DIGITAL FILTERS FOR NUMERICAL WEATHER PREDICTION

Peter Lynch
Meteorological Service
Dublin, Ireland

HIRLAM Technical Report No. 10

January, 1993

ABSTRACT

Digital filters, originally developed in the context of Discrete-time Signal Processing (DSP), have recently been used for the analysis and solution of problems in geophysics. In the field of numerical weather prediction these filters have been applied to the problems of initialization and objective analysis, and they have considerable potential for other applications in NWP.

The theory of digital filters is presented in this report, with the interests of atmospheric modellers in mind. Both non-recursive and recursive filters are discussed, and applications of both of these to initialization are described. The purpose of the report is to provide a gentle introduction to digital filtering, with emphasis on the aspects thought to be most important for NWP applications. The report should give the reader easier access to the more recondite treatments of filtering available in the DSP literature.

Note: Figures not included. Please contact author (Peter.Lynch@met.ie) for a complete paper copy of this report.

CONTENTS

1. INTRODUCTION

Digital Signal Processing (DSP) plays a central rôle in modern telecommunications, music recording and reproduction, television, speech and image processing, medical diagnosis and remote sensing. Digital filters (DFs) are now ubiquitous, being found in wrist-watches, CD players, touch-tone telephones, speech and music synthesizers, CAT scanners, doppler radars and virtually all modern measuring instruments. More recently, these filters have been used for processing geophysical data, and they have considerable potential for application in the area of numerical weather prediction.

In the field of NWP, digital filters have been applied to the problems of initialization (Lynch and Huang, 1992; Huang and Lynch, 1992) and objective analysis (Lorenc, 1992). They may also provide a means of systematically constructing frequency-selective integration schemes (Lynch, 1991b). They provide a promising means of imposing constraints in four-dimensional data assimilation, and have potential for application in a number of other areas.

Simple analog filters are considered in §2, and the key ideas of filtering are introduced. The basic theory of Digital Filters (DFs) is reviewed in §3. Design methods for non-recursive filters are presented in §4, and an application to initialization is described briefly. One great advantage of a non-recursive filter is the possibility to make its phase linear in frequency. This implies distortion-free output, delayed in time by half the span. But this can also be a drawback: the non-recursive filter cannot provide output valid at an extremity of the input span. This, coupled with the irreversibility of a diabatic integration, makes application to diabatic initialization problematical. A recursive filter may be devised which circumvents this problem. The basic design methods for recursive filters are presented in §5, and an application to initialization of a barotropic model is described in §6. Some concluding remarks are found in §7.

An extensive literature on digital filters exists in Digital Signal Processing publications. A selection of references to the DSP books which have been found to be most helpful can be found at the end of this report.

2. BACKGROUND: ANALOG FILTERS

2.1 The Fundamental Idea of Filtering

The process of filtering involves the selection of those components of an assemblage having some particular property, and the removal or elimination of those components which lack it. A filter is any device or contrivance designed to carry out such a selection. We are primarily concerned with filters as used in signal processing. The selection principle for these is generally based on the frequency of the signal components. There are a number of ideal types (lowpass, highpass, bandpass and bandstop) corresponding to the range of frequencies which pass through the filter and those which are rejected. In many cases the input consists of a low-frequency (LF) signal contaminated by high-frequency (HF) noise, and the information in the signal can be isolated by using a lowpass filter which rejects the noise. Such a situation is typical for the applications to meteorology discussed below.

Filter theory originated from the need to design electronic circuits with precise frequency-selective characteristics, for radio and telecommunications. These analog filters were constructed from capacitors and inductors, and acted on continuous time signals. More recently, discrete time signal processing has assumed prominance, and the technique and theory of *digital filtering* has evolved. Analog filters are usually realized as electronic circuits; digital filters may also be implemented in hardware using integrated curcuits, but are more commonly realized in software: the input is processed by a program designed to perform the required selection and compute the output.

2.2 A Simple Example

The basic ideas and (relevant) jargon of filtering can be conveniently introduced by means of a simple example. Let us consider the linear second order *o.d.e.* governing a forced, damped harmonic oscillator:

$$\ddot{y} + 2\sigma_0\dot{y} + (\sigma_0^2 + \omega_0^2)y = x(t). \tag{2.1}$$

The coefficient $\sigma_0 > 0$ determines the damping, and ω_0 is the natural frequency of the system. We call $x(t)$ the forcing, stimulus or excitation, or simply the *input* , and $y(t)$ the response or *output* . The solution of the equation tells us the output $y(t)$ for a given input $x(t)$. In the simple but important case of a sinusoidal forcing $x(t) = \exp(i\omega t)$, a particular (forced) solution is given by

$$y_F(t) = H(i\omega) \cdot \exp(i\omega t)$$

where the *response* function, *transfer* function or *system* function, $H(i\omega)$, is

$$H(i\omega) = \frac{1}{(\sigma_0^2 + \omega_0^2 - \omega^2) + 2i\sigma_0\omega}.$$

This function depends on the system parameters σ_0 and ω_0 and on the forcing frequency ω. Its complex character indicates that both the magnitude and phase of the output are changed relative to the input. For an undamped oscillator ($\sigma_0 = 0$) we have the possibility of unbounded response for $\omega \to \omega_0$. With weak positive damping ($\sigma_0 < \omega_0$) there is resonance: the amplitude of the output is maximum for forcing near the natural frequency, $\omega \approx \omega_0$. With stronger damping ($\sigma_0 \geq \omega_0$) the magnitude of $H(\omega)$ is maximum at $\omega = 0$. In addition to the particular solution forced by $x(t)$, the homogeneous form of (2.1) has a characteristic solution

$$y_T(t) = e^{-\sigma_0 t}\big(C_+ \exp(i\omega_0 t) + C_- \exp(-i\omega_0 t)\big).$$

The coefficients $C_\pm$ are determined by the initial conditions. For finite damping ($\sigma_0 \neq 0$) $y_T(t)$ dies out exponentially with time and is called the *transient* solution. The full response is $y(t) = y_F(t) + y_T(t)$, comprising forced plus transient components.

Equation (2.1) models a number of basic mechanical and electrical systems. For example, if $v(t)$ is the input voltage applied to a series RLC circuit, and q is the charge on the capacitor, Kirchhoff's voltage law implies

$$L\ddot{q} + R\dot{q} + \left(\frac{1}{C}\right) q = v(t).$$

This equation is isomorphic to (2.1) with the identifications $x \to v/L$, $y \to q$ and

$$\sigma_0 = \left(\frac{R}{2L}\right), \qquad \omega_0^2 = \left[\frac{1}{LC} - \left(\frac{R}{2L}\right)^2\right].$$

For $\omega_0^2 > 0$ the transient solution is oscillatory or under-damped; for $\omega_0^2 < 0$ it is exponential or over-damped; the case $\omega_0^2 = 0$ is called *critical damping* and is the value for which the transient solution decays fastest.

Further analysis of (2.1) is facilitated by introducing the Laplace transform (see, *e.g.,* Doetsch, 1971). The transform of the equation is

$$\big(s^2 + 2\sigma_0 s + (\sigma_0^2 + \omega_0^2)\big)\hat{y}(s) = \hat{x}(s) + (s + 2\sigma_0)y(0) + \dot{y}(0) \tag{2.2}$$

(hats denote transforms). The solution of this algebraic equation is

$$\hat{y}(s) = H(s)\big[\hat{x}(s) + (s + 2\sigma_0)y(0) + \dot{y}(0)\big]. \tag{2.3}$$

where the transfer function $H(s)$ is given by

$$H(s) = \frac{1}{s^2 + 2\sigma_0 s + (\sigma_0^2 + \omega_0^2)}. \tag{2.4}$$

For $\omega_0^2 > 0$ this function has two complex conjugate poles in the s-plane:

$$H(s) = \frac{1}{(s - s_p)(s - \bar{s}_p)} \tag{2.5}$$

where $s_p = -\sigma_0 + i\omega_0$. The positions of the poles are determined by the system parameters σ_0 and ω_0, and the character of the transient solution depends on them. But the *forced* response is also strongly influenced by the location of these poles. Using the convolution theorem for the Laplace transform, the solution $y(t)$ is obtained by inverting (2.3):

$$y(t) = e^{-\sigma_0 t}\big(C_+ \exp(i\omega_0 t) + C_- \exp(-i\omega_0 t)\big) + \int_0^t h(\tau)x(t-\tau)\, d\tau \tag{2.6}$$

where $h(t)$ is the inverse transform of $H(s)$ and the coefficients $C_\pm$ are determined from the initial conditions $y(0)$ and $\dot{y}(0)$. We do not need explicit expressions for $C_\pm$, noting only that the transient component decays with an e-folding time $1/\sigma_0$, and focus on the forced response represented by the integral term.

For an impulse input, $x(t) = \delta(t)$, the integral may be evaluated by inspection and is equal to $h(t)$. Thus, the impulse response is the inverse transform of the transfer function:

$$h(t) = \mathcal{L}^{-1}\{H(s)\} = \left(\frac{\exp(s_p t) - \exp(\bar{s}_p t)}{s_p - \bar{s}_p}\right).$$

For a sinusoidal input, $x(t) = \exp(i\omega t)$, the evaluation of the integral is more awkward, but elementary, and the response is

$$\frac{(\bar{s}_p - i\omega)\exp(s_p t) - (s_p - i\omega)\exp(\bar{s}_p t) + (s_p - \bar{s}_p)\exp(i\omega t)}{(s_p - \bar{s}_p)(s_p - i\omega)(\bar{s}_p - i\omega)}.$$

Ignoring the evanescent components, the steady-state response is

$$\left[\frac{1}{(s_p - i\omega)(\bar{s}_p - i\omega)}\right] e^{i\omega t} = H(i\omega)\cdot x(t).$$

Thus, the pure sinusoidal input gives rise to an output of the same frequency, whose amplitude and phase are determined by $H(i\omega)$, the transfer function evaluated on the imaginary axis of the s-plane.

2.3 Analysis of the Transfer Function $H(s) = \dfrac{1}{(s - s_p)(s - \bar{s}_p)}$.

The response function (2.5) has the character of a low-pass filter. How can we see this? Its value at the low-frequency limit $\omega = 0$, the so-called dc gain, is

$$H(0) = \frac{1}{\sigma_0^2 + \omega_0^2}. \tag{2.7}$$

For large $|s|$, $H(s)$ behaves like s^{-2}, so its amplitude tends to zero as $\omega \to \pm\infty$, and high frequency inputs are effectively damped.

The behaviour of the transfer function on the imaginary axis is of particular interest. We can write $H(i\omega)$ in terms of its amplitude and phase:

$$H(i\omega) = M(\omega)\cdot \exp\bigl(i\varphi(\omega)\bigr)$$

and expressions for $M(\omega)$ and $\varphi(\omega)$ follow from (2.5):

$$M(\omega) = \sqrt{\frac{1}{[\sigma_0^2 + (\omega_0 - \omega)^2][\sigma_0^2 + (\omega_0 + \omega)^2]}}\,, \tag{2.8}$$

$$\varphi(\omega) = -\tan^{-1}\left(\frac{\omega - \omega_0}{\sigma_0}\right) - \tan^{-1}\left(\frac{\omega + \omega_0}{\sigma_0}\right). \tag{2.9}$$

We will examine three properties of the transfer function $H(s)$ which are crucial for understanding the suitability of such filters for applications: (a) Cut-off frequency, (b) Phase-error and Delay and (c) Transient response and Start-up time.

(a) Cut-off frequency: Prototype Filters

Since $H(s)$ tends to zero for large $|s|$, there exists a contour in the s-plane outside which its magnitude-squared is less than half the value $|H(0)|^2$. The point $i\omega_c$ where this contour intersects the imaginary axis is called the cutoff frequency:

$$|H(i\omega_c)|^2 = \tfrac{1}{2}|H(0)|^2. \tag{2.10}$$

It is the half-power frequency, or 3-deciBel point, since the attenuation at ω_c is

$$-20\log\left|\frac{H(i\omega_c)}{H(0)}\right| = -10\log\left|\frac{H(i\omega_c)}{H(0)}\right|^2 = 10\log 2 \approx 3\text{ dB}.$$

Using (2.7) and (2.8) in (2.10) yields a relationship between the parameters σ_0 and ω_0:

$$2(\sigma_0^2+\omega_0^2)^2 = (\sigma_0^2+\omega_0^2-\omega_c^2)^2 + 4\sigma_0^2\omega_c^2 \tag{2.11}$$

It is convenient to consider *prototype* filters, for which $\omega_c = 1$; the general case is recovered by the the transformation $s \to s/\omega_c$. Then (2.11) becomes

$$(\sigma_0^2+\omega_0^2)^2 = 2(\sigma_0^2-\omega_0^2)+1. \tag{2.12}$$

This is the equation of a curve in the s-plane. It is a particular case of a family of curves called Cassinian ovals (see Fig. 2.1). The general equation for these curves, in more familiar cartesian notation, is

$$(x^2+y^2)^2 = 2(x^2-y^2)+(a^4-1).$$

The value $a^2 = \sqrt{2}$ corresponds to (2.12).

Since all prototype filters (of the type under consideration) have parametes (σ_0, ω_0) lying on the curve (2.12), there is in effect one free parameter: we are free to impose one further condition on the filter. Three special choices will be considered:

(i) The Butterworth (BW) filter has an amplitude response which is maximally flat in the pass-band

(ii) The Bessel or Flat-delay (FD) filter has a phase response which is as near linear as possible in the pass-band

(iii) The Quick-start (QS) filter has a transient response which decays as rapidly as possible.

First consider the magnitude response. The first derivative of $M(\omega)$ with respect to ω vanishes at $\omega = 0$ for all choices (σ_0, ω_0). The imposition of the condition

$$\frac{d^2M(\omega)}{d\omega^2} = 0 \quad \text{at} \quad \omega = 0 \tag{2.13}$$

implies that the amplitude response is as flat as possible at the low-frequency limit. This is the definition of the Butterworth filter. It leads immediately to the requirement that $\omega_0 = \pm\sigma_0$, which, together with (2.12) implies

$$\sigma_0 = \frac{1}{\sqrt{2}}, \qquad \omega_0 = \pm\frac{1}{\sqrt{2}}. \tag{2.14}$$

Thus, the representative point for the Butterworth filter lies where the curve (2.12) intersects the unit circle (Fig. 2.2).

(b) Phase-error and Group Delay

The response to a sinusoidal input $x(t) = \exp(i\omega t)$ is

$$y(t) = H(i\omega)x(t) = M(\omega)\exp\big[i\big(\omega t + \varphi(\omega)\big)\big].$$

The case of linear phase $\varphi = -t_0\omega$ is especially important, since it corresponds to delay in time for all components of the input irrespective of frequency:

$$y(t) = M(\omega)\exp\big[i\omega(t - t_0)\big].$$

The time delay is given in this case by $t_0 = -d\varphi/d\omega$. Generalising this idea, we define the group delay, δ, as the derivative of the phase response (2.9) with respect to frequency:

$$\delta = -\frac{d\varphi}{d\omega} = \left\{\frac{\sigma_0}{\sigma_0^2 + (\omega - \omega_0)^2} + \frac{\sigma_0}{\sigma_0^2 + (\omega + \omega_0)^2}\right\}.$$

We will refer to the value of δ at $\omega = 0$ simply as the *delay*:

$$\delta_0 = \frac{2\sigma_0}{\sigma_0^2 + \omega_0^2}. \tag{2.15}$$

A straightforward calculation shows that $d\delta/d\omega = 0$ is automatically satisfied at $\omega = 0$. The condition

$$\frac{d^2\delta}{d\omega^2} = 0 \quad \text{at} \quad \omega = 0 \tag{2.16}$$

implies that the phase response is as near linear as possible in the pass-band. We take (2.16) as the definition of the Flat-delay (FD) filter. The condition (2.16) implies $\omega_0 = \pm\sigma_0/\sqrt{3}$ which, in conjunction with (2.12) yields

$$\sigma_0 = \sqrt{\frac{3\gamma}{4}}, \qquad \omega_0 = \pm\sqrt{\frac{\gamma}{4}}, \tag{2.17}$$

where $\gamma = (1+\sqrt{5})/2$. Devotees of Fibonacci (1175?–1250?) will recognise this as the *golden ratio*, which pops up in the most surprising places. The representative point for the FD filter is shown in Fig. 2.2. The Flat-delay filter is a special case of a *Bessel* filter (see Kuo, 1966), a filter whose phase response is maximally flat and which can be constructed using Bessel polynomials.

(c) Transient Decay and Start-up Time

The transient response of the filter decays with an e-folding time $1/\sigma_0$. Thus, the most rapid decay obtains for the largest possible value of σ_0. This is obviously the point where the curve (2.12) intersects the horizontal axis, with coordinates

$$\sigma_0 = \sqrt{\sqrt{2}+1} = \sqrt{\frac{1}{\sqrt{2}-1}}, \qquad \omega_0 = 0 \tag{2.18}$$

(see Fig. 2.2). The output of the filter based on this choice should reach a steady state faster than for any other prototype filter. It corresponds to the case of critical damping for a harmonic oscillator. We shall call the filter based on this choice the Quick-start (QS) filter (no more respectable name having been found in the literature).

The time-constant $1/\sigma_0$ is a measure of the decay-rate of the transient response of a filter. We shall define this to be the *start-up* time

$$T_0 = 1/\sigma_0. \tag{2.19}$$

In time T_0, the transient is reduced by $1/e$, or to 37% of its initial value. By $2T_0$ it has decayed to 14%, and after $3T_0$ to 5% of its starting value. The delay and start-up times for the three special filters which we have discussed are tabulated below. Note that, while the delay times are roughly the same for all three filters, the start-up times are very different. For the QS filter the start-up time is equal to half the delay.

We define the non-dimensional ratio of the start-up and delay times:

$$\Gamma = \frac{\text{Start-up}}{\text{Delay}} = \frac{T_0}{\delta_0}.$$

For application to initialization it is desirable that Γ be as small as possible, so as to achieve maximum attenuation of the transient within the delay period. The minimum value $\Gamma = \frac{1}{2}$ is attained by the Quick-Start filter.

The amplitude response and group delay of the three special prototype filters (BW: Butterworth, FD: Flat-Delay, QS: Quick-Start) are shown in Fig. 2.3. The flat amplitude response of the BW filter is an attractive feature, but is counter-balanced by a group delay which is far from constant. The FD filter clearly has the most nearly constant delay in the pass band, but its amplitude response is poorer than for BW. The QS filter amplitude response is similar to that of FD; its group delay is not as good, but is more nearly constant than that of the BW filter for low frequencies.

2.4 From Analog to Digital Filters

For a general analog filter the output is a weighted sum of derivatives of the input and output signals. Such a filter can be described by a constant-coefficient differential equation

$$y(t) = \sum_{k=0}^{K} a_k \frac{d^k x(t)}{dt^k} + \sum_{k=1}^{N} b_k \frac{d^k y(t)}{dt^k}. \tag{2.21}$$

If we assume an input of the form $x(t) = \exp(st)$ and seek a solution $y(t) = H(s)x(t)$ for the output, the transfer function $H(s)$ must be of the form

$$H(s) = \frac{\sum_{k=0}^{K} a_k s^k}{1 - \sum_{k=1}^{N} b_k s^k}. \tag{2.22}$$

This is a ratio of polynomials, or a rational function of s. It may be expressed as

$$H(s) = G\frac{(s-\zeta_1)(s-\zeta_2)\cdots(s-\zeta_K)}{(s-p_1)(s-p_2)\cdots(s-p_N)}, \tag{2.23}$$

determined completely by the gain G and the positions of the zeros ζ_i and poles p_j.

There are two means of converting a filter from analog to digital form, *i.e.,* of going over from continuous time to discrete time. We can discretise the differential equation (2.21) or transform the response function $H(s)$ to the discrete domain. The two possibilities are closely related, as we shall now show. For simplicity, we consider the first order filter

$$\dot{y} + by = cx. \tag{2.24}$$

The transfer function can be written down by inspection:

$$H(s) = \frac{c}{s+b}, \tag{2.25}$$

so that, for an input $x(t) = \exp(st)$, the output is

$$y(t) = \frac{c}{s+b}\, x(t), \tag{2.26}$$

Discretising (2.24) in time with centred approximations, yields

$$\left(\frac{y_k - y_{k-1}}{\Delta t}\right) + b\left(\frac{y_k + y_{k-1}}{2}\right) = c\left(\frac{x_k + x_{k-1}}{2}\right).$$

This enables the output y_k to be expressed in terms of inputs and previous outputs:

$$y_k = \left(\frac{c\Delta t}{2 + b\Delta t}\right)(x_k + x_{k-1}) + \left(\frac{2 - b\Delta t}{2 + b\Delta t}\right) y_{k-1}\, . \tag{2.27}$$

We can define *advance* and *delay* operators Z and Z^{-1} by

$$Zx_k = x_{k+1} \qquad Z^{-1}x_k = x_{k-1}$$

and write (2.27) in the form

$$y_k = \frac{c\Delta t}{2 + b\Delta t}(1 + Z^{-1})x_k + \frac{2 - b\Delta t}{2 + b\Delta t}Z^{-1}\, y_k\, .$$

This can be solved immediately for y_k in the following form:

$$y_k = \frac{c}{\frac{2}{\Delta t}\left(\frac{Z-1}{Z+1}\right) + b}\, x_k. \tag{2.28}$$

This is formally similar to (2.26) if we make the identification

$$s = \frac{2}{\Delta t}\left(\frac{Z-1}{Z+1}\right). \tag{2.29}$$

Thus, instead of starting from the differential equation (2.24), we can start with the analog transfer function (2.25) and make the substitution of Z for s using the *bilinear transformation* (2.29). It is normally much easier to work with transfer functions than with differential equations. The bilinear transformation will be discussed further in the following sections.

3. DIGITAL FILTERS

In this section the definitions and basic properties of digital filters are presented. The simplest DFs are non-recursive—the input is processed to produce the output. In contrast, recursive DFs involve *feedback* of the output, which enables them to achieve higher performance for given computational expenditure. Both types of DF are discussed in this section.

Numerous accounts of digital filters are available in publications on digital signal processing; a