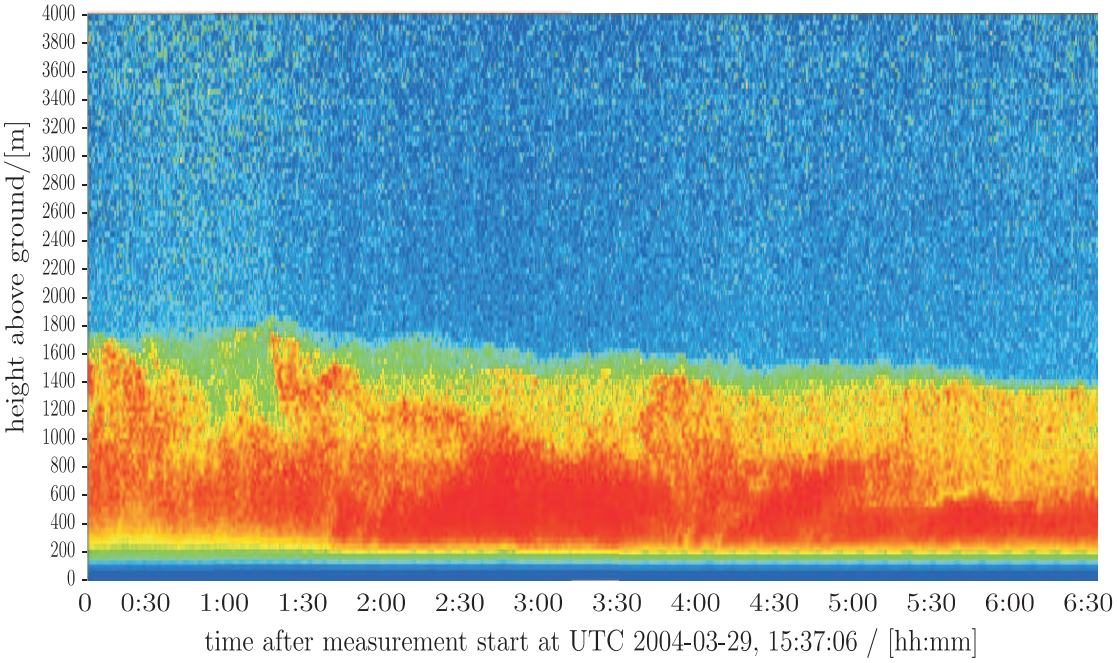

Figure 11: PBL height development at Leipzig in the evening of March 29th, 2004. Time resolution: 30 s, height resolution: 75 m

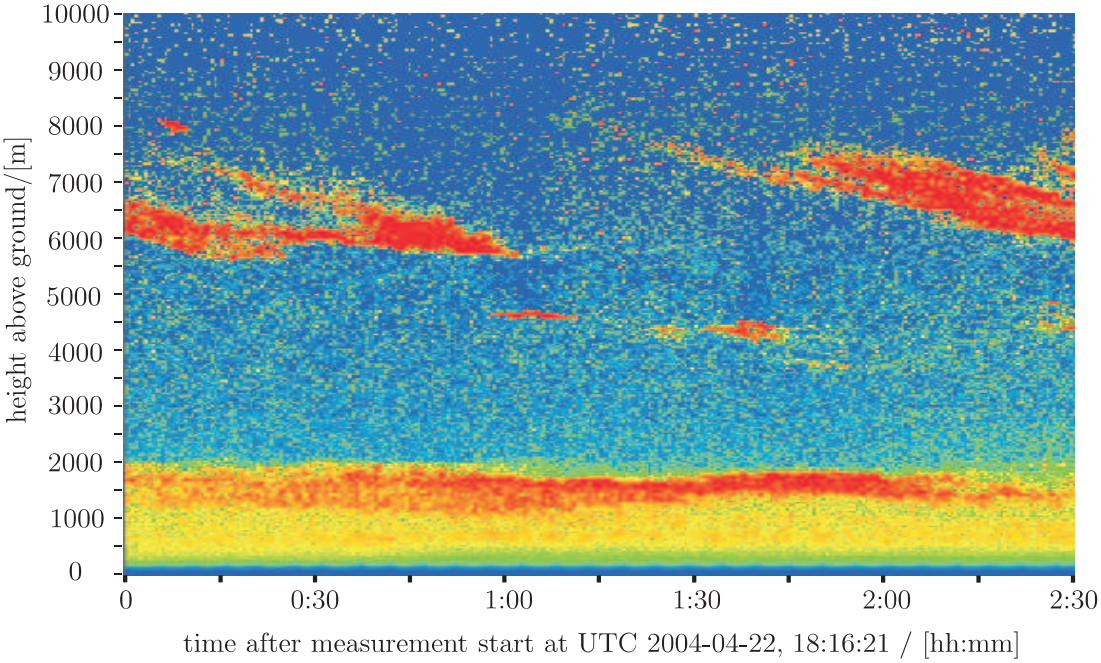

Figure 12: Nighttime cirrus cloud formation and development on April 22nd, 2004. Time resolution: 30 s, height resolution: 75 m

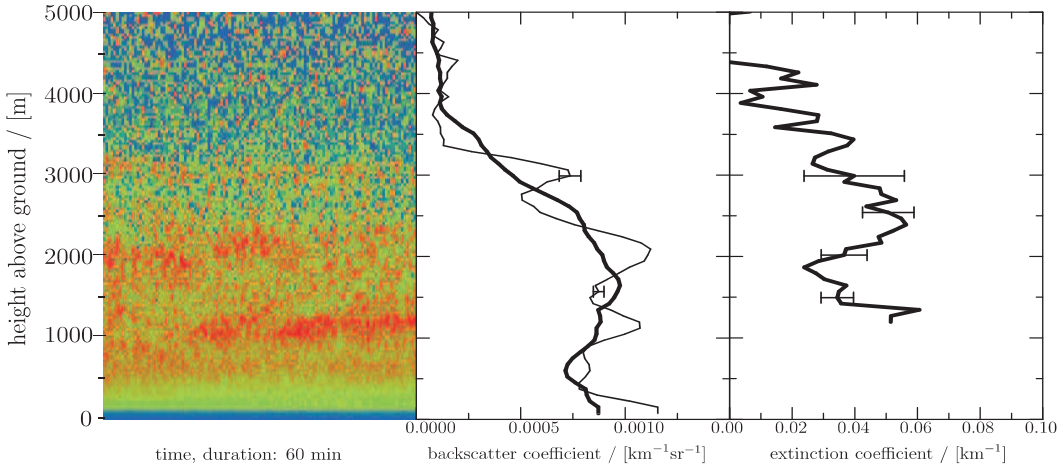

Figure 13: Nighttime measurement begun on August 30$^{\text{th}}$, 2003, at 23:13 UTC showing the range-corrected elastic channel data, and the backscatter and extinction coefficient profiles averaged over the 1 h duration of the measurement. The reference backscatter coefficient was chosen to be 0.0001 km^{-1}sr^{-1} at 5400 m, where no significant aerosol signal is measured. The thick lines were smoothed over 600 m below 1400 m and 1200 m in the higher regions. The high resolution backscatter profile indicated by the thin line was smoothed over 300 m. Since incomplete overlap affected the measurements up to 1100 m, the extinction profile begins at this height.

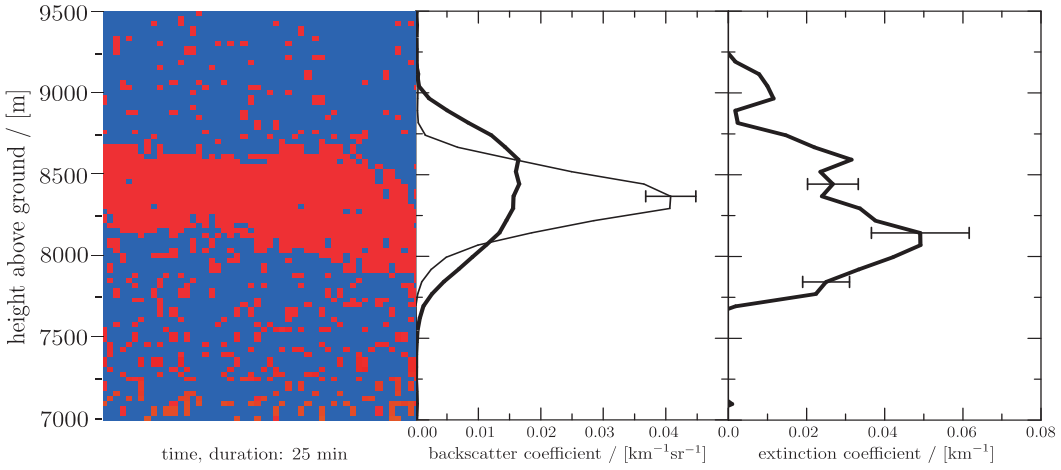

Figure 14: Nighttime measurement from August 30$^{\text{th}}$, 2003, at 23:30 UTC showing the range-corrected elastic channel data, and the backscatter and extinction coefficient profiles averaged over 25 min. The reference backscatter coefficient was chosen to be 0.0001 km^{-1}sr^{-1} at 4800 m. For the thick lines, the profiles were smoothed over 900 m below 8500 m and over 1200 m in the higher ranges. The high resolution backscatter profile indicated by the thin line was smoothed over 300 m.

3.4 Wind Lidar

In this chapter I describe the development of the data acquisition system and analysis software for the wind lidar. The description begins with a brief introduction to the complete system, followed by a closer look at hardware components directly related to data acquisition and analysis. Following this, I will explain the important algorithms and structures in the acquisition and analysis program.

3.4.1 Setup

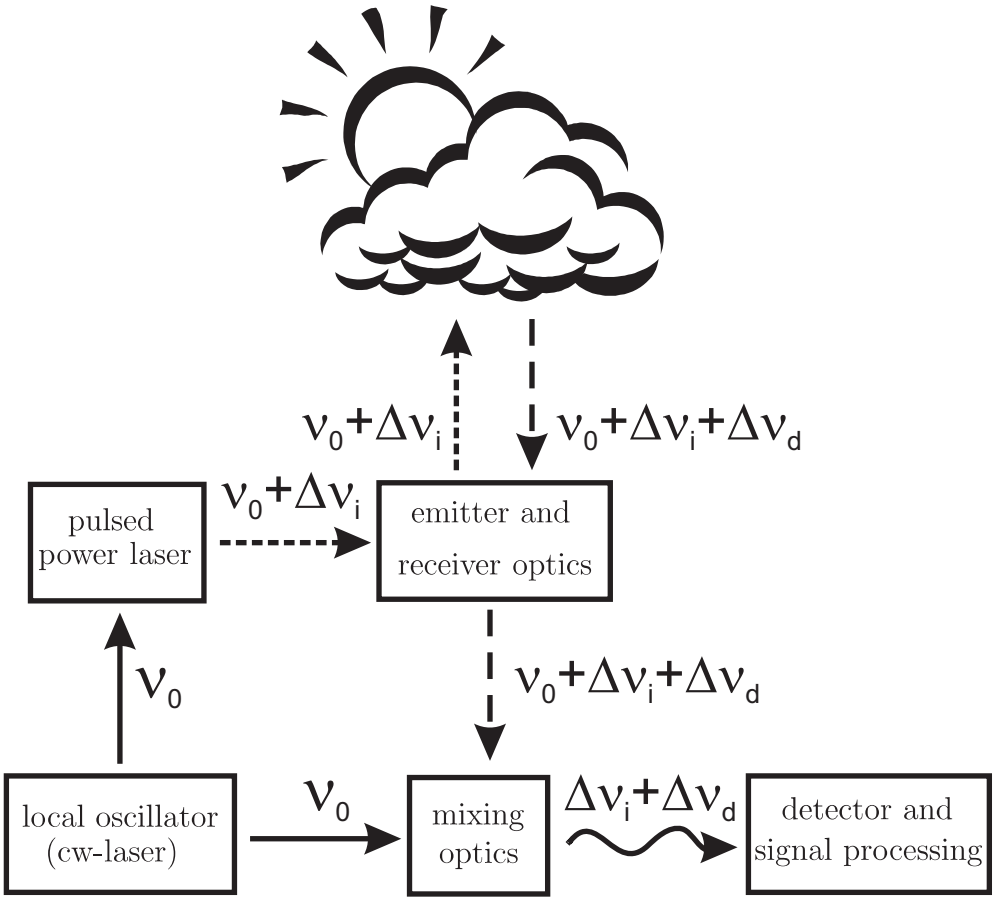

Figure 15: Schematic setup of the wind lidar system.

Prerequisite to the understanding of the tasks required of the data acquisition system and analysis program is a basic knowledge of the complete system. To this end, I will comment on the schematic system overview shown in figure 15. This figure portrays the heterodyne measurement method used in the wind lidar to measure atmospheric wind speeds (see section 2.2.1 for an in-depth treatment). A local oscillator creates the reference frequency ν_0. The local oscillator is a **c**ontinuous **w**ave (cw) laser operating

in the TEM00 mode. Part of the radiation from the local oscillator is used in the *injection seeding* of the power laser, which is a pulsed high-energy laser whose output is emitted into the atmosphere. Injection seeding is used to select a specific mode in the power laser and thus couples the power laser to the local oscillator. The injection is accomplished via an acousto-optical modulator. The acousto-optical modulator offsets the frequency of the photons through Bragg reflections on propagating sound waves before injecting them into the power laser. The frequency of the injected photons thus differs from that of the local oscillator by a constant offset, or *intermediate frequency*, of $\Delta\nu_i$. The frequency of the radiation emitted into the atmosphere is then $\nu_0 + \Delta\nu_i$. The spectral width of this frequency line sets a minimum to the smallest measurable Doppler shift, and depends upon the pulse length Δt_P. Optimal pulse length is selected by considering both the desired spatial resolution and the desired frequency resolution $\Delta\nu_{min}$, or the desired smallest measurable Doppler shift. The time-bandwidth product is the deciding quantity. If the pulse is very long, spatial resolution is lost as we cannot determine from what location within the volume illuminated by the propagating pulse the backscattered photons originate. The time-bandwidth product (TBP) is in general calculated from [36]:

$$TBP = \frac{\Delta t_P}{\tau_c} = \Delta\nu_{min}\Delta t_P, \tag{3.4.1}$$

where $\tau_c = 1/\Delta\nu_{min}$ is the coherence time or the temporal length of the pulse. For a Gaussian pulse profile in the time domain, $TBP \approx 0.441$. If the pulse length were 400 ns, corresponding to a spatial resolution of 60 m, then $\Delta\nu_{min}$ is 1.1025 MHz.

Part of the energy from each pulse emitted into the atmosphere is backscattered by particles. If the volume of air in which the backscattering occurs is in motion along the propagation direction of the pulse, the scattered light is shifted relative to the incident light by $\pm\Delta\nu_d$, where $+$ is for positive (toward observer) and $-$ for negative (away from observer) wind speeds. The receiver optics captures the backscattered photons, now of frequency $\nu_0 + \Delta\nu_i \pm \Delta\nu_d$, and passes them to the mixing optics where they are coherently mixed with radiation from the local oscillator of frequency ν_0. The mixed signal of frequency $\Delta\nu_i \pm \Delta\nu_d$ is detected by a photodetector, amplified, and sent to the data acquisition system for processing. Details of the optical setup can be found in [10] and [37].

3.4.2 Electronics and Data Acquisition

The first stage of my project involved the selection of appropriate data-acquisition hardware. The hardware encompasses an acquisition card and computer. For the selection of the acquisition card, the source and type of signal to be measured need be considered. Important here is that the system contains two photodiodes. One is used to detect the offset frequency of the outgoing pulse, and the other the atmospheric return signal. The data acquisition system must switch from the offset to the atmo-

spheric return channel after an adjustable delay of $10^{-5} - 10^{-6}$ s. Since no two-channel acquisition card could be found which was comparable to the single-channel PDA1000 from Signatec in terms of DMA transfer rate and noise floor, we decided to perform the channel switching with an external circuit and purchase the PDA1000.

The PDA1000 is an 8-bit analog-to-digital converter which can sample at a maximum rate of 1 GHz. It connects to the 64-bit PCI bus and can transfer data stored in its 256 MB of on-board memory at a peak rate of 266 MB/s. The number of actual data points B_R required for a single shot with the wind lidar is

$$B_R = F_{samp} T_{offset} + \frac{2D_{max}}{c_0} F_{samp}, \tag{3.4.2}$$

where D_{max} is the furthest distance to be measured in [m], F_{samp} the acquisition rate in [Hz], T_{offset} the time duration of the frequency offset sampling in [s], and c_0 the speed of light in [m/s].

The PDA1000, however, requires that the total number of bytes B_T transferred from the card's on-board RAM to the PC be an even multiple of 4096 B. The amount of data to be transferred will therefore be larger than B_R in general, and is calculated from

$$B_T = \begin{cases} B_R, & \text{if } B_R \bmod 4096 = 0, \\ B_R - B_R \bmod 4096 + 4096, & \text{if } B_R \bmod 4096 \neq 0. \end{cases} \tag{3.4.3}$$

where the "mod" operator is defined such that $x \bmod y = x - \frac{x}{y} * y$.

To prevent the computer from transferring unnecessarily large volumes of data over the bus, and to keep the amount of data to be processed to a minimum, the expected frequency range of the signal needs to be known. The sampling rate of the card can then be selected accordingly. In this wind lidar system, zero wind speed corresponds to an intermediate frequency of 80 MHz. Accounting for absolute wind speeds of 20 m/s, frequencies ranging from 60 to 100 MHz should be detectable with the acquisition system. With these considerations, we run the card at an acquisition rate of 250 MHz. This places the Nyquist frequency at 125 MHz, sufficiently above the largest expected frequency of 100 MHz to allow for the proper sampling of wind signals while simultaneously reducing the load on the data acquisition system by a factor of four as compared to a 1.0 GHz sampling rate. To allow for the future addition of an anti-alias filter, the frequency band from 100 to 125 MHz should not be used due to filter roll-off and possible aliasing effects from frequency components within the tail of the filter.

The time required to transfer the sampled data points from the acquisition card to the computer is an important factor in determining the maximum shot repetition rate. It results from the necessity of transferring all data out of the card's RAM before new data is acquired, and can be calculated from equations (3.4.2) and (3.4.3). In a typical

measurement sampled at 250 MHz and extending to 10 km, with 1 µs reserved for offset determination, the total amount of data B_T to be transferred to the computer's RAM each shot is 20480 B. Entering this value along with the PDA1000's specified peak PCI transfer rate of 266 MHz into equation (3.2.2), it can be seen that a minimum transfer time of 9.6 µs is required to move the data into the computer's RAM. This is well within the 2 ms time limit posed by the targeted 500 Hz laser trigger frequency.

Referring back to section 2.2.3, it can be seen that digitizing the signal with 8-bit resolution adds an rms-noise component equal to $\frac{1}{900}V_{pp}$. Since we are expecting large SNRs in the regions of interest, this added noise component should not significantly affect signal detectability.

3.4.3 Program Structure

To meet the desired laser shot repetition rate of 500 Hz it was necessary to design the program with maximum execution speed and efficiency as the primary goal. In a typical measurement, a height range of 5 km with 20 m resolution is studied. The control program would then have to perform an FFT for each height bin of 20 m. In this case, 250 FFTs would be necessary to analyze the atmospheric return signal. In addition, one large (i.e., 256 point) FFT is performed on the outgoing pulse to measure the frequency offset between the master and power oscillator. Since all these FFTs have to be performed at 500 Hz, the total number of FFTs per second becomes approximately 125500. Depending on the desired height resolution and data acquisition frequency, a minimum 32- or 64-point FFT must be calculated for each height bin. For a desired spatial resolution R_Z, the number of sampled points N_S required can be calculated from the following equation:

$$N_S = \frac{c_0}{2\nu_s} R_Z, \tag{3.4.4}$$

where ν_s is the sampling rate. Due to zero padding, however, the input array to each FFT is in general larger than N_S.

For 25 m resolution, eq. (3.4.4) indicates that 15 sampled points are sufficient. Only one zero need be padded to make the FFT input array length the required power of two. Sixteen points, however, are insufficient for accurate frequency determination. As many zeros as possible should be padded, but computer resource and calculation time considerations limit the practical FFT input size in this system to 1024. The decision is left entirely open to the experimenter, but will ultimately depend upon shot repetition rate and processed height range. In the program, the desired height resolution is entered in meters, along with the FFT input size. The user is passively informed of the number of zeros that will be appended to the sampled points before being processed by the FFT algorithm.

To analyze the atmospheric return signal, the program performs the following operations:

1. The acquisition card waits for a trigger and acquires the number of points required for the requested maximum height, plus a number of extra points for offset determination.

2. The offset of the outgoing pulse is determined and subtracted from a reference offset (the reference offset is chosen to equal the offset of the first shot of the measurement). If the difference is within the allowable tolerance specified by the user (generally 100 kHz or 10 cm/s), then processing continues. Otherwise the shot is rejected and the card waits for the next trigger. Alternatively, the power spectra of all height bins for the shot is shifted by the difference such that the offset frequencies align. The choice depends on the stability in the offset frequency.

3. The power spectrum of each height bin is calculated and added to a running total for that height bin.

4. If the number of shots to average has been reached, the peak frequency of each height bin is determined from the summed spectra by finding the highest point of an inverted parabola fitted to the highest bin and its left and right neighbors, see section 3.4.4. The running totals are reset to zero.

5. The reference offset is subtracted from each peak frequency to create the array of the Doppler shifts ($\Delta\nu_d$ values) corresponding to wind speeds.

6. The array of Doppler shifts is written to the hard disk and displayed.

To complete the above tasks at the required repetition frequency for large height ranges, it was necessary to multithread the program. In total, twenty-six threads were assigned to the task. These threads can be classified into six groups:

1. Three threads are used for data acquisition. One runs the card, the other moves the data from the card to the PC, and a third copies the data to the data analyzer.

2. One thread determines the offset of the outgoing pulse and either rejects shots exceeding the allowable tolerance, or determines the frequency shift required to align the offset with the reference offset.

3. Sixteen threads process the acquired data to determine the power spectra of each height bin, and, if need be, perform shot accumulation for time averaging.

4. Four threads process the power spectra for their peak frequencies after the requested number of shots have been accumulated.

5. One thread saves the array of peak frequencies to the hard disk along with the current UTC time, height range, number of shots averaged per cycle, and height resolution.

6. A final thread displays the data as a vertical colored column, where the color represents the speed, the vertical axis the height, and the horizontal axis the time.

Multithreading the application in this manner effectively increased the program's maximum processing frequency from 90 Hz to the card's apparent maximum acquisition rate of 450 Hz.

In addition to processing atmospheric return, it is necessary to characterize the jitter of the power oscillator from the stability in the measured offset frequency of the outgoing pulse. For this purpose, further functionality was added to the program. This function performs the following operations:

1. The user enters the number of points that are to be used in the jitter determination.

2. The card waits for a trigger and acquires the data.

3. A maximum 16384-point FFT is performed on the acquired data, which is windowed and padded with zeros as required or specified.

4. The peak frequency is found from the FFT from any one of five peak finding algorithms: quadratic fit, Quinn's first or second estimator, barycentric, or Jain's method, see section 3.4.4.

5. The array of peak frequencies is analyzed to find the mean and standard deviation, and a distribution array of user-specified length is constructed. The bins of the distribution array contain the number of measured frequencies which fall within the bins frequency range. The first bin of the distribution begins at the lowest measured frequency, the last bin ending at the highest frequency.

6. Statistical analysis is performed on the distribution (i.e., the mean frequency and standard deviation are calculated), and a normal distribution with the same mean and standard deviation is calculated for visual comparison purposes.

7. Both the measured distribution and the calculated normal distribution are displayed in the same window.

8. The data is stored to the hard disk, with the options of saving the distribution, the values of all calculated frequencies, and the results of the statistical calculations.

Summing the Power Spectra of Shots with Differing Offset Frequencies In any measurement requiring shot-to-shot accumulation, accumulating shots whose offsets differ from the reference offset by more than the minimum desired frequency resolution or specified shot tolerance will certainly decrease frequency resolution if not ruin the data. This results when the frequency spectrum of the *same* wind speed in the *same* range bin is shifted when measured by shots with *different* offsets. The magnitude of the dislocation is equal to the magnitude of the difference between the offset frequencies. One solution to this problem is to discard shots whose offsets are out of range. This solution requires the least computing overhead, but is unacceptable if the shot-to-shot offset stability is poor, or if the offset gradually drifts over time. If the measured offset frequencies follow a Gauss distribution, and the measured standard deviation of the distribution is comparable to or larger than the desired minimum frequency resolution, rejecting shots with out-of-range offsets results in an unacceptably high rejection ratio. For example, if the mean offset frequency is 80 MHz, and the standard deviation is equal to the tolerance of 0.1 MHz, then 31.7% of the shots will be rejected. Before these shots can be used in accumulation, however, their power spectra must be shifted such that their offset frequencies correspond with the reference offset. For this shifting to be feasable, the bin spacing in the frequency domain must be on the order of the desired minimum frequency resolution. If a speed resolution of 10 cm/s is desired, this corresponds to 100 kHz. Such bin spacing can be achieved by performing a 2500-point FFT. Since the FFT routine requires an input size equal to a power of two, the closest size is 2048. This leads to a frequency-domain bin spacing of 122 kHz, or a best-case speed resolution of 12.2 cm/s. This method of shifting the spectra increases the number of calcuations required in each FFT to 22528 from the 384 or 160 that would have been necessary for a 64- or 32-point FFT, respectively. This means an approximate 140-time increase in the number of calculations required of a 32-point FFT, and a corresponding decrease in the height range that can be calculated at the same frequency.

3.4.4 Peak Finding Algorithms

There are several methods to find the largest frequency component of a sampled signal from its transform. The fastest and least accurate is simply to assume that the frequency domain sample of greatest amplitude precisely represents truth. This method has the serious disadvantage of preventing the accurate determination of frequency peaks not exactly corresponding with one of the basis vectors in the frequency domain, i.e., when the signal frequency component lies between bins. The maximum error is then $\pm\frac{1}{2}$ the spectral bin width. In the case of a 64-point FFT sampled at 125 MHz, this error would be 1.89 MHz. Since we desire knowledge of the true frequency within $\pm$100 kHz, the fastest method is of little value in frequency spectra where the bin width is not at most on the order of 100 kHz. A 2048-point FFT reduces the bin width to 122 kHz, a value acceptably close to 100 kHz. In FFT spectra with larger bin spacing,

a peak finding algorithm must be applied to the frequency domain samples to interpolate the true location of the peak. Fundamental to all the peak finding algorithms examined here is the property that inter-bin frequencies exhibit spectral leakage and thus have non-zero frequency components across all frequency samples. The amplitude of the frequency samples diminishes with distance from the actual frequency location, and thus only the amplitudes of frequency samples near truth need be used to determine the location of the frequency peak. In this work, I have investigated five such algorithms: quadratic fit, Quinn's first and second estimators, Jain's method, and the barycentric method.

The algorithms themselves are shown below. In the notation used here, k represents the index of the frequency domain sample with the largest amplitude in the frequency range under study, $X[n]$ represents the complex n^{th} bin of the FFT, where $|X[n]|$ is the magnitude, $X[n].r$ and $X[n].i$ are the real and imaginary components, respectively, and k' is the interpolated bin location.

Quadratic Method

1. $y_1 = |X[k-1]|$
2. $y_2 = |X[k]|$
3. $y_3 = |X[k+1]|$
4. $d = \frac{(y_3 - y_1)}{(2(2y_2 - y_1 - y_3)}$
5. $k' = k + d$

Barycentric method

1. $y_1 = |X[k-1]|$
2. $y_2 = |X[k]|$
3. $y_3 = |X[k+1]|$
4. $d = \frac{(y_3 - y_1)}{y_1 + y_2 + y_3}$
5. $k' = k + d$

Quinn's First Estimator

1. $a_p = \frac{X[k+1].rX[k].r + X[k+1].iX[k].i}{X[k].rX[k].r + X[k].iX[k].i}$
2. $d_p = \frac{-a_p}{1 - a_p}$

3. $a_m = \frac{(X[k-1].rX[k].r+X[k-1].iX[k].i}{X[k].rX[k].r+X[k].iX[k].i}$
4. $d_m = \frac{a_m}{1-a_m}$
5. if $(d_p > 0)$ and $(d_m > 0)$ then $d = d_p$ otherwise $d = d_m$
6. $k' = k + d$

Quinn's Second Estimator

1. $\tau(x) = \frac{1}{4}\lg(3x^2 + 6x + 1) - \frac{\sqrt{6}}{24}\lg\left(\frac{x+1-\sqrt{\frac{2}{3}}}{x+1+\sqrt{\frac{2}{3}}}\right)$
2. $a_p = \frac{X[k+1].rX[k].r+X[k+1].iX[k].i}{X[k].rX[k].r+X[k].iX[k].i}$
3. $d_p = \frac{-a_p}{1-a_p}$
4. $a_m = \frac{X[k-1].rX[k].r+X[k-1].iX[k].i}{X[k].rX[k].r+X[k].iX[k].i}$
5. $d_m = \frac{a_m}{1-a_m}$
6. $d = \frac{d_p+d_m}{2} + \tau(d_p^2) - \tau(d_m^2)$
7. $k' = k + d$

Jain's Method

1. $y_1 = |X[k-1]|$
2. $y_2 = |X[k]|$
3. $y_3 = |X[k+1]|$
4. if $y_1 > y_3$ then:
   - $a = \frac{y_2}{y_1}$
   - $d = \frac{a}{1+a}$
   - $k' = k - 1 + d$

   otherwise:
   - $a = \frac{y_3}{y_2}$
   - $d = \frac{a}{1+a}$
   - $k' = k + d$

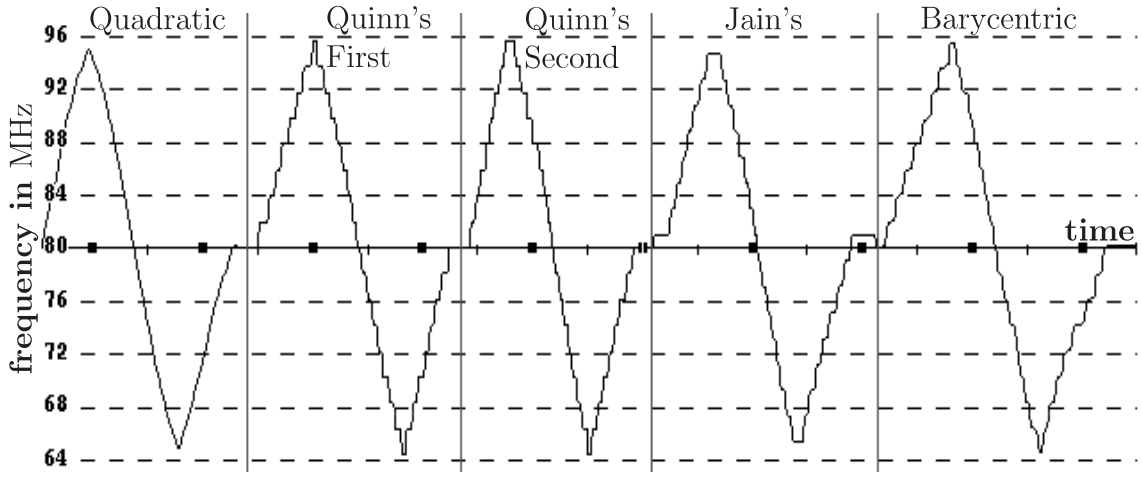

Figure 16: Comparison of five peak finding algorithms on a 128-point FFT analysis of an input frequency varied between 65 and 95 MHz.

In the study shown in Fig. 16, I continuously varied the sampled frequency from 65 to 95 MHz, changing the peak finding algorithm at each positively sloped crossing of the 80 MHz line. Although each method truly interpolates a peak frequency that lies between bins, all but the quadratic method display "steps" in the output shown in Fig. 16. The size of the steps seen in Fig. 16 displayed no dependency on the chosen peak finding algorithm (excluding the quadratic method), and was equal to the frequency bin spacing. What did vary, however, was the offset of the step levels relative to the frequency domain samples. The existence of pronounced steps, or bias in the interpolation algorithm, in all but the quadratic method led to my selection of this method of interpolation. The second choice would be the barycentric method, but this showed very poor mid-bin performance. Both Jain's method and Quinn's first estimator, which contain branching statements, exhibit additionally a large standard deviation for frequencies which cause the branching statement argument to lose meaning and randomly oscillate. In Tab. 3, the performance of the various peak finding algorithms can be compared.

3.4.5 Experimental Tests and Results

The experimental test results discussed in this section primarily relate to software performance analysis. In addition to performance tests, the software was used to determine both the stability in the offset frequency of the power oscillator and the effectiveness of the circuit intended to stabilize the offset. This circuit adjusts laser cavity length with piezo-electric crystals. The standard deviation in the stabilized offset frequency is used to determine the method of shot accumulation, namely if the power spectra of a large portion of shots require pre-accumulation offset correction, as discussed in section 3.4.3.

Table 3: Comparison of various peak finding algorithms. To generate the mean μ and standard deviation σ, 5000 32-point FFTs were calculated for each entry. For the first column, the frequency generator was set to 31.25 MHz. This frequency corresponds with a frequency sample, since the acquisition rate was 125 MHz. The generator's frequency was set to 33.203125 MHz for the second column. This value lies $\frac{1}{4}$ of the distance between bins. For the final column, truth was 35.15625 MHz, which lies exactly between samples in the frequency domain. A four-term Blackman-Harris window was applied to the data in all cases. Frequency samples are spaced by 7.8125 MHz.

Peak Finder	μ **and** σ **in** MHz		
	Truth: 31.250 MHz, on sample	Truth: 33.203 MHz, sample $+\frac{1}{4}$ bin	Truth: 35.156 MHz, mid-bin
quadratic	31.23, 0.220	32.67, 0.130	35.16, 0.060
Quinn's 1^{st}	28.09, 0.120	28.45, 0.150	failed
Quinn's 2^{nd}	31.25, 0.120	31.98, 0.140	35.41, 2.500
Jain's	31.13, 2.570	34.42, 0.010	35.16, 0.010
barycentric	31.24, 0.240	32.69, 0.230	35.27, 1.100

Performance Test I: Dependence of the Analysis Frequency on the Spatial Extent of the Profile In Fig. 17, the software's maximum analysis frequency in dependence on the spatial extent of the profile to be analyzed is plotted for 64-point FFTs. The maximum analysis frequency was estimated by increasing the trigger frequency until the program's "unprocessed shot counter" began to rapidly increase. The largest trigger frequency before the sharp rise in the number of unprocessed shots was taken to be the maximum analysis frequency. Keeping all other parameters the same, doubling the length of the profile should double the number of calculations and thus double the time required for profile analysis. Since time is inversely proportional to frequency, an inverse relationship between maximum analysis frequency and profile length was predicted. As expected, an inverse function fit well with the data. From the determined fit function also shown in Fig. 17, the time required for the calculation of a single 15 m height bin with a 64-point FFT is (0.35 ± 0.15) ms.

Performance Test II: Dependence of the Analysis Frequency on the FFT input size This test is important for the offset correction algorithm, which requires the bin spacing in the power spectrum of each height bin to be on the order of the acceptable tolerance (see section 3.4.3). The number of calculations, and thus the calculation time, for a single FFT is proportional to $N\log_2 N$. As such, the analysis frequency should be inversely proportional to the same value. Fig. 18 shows the measured maximum analysis frequency against $(N\log_2 N)^{-1}$. Contrary to expectations, a single linear equation does not fit the dataset very well. This might be related to the

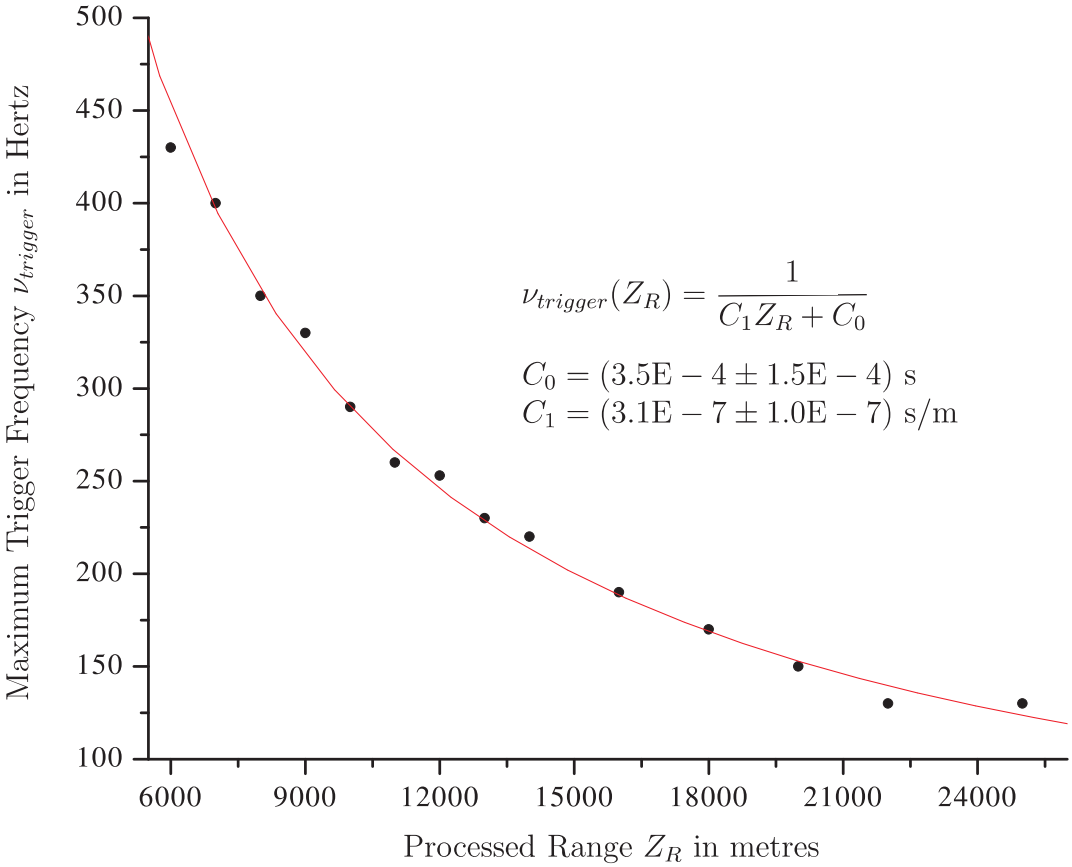

Figure 17: 64-point FFTs were performed on each 15 m height bin in the distance ranges indicated. The maximum captured trigger frequency is indicated on the vertical axis.

increased time required to move data into and out of the FFT subroutine, the added overhead related to memory allocation as array size increases, and power spectrum calculations. As such, I separated the plotted points into two apparently distinct linear regions. The performance rapidly decreases as array size increases in the linear region where $N > 64$. By extrapolation, a 1024-point FFT calculating the same 10 km region could process a maximum trigger frequency of 95 ± 18 Hz.

Effects of Zero Padding on the Standard Deviation and Measurement Accuracy Table 4 indicates the effects of zero padding on both the standard deviation σ and the accuracy of the frequency determination, where accuracy is defined as the difference between the true frequency and the mean. The true frequency was given by a digital function generator with specified 8-digit frequency accuracy outputting a 50 MHz sine wave. The total input size of the FFT was varied between 16 and 2048, where in each case only 16 of the input points were real sampled data points. After acquisition, the 16 points were windowed by a 4-term Blackman-Harris apodization

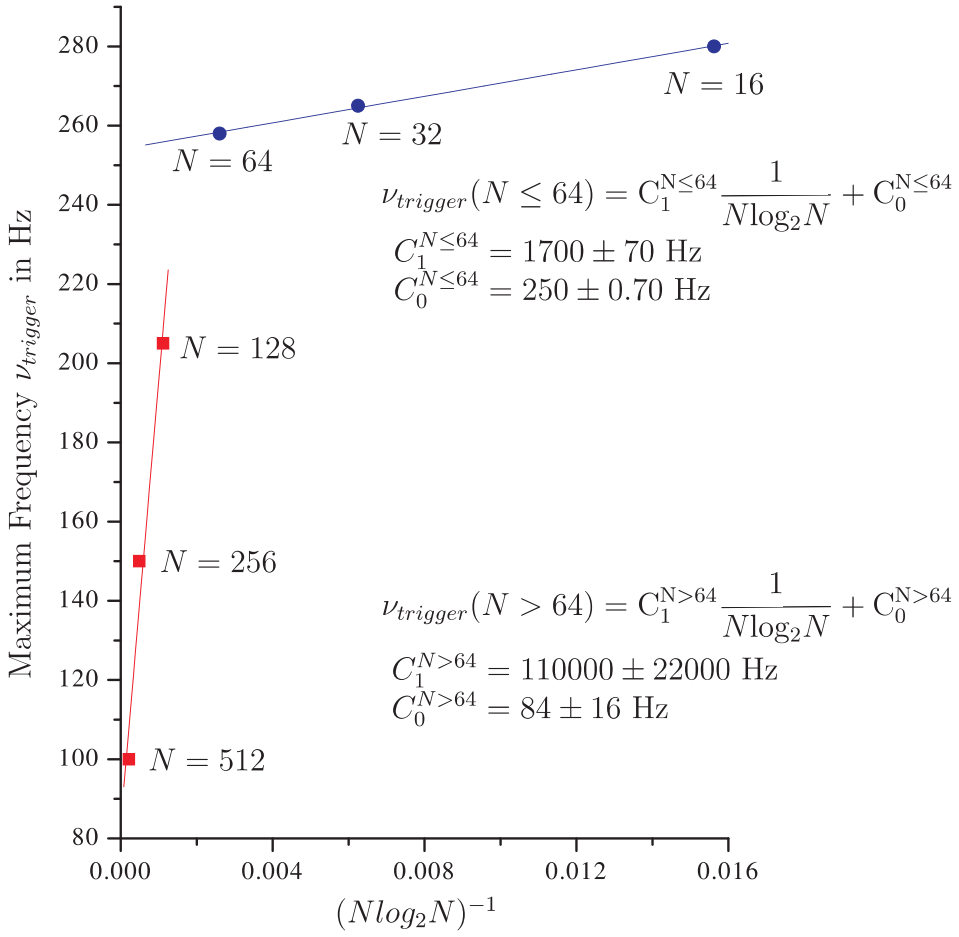

Figure 18: N-point FFTs were performed on all 672 height bins (spanning 10 km). The maximum captured trigger frequency is indicated on the vertical axis.

function before being processed by the FFT algorithm. As can be seen, both the standard deviation and the accuracy were improved by zero padding. After padding the 16 sampled points with 48 zeros, however, no further improvement was achieved in the accuracy and further improvements in the standard deviation were minimal. Performing the same measurements without first windowing the data caused the standard deviation to increase by approximately three orders of magnitude in all cases where zeros were padded. Looking directly at the power spectrum, the reason became obvious. The spectral leakage from the discontinuity onto the basis vectors near the true frequency was large and unpredictable in nature. This caused the quadratic peak finder algorithm to perform poorly since the amplitudes used to determine the peak location were not dominantly caused by the true frequency, but were modified by the addition of the large spectral leakage components arising from the discontinuity. Since mismatching and the height of the discontinuity change from shot to shot in an unpredictable fashion (determined primarily by the phase of sampled waveform), the spectral leakage also

varied and the resultant standard deviation was large.

Table 4: Determining the effects of zero-padding on frequency determination. For each row of the table, the outcomes of 10000 FFTs were used in the statistical calculations. The total input size to the FFT consisted of 16 measured data points in addition to the zeros, and was thus a power of two. The measurements were performed on a 50.000000 MHz, 200 mV peak-to-peak sine wave. A four-term Blackman-Harris window was used, and the peak was interpolated with a quadratic fit.

Padded Zeros	σ **in** kHz	**Accuracy in** MHz
0	3471	−2.336
16	1587	−0.563
48	1172	+0.235
112	1095	+0.236
240	1080	−0.237
496	1075	−0.222
1008	1077	−0.243
2032	1078	−0.228

Dependency of the Accuracy and Standard Deviation on the Number of Signal Samples A curious phenomenon was observed during the dependency measurement of the accuracy and standard deviation on the number of sampled points N_S included in an FFT of set size N, as shown in Fig. 19. Although the standard deviation tended toward zero as the number of measured points in the 256-point FFT increased, the measured frequency simultaneously moved further from truth. For the trend in the standard deviation, measured data points were well fit by the function:

$$\sigma = \frac{const.}{N_S^2}, \tag{3.4.5}$$

where $const. = 528000$ kHz. The shift of the mean measured frequency from the true frequency was measured and the corresponding accuracy calculated. The trend in the accuracy is represented well by the second-order polynomial equation

$$accuracy = A_0 + A_1 N_S + A_2 N_S, \tag{3.4.6}$$

where

A_0 $\ldots -5.2 \pm 0.2$ Hz,
A_1 $\ldots 0.011 \pm 0.004$ Hz,
A_2 $\ldots 83 * 10^{-5} \pm 1.5 * 10^{-5}$ Hz.

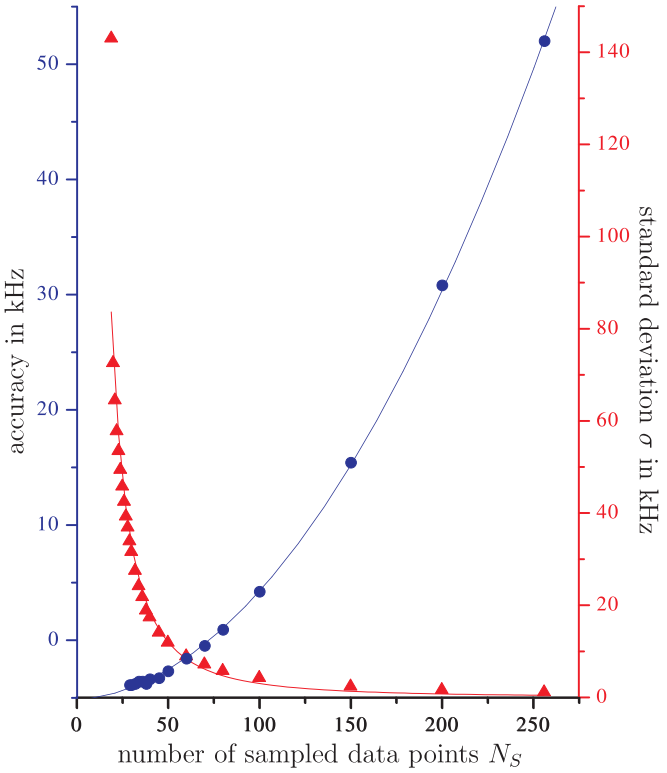

Figure 19: Effect of the number of sampled data points on the standard deviation and accuracy of frequency determination in a 256-point FFT. For each of the ten-thousand FFTs used to determine each plotted data point, a 50.000000 MHz 100 mV peak-to-peak sine wave was sampled N_S times at 250 MHz. N_S was varied between 19 and 256, and the remaining $256-N_S$ input points were set to zero. A Blackman-Harris window was applied to the N_S sampled points before the zero-padding. The frequency peak was located by quadratic interpolation.

This increase in the bias with the number of data points is caused by the window function, as verified by repeating measurements with and without windowing for various FFT sizes and frequencies. If the frequency is chosen to correspond with a basis vector such that windowing is redundant (and therefore not applied), 5000 64-point FFTs calculated on a 46.87500 MHz true frequency produce $\mu = 46.875008$ MHz and $\sigma = 0.000016$ MHz. This is a clear indication that the systems are functioning correctly. Since the maximum measured shift never exceeded 50 kHz, and is thus within the 100 kHz tolerance, it should not prove to be a problem.

Use of Software to Quantify Power Laser Offset Jitter The software was used to quantify the time until stability in the offset frequency of the power oscillator is reached, as well as the attainable stability. From Fig. 20, it can been seen that the

stabilizing circuit requires approximately 2 min to settle.

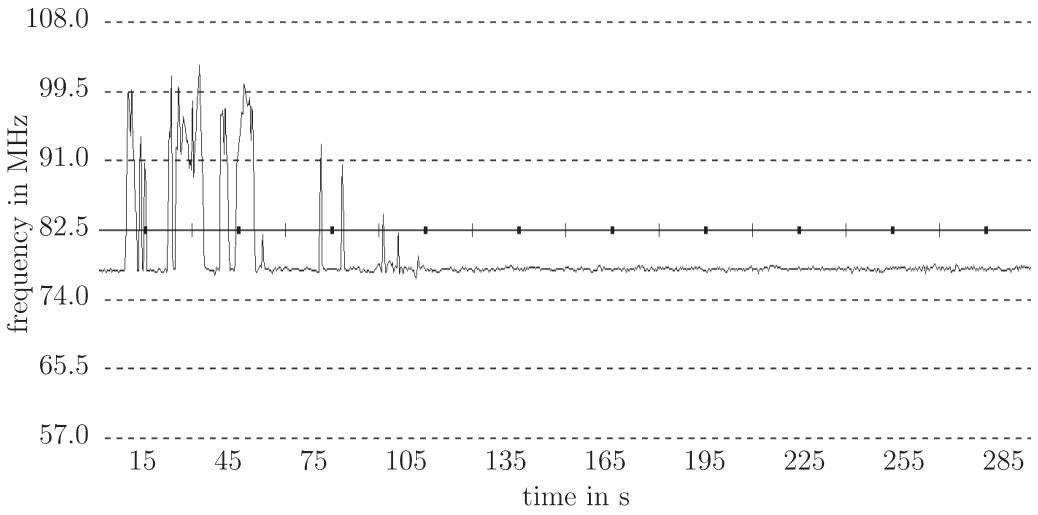

Figure 20: Stabilization of the power laser occurs after 120 s.

Once settled, the offset frequency mean and standard deviation for this particular measurement was 81.6±2.4 MHz, as shown in Fig. 21. Rejecting bad shots would therefore result in the rejection of 97% of all shots and is thus unacceptable. Without improved stabilization, the offset alignment algorithm must be applied before shot-to-shot spectrum summation is possible.

The sampled signal can be seen in Fig. 22. After approximately 500 samples, the pulse has diminished indicating a pulse length on the order of 1 µs. The sine wave oscillations in the pulse amplitude result from heterodyning.

Before sending the sampled signal to the FFT algorithm, it is multiplied by a Blackman-Harris window. The result of the signal multiplied by the window can be seen in Fig. 23. The calculated peak frequencies for a measurement of $5 \cdot 10^5$ shots is shown in Fig. 24. These calculated values were used for the statistical analysis of Fig. 21.

The figures 20 to 24 present the data as it is acquired and processed by the program. They indicate the stages from acquisition to analysis. The user can follow these stages in real time and witness the effects achieved by modifying measurement parameters. The program has proven its functionality and usefulness in the wind lidar system. With the software it is possible to quantify the jitter in the offset frequency of the power oscillator in addition to processing the atmospheric return signal.

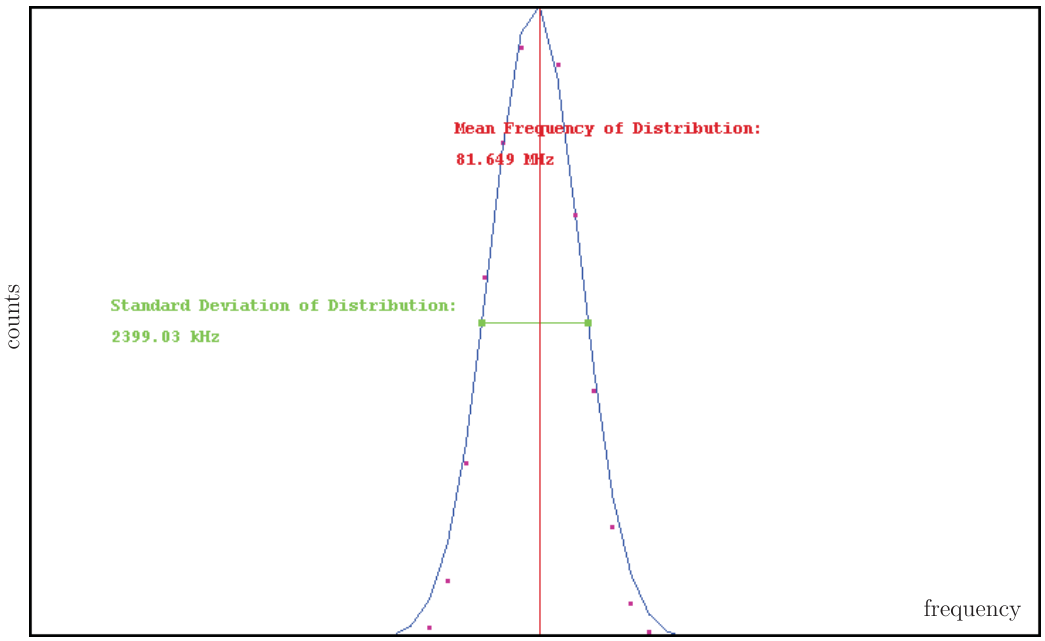

Figure 21: Statistics on 500000 offset frequency determinations. A normal distribution with the same mean and standard deviation is shown for comparison.

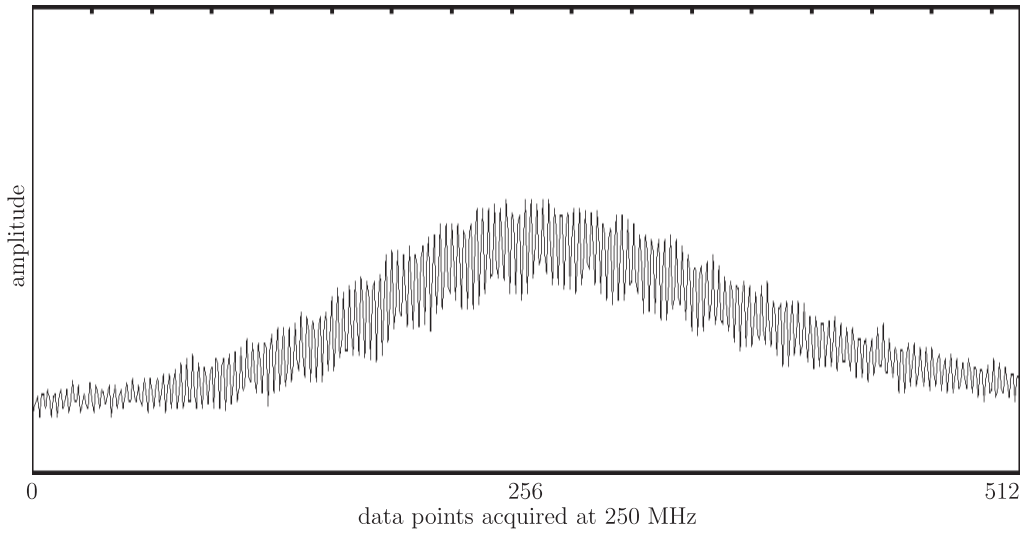

Figure 22: The output of the power laser mixed with the local oscillator. The Gaussian form of the power laser pulse can be seen, as well as the sine wave resulting from the heterodyne measurement.

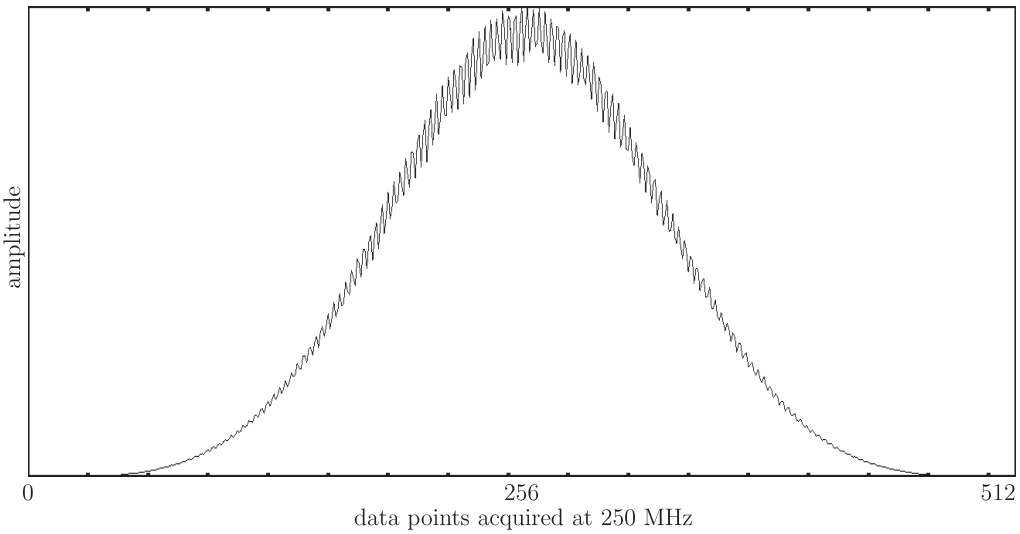

Figure 23: The raw signal is multiplied by a window before being analyzed for its frequency components.

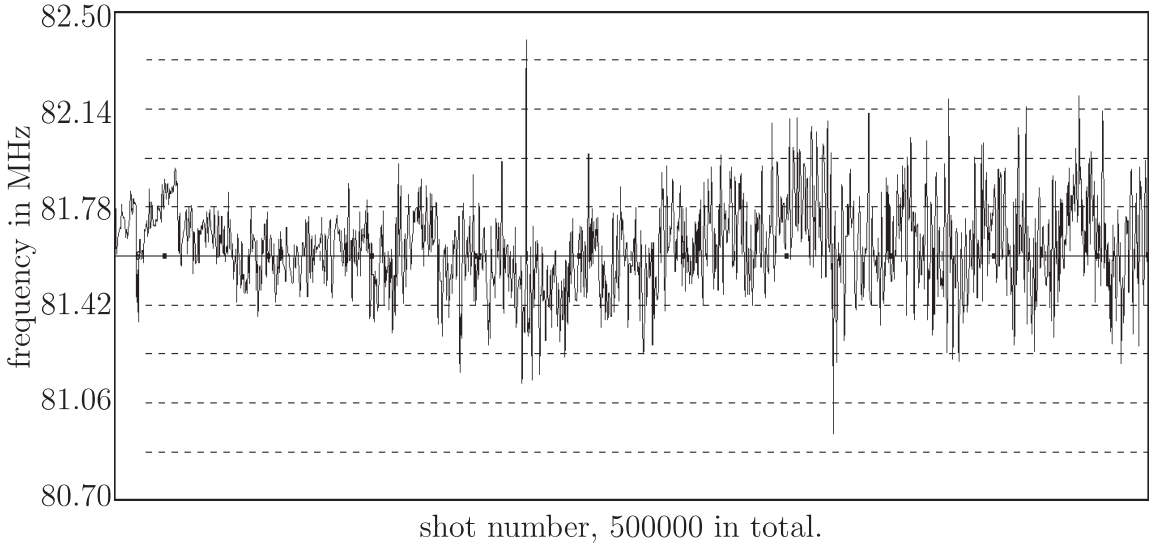

Figure 24: The offset frequencies of the power laser are plotted here for 500000 shots. Frequency in MHz can be seen on the vertical axis.

4 Conclusions and Recommendation for Further Work

The purpose of this work was to develop the data acquisition and analysis systems for a portable Raman lidar and a wind lidar. In the introduction, the necessity of research into atmospheric particle properties and particle transport mechanisms was clarified. In chapter 2, the applied lidar methods and the mathematical background for the algorithms used in the data analysis software were introduced. The data acquisition and analysis system for the mini Raman lidar system was successfully developed. Using measurements made with the system, the PBL height could be seen as well as cirrus clouds in the free troposphere. Particle backscatter and extinction coefficient profiles were determined. The extinction values calculated have been shown to be in good agreement with measurements made with other systems [10]. The software has proven to be reliable. Some work related to automation is still open, specifically the ability to pre-program a measurement schedule, regular delivery of calculated profiles and time development via email or web-publishing of results on the homepage of the IFT, and independent filter value selection based on peak count rate.

The second part of this thesis, namely the development of the data acquisition and analysis system for a wind lidar, was successfully accomplished. First, the frequency tolerance (or precision) and accuracy required in the Doppler frequency measurements were specified along with the minimal desired height resolution and shot repetition rate. The data analysis system and algorithms were optimized to achieve the specifications. The specified frequency accuracy is attained by multiplying the sampled data points by a 4-point Blackman-Harris apodization function and padding zeros before calculating the discrete Fourier transform via the FFT algorithm. The thus calculated power spectra are summed over shots, after correcting for possible drift in the offset of the power oscillator output frequency relative to the local oscillator. Particular attention was paid to the speed with which the analysis software calculates height profiles, since the increased shot repetition rate resulting from optimized code allows improved statistical calculations. Through the introduction of multiple execution threads to the source code and optimization of time-critical loops, the execution time for the processing of an 8-km range at 15-m resolution was reduced by a factor of 5 such that the desired shot repetition rate of 500 Hz was reached. Testing of the complete system can only be performed when the power oscillator is complete. The software source code was well commented and designed in a modular way to ease future modifications such as the addition of a scanner for three dimensional measurements.

In conclusion, using the lidar expertise of the IFT's staff, it was possible to construct a small and portable largely automated dual-channel Raman lidar. This Raman lidar can measure the backscatter properties of atmospheric particles and at night additionally the particle extinction coefficients. It is used to collect significant volumes of data and is a useful tool in planning measurements with the larger systems of the IFT, which are

expensive to operate. The powerful software developed for the wind lidar is capable of precise and accurate frequency determination and is useful for quantifying the jitter and drift in the output of the power oscillator in addition to performing the tasks of calculating spatially and time resolved wind speed profiles at a high repetition rate. The goals were achieved, and soon the IFT will possess a functioning advanced wind lidar system, thus enabling research into the turbulent particle exchange processes between the planetary boundary layer and the free troposphere.

Appendices

Appendix A Program GUIs

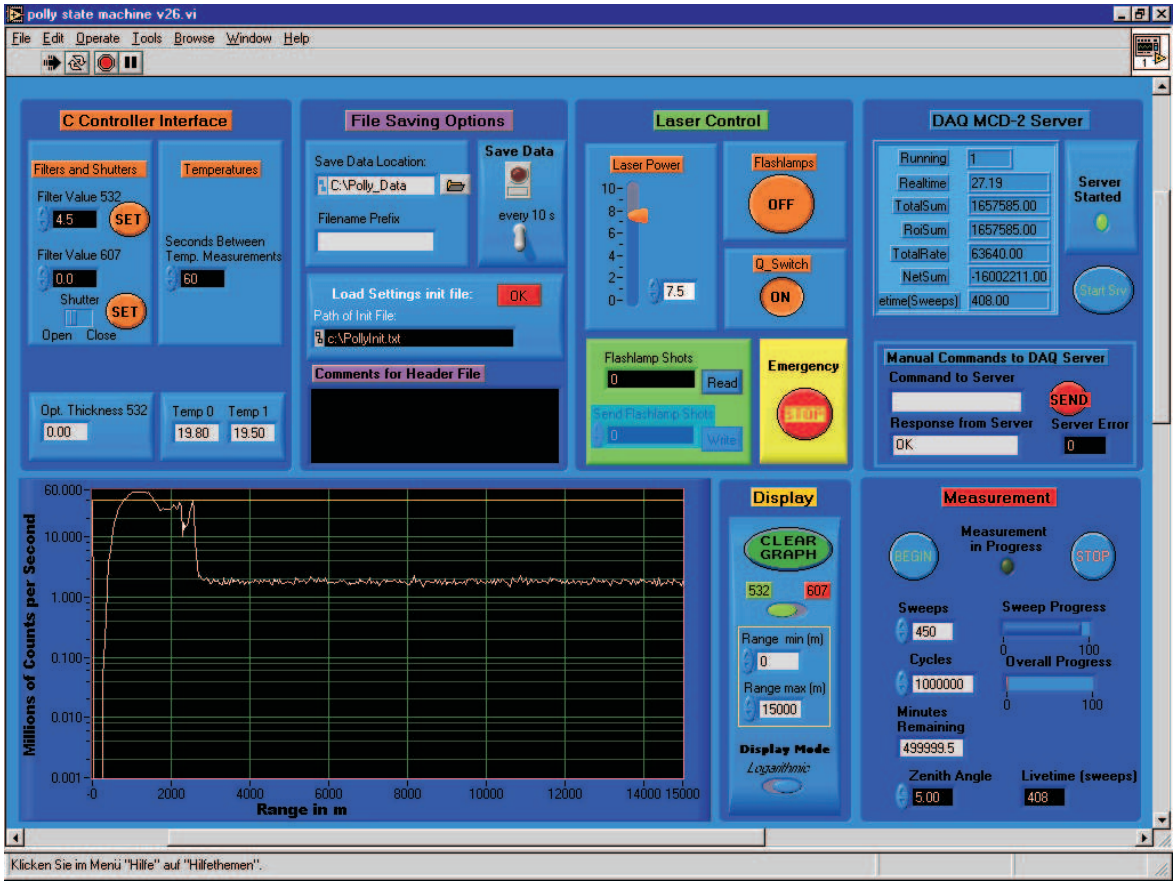

Figure 25: Graphical user interface for the control of the portable Raman lidar system. Related controls are organized into visually separate boxes. The graph shows the output of the current measurement cycle. In the plot visible on-screen, a thick cloud at 2500 m was preventing measurements at higher altitudes.

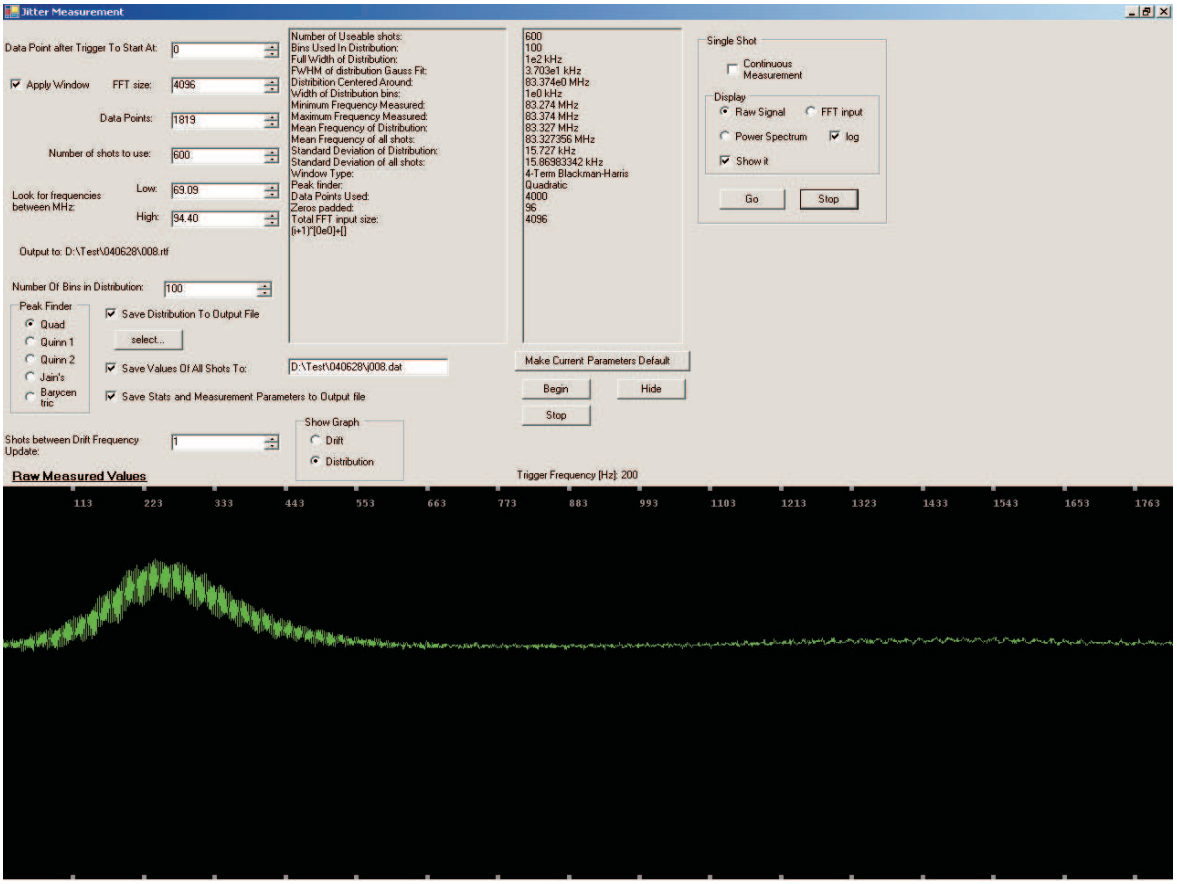

Figure 26: Graphical user interface for determination of the jitter in the offset of the wind lidar's power laser. It incorporates all required functions. These include file saving, FFT parameters, peak finding algorithm selection, and statistical options such as the number of distribution bins.

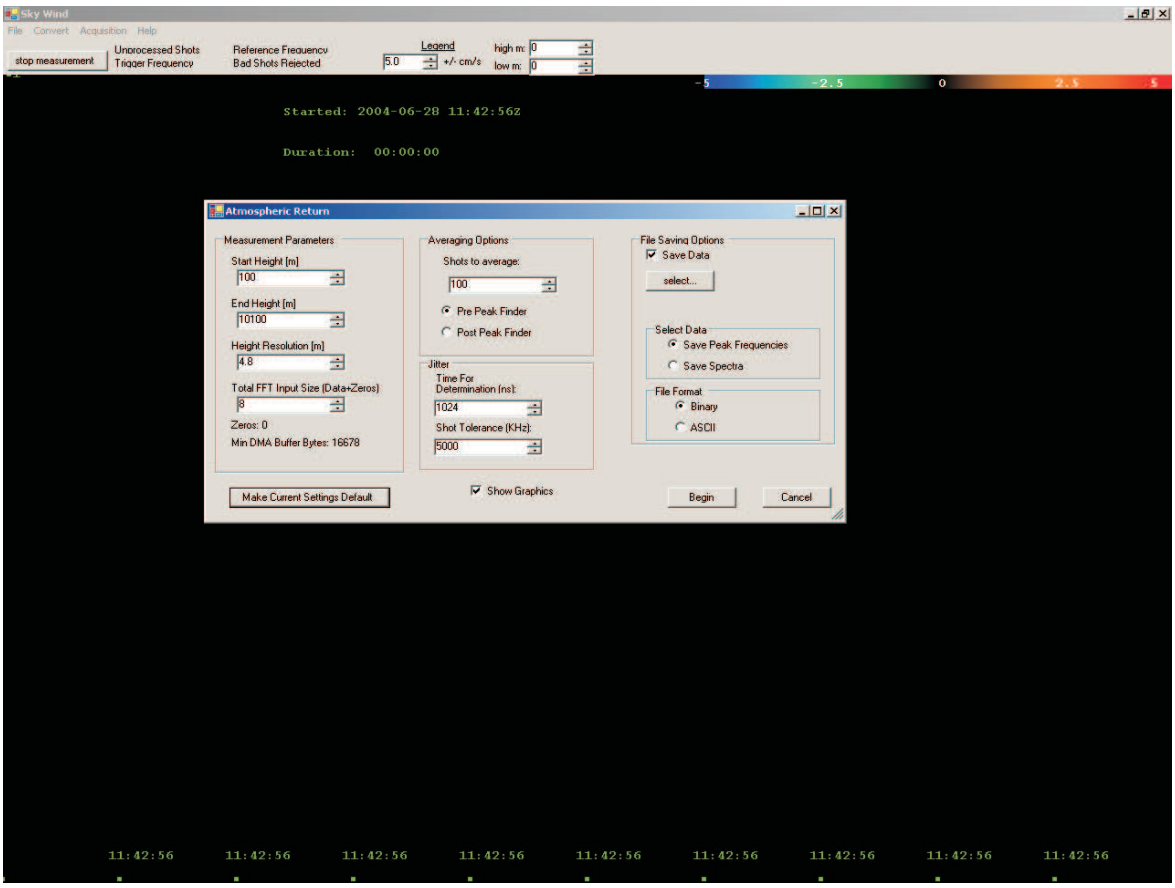

Figure 27: This displays the window for the atmospheric return signal of the wind lidar. Parameters include distance range, height resolution, offset frequency tolerance, as well as various file saving options. The range of velocities can be adjusted as the measurement progresses. The color bar is shown in the upper right hand corner.

References

[1] J. T. Houghton, Y. Ding, D. J. Griggs, M. Noguer, and P. J. van der Linden. IPCC - Intergovernmental Panel on Climate Change, Climate Change 2001: The Scientific Basis, Summary for Policymakers.
URL http://www.grida.no/climate/ipcc_tar/
wg1/pdf/WG1_TAR-FRONT.PDF, September 2003.

[2] D. Althausen, D. Mller, A. Ansmann, U. Wandinger, H. Hube, E. Clauder, and S. Zrner. Scanning six-wavelength eleven-channel aerosol lidar. *J. Atmos. and Oceanic Technol.*, 17:1469–1482, 2000.

[3] I. Mattis, A. Ansmann, D. Althausen, V. Jaenisch, U. Wandinger, D. Mller, Y. F. Arshinov, S. M. Bobrovnikov, and I. B. Serikov. Relative-humidity profiling in the troposphere with a Raman lidar. *Appl. Opt.*, 41:6451–6462, 2002.

[4] A. Ansmann, U. Wandinger, A. Wiedensohler, and U. Leiterer. Lindenberg Aerosol Characterization Experiment 1998 (LACE 98): Overview. *J. Geophys. Res.*, 107:8129–8140, 2002.

[5] P. B. Russell and J. Heintzenberg. An overview of the ACE 2 clear sky column closure experiment (CLEARCOLUMN). *Tellus, Ser. B*, 52:463–483, 2000.

[6] V. Ramanathan et al. Indian Ocean Experiment: An integrated analysis of the climate forcing and effects of the great Indo-Asian haze. *J. Geophys. Res.*, 106:28371–28398, 2001.

[7] J. Bsenberg et al. EARLINET: A European Aerosol Research Lidar Network. *Laser remote sensing of the atmosphere. Selected papers of the 20th International Laser Radar Conference*, pages 155–158, 2001.

[8] C. L. Korb, B. Gentry, and C. Weng. The edge technique – theory and application to the lidar measurement of atmospheric winds. *Appl. Opt.*, 31:4202–4213, 1992.

[9] C. J. Grund, R. M. Banta, J. L. George, J. N. Howell, M. J. Post, R. A. Richter, and A. M. Weickmann. High-resolution Doppler lidar for boundary layer and cloud research. *J. Atmos. and Oceanic Technol.*, 18:376–393, 2001.

[10] R. Engelmann. Entwicklung eines Mini-Ramanlidar und Aufbau einer kombinierten Sende- und Empfangseinheit fr ein Doppler-Wind-Lidar. Master's thesis, Universitt Leipzig, September 2003.

[11] R. M. Measures. *Laser Remote Sensing - Fundamentals and Applications*. Krieger Publishing Company, 1992.

[12] C. McLinden. Mie scattering.
URL http://www.ess.uci.edu/ cmclinden/link/xx/node19.html, July 1999.

[13] B. W. Carroll and D. A. Ostlie. *An Introduction to Modern Astrophysics*. Addison-Wesley Publishing Company, Inc., 1996.

[14] S. W. Smith. *The Scientist and Engineer's Guide to Digital Signal Processing*. California Technical Publishing, 2nd edition, 1999.

[15] F. J. Harris. On the use of windows for harmonic analysis with the discrete Fourier transform. *Proceedings of the IEEE*, 66 No.1:51–84, January 1978.

[16] J. W. Cooley and J. W. Tukey. An algorithm for the machine calculation of complex Fourier series. *Mathematics Computation*, 19:297–301, 1965.

[17] D. Michel. *Grundlagen und Methoden der kernmagnetischen Resonanz*. Akademie-Verlag Berlin, 1981.

[18] J. Templeman and A. Olsen. *Microsoft Visual C++ .NET Step by Step*. Microsoft Press, 2003. Version 2003.

[19] A. McGrath. *An Erbium:Glass Coherent Laser Radar for Remote Sensing of Wind Velocity*. PhD thesis, The Flinders University of South Australia, School of Earth Sciences, Faculty of Science and Engineering, August 1998.

[20] A. Hunt and D. Thomas. *The Pragmatic Programmer*. Addison Wesley, Boston, 2000.

[21] D. Lafreniere. State machine design in C++.
URL http://www.cuj.com/documents/s=8039/cuj0005lafrenie/, May 2000. lafreni@pacbell.net.

[22] D. Thomas and A. Hunt. State machines. *IEEE Software*, pages 10–12, November/December 2002.

[23] J. P. Boyd. Scientific visualization and information architecture. University of Michigan,
URL http://www-personal.engin.umich.edu/ jpboyd/, 2000.

[24] J. Tyson. How PCI works.
URL http://computer.howstuffworks.com/pci.htm/printable.

[25] J. Tyson. How IDE controllers works.
URL http://computer.howstuffworks.com/ide.htm/printable.

[26] J. Tyson. How SCSI works.
URL http://computer.howstuffworks.com/scsi.htm/printable.

[27] Advanced Computer & Network Corporation. Raid.
URL http://www.acnc.com/raid.html, 2004.

[28] S. Weinhold. Raid systems, personal communication, 2003.

[29] A. Ansmann, M. Riebesell, and C. Weitkamp. Measurements of atmospheric aerosol extinction profiles with a Raman lidar. *Opt. Lett.*, 15:746–748, 1990.

[30] A. Ansmann, M. Riebesell, U. Wandinger, C. Weitkamp, and W. Michaelis. Independent measurement of extinction and backscatter profiles in cirrus clouds by using a combined Raman elastic-backscatter lidar. *Appl. Opt.*, 29:3266–3272, 1992.

[31] U. Wandinger. *Raman Lidar, In: Lidar – Range-resolved optical remote sensing of the atmosphere*. Springer, in press, 2004.

[32] P. Horowitz and W. Hill. *The Art of Electronics*. Cambridge University Press, second edition, 1995.

[33] D. N. Whiteman. Examination of the traditional Raman lidar technique. – I. Evaluating the temperature-dependent lidar equations. *Appl. Opt.*, 42:2571–2592, 2003.

[34] C. Cornwall, A. Horiuchi, and C. Lehman. General solar position calculations. URL http://www.srrb.noaa.gov/highlights/sunrise/solareqns.PDF, December 2003.

[35] R. Brodbeck. Die Zeitgleichung - Eine einfache Formel zu Sonnenaufgang und Untergang.
URL http://lexikon.astronomie.info/zeitgleichung/, April 2004.

[36] R. Trebino. Optics lectures. Technical report, Georgia Institute of Technology, URL http://www.physics.gatech.edu/gcuo/UltrafastOptics/, January 2004.

[37] E. Zeromskis, U. Wandinger, D. Althausen, R. Engelmann, P. Rhone, and R. Foster. Coherent Doppler lidar for wind profiling in the lower atmosphere. *ISTP 2003, Sixth International Symposium on Tropospheric Profiling: Needs and Technologies*, pages 71–73, 2003.

List of Figures

List of Tables

Acknowledgments

I'd like say a few words of thanks to a number of people without whom this thesis never would have been possible.

My adviser Dietrich Althausen for providing me with the opportunity to do my diploma work at the IFT. His expertise were a continuous source of knowledge, and his constant desire for progress reports kept me on-track. His friendship is invaluable to me, and I thank him for the patience and understanding he showed when things weren't going the way he hoped.

Ulla Wandinger for her careful last-minute proofreading and corrections of both the scientific content and language in this thesis. I got the distinct impression that her English is better than mine. Her requests for new chapters to correct cost me many hours I would have rather spent in bed, but helped me complete everything on time. I would also like to thank her for teaching me how to interpret the data that Polly provides.

Egidijus Žeromskis for his willingness to help, quick delivery of books, papers, and links which he thought would be useful, and for proofreading. Also for putting up with the pigsty I've transformed his lab into, and not reporting me for eating spaghetti here. I'm happy that my program is useful to him, and wish him all the best in the completion of his PhD thesis.

My experimental physics professor Dieter Michel for inspiring me to pursue my studies, and making me feel welcome in a new country and renowned university before I knew which way was up. He was always willing to put in the extra effort to help his students achieve their best. I would also like to thank him for taking time out of his busy schedule to help me understand the mathematical problems facing me in the development of the wind lidar software. I'm glad that I had the opportunity to repay him in a small way by proofreading an English paper for him. I suspect, however, that his English is also better than mine, and thus my help was of little value!

Robin Foster for constantly keeping me from getting any work done with his interesting stories and news reports... ...and video games. Also for his considerable knowledge regarding digitization and his expertise in Matlab and LabView.

My colleague and friend Ronny Engelmann for his seeming ability to do everything, and do it well. Any project begun with him is as sure to be enjoyable as it is to reach a successful conclusion. His interest in so many things is catchy, and the reason that I took up openGL and now use so much open source software, not to mention that I probably never would have written this thesis in LaTeX had he not introduced me to it. Many things contained in this work were learned from him one way or another.

Albert Ansmann for always being enthusiastic, interesting, and friendly. I picked up some basics of meteorology from him. I once rode my motorcycle through the rain when he predicted sunshine though. Although he probably doesn't remember, I sure do.

Anja Immisch for putting up with me. I wouldn't have bothered had I been her. But I thank her for it!

My dad for supporting me financially, Gregory for showing me what a family is supposed to be like, Christopher for constantly reminding me to think about things I'd rather ignore, like my future, and Emilie for being the mother to Huckelberry and Luc, the two cutest little boys I've ever seen.

My skateboard. OK it isn't a person, but without it I never would have met some of my best friends like Jean-Paul Adolph, Tristan Schultze, Patrick Kotte, Tobias Golde, and Kai-Uwe Kauders. These people not only supported me in my work and put up with my constant work-related tardiness, but provided me with necessary distractions. Once again, Tristan and Ralf, thanks for the Polly logo.

Also, I'd like to thank Alex Hertsch. A fellow student and friend to me over the years, this thesis concludes my side of a promise we made to each other in first semester that we wouldn't give up until we'd finished!

Statement of Authenticity

This diploma thesis contains no material which has been accepted for the award of any other degree or diploma in any tertiary institution, and to the best of the author's knowledge and belief, this diploma thesis contains no copy or paraphrase of material previously published or written by another person, except where due reference is made in the text of this thesis.

Upon positive assessment, copies of this diploma thesis will be provided for the libraries of the Faculty of Physics and Earth Sciences, University of Leipzig, and the Institute for Tropospheric Research, Leipzig.

---

date signature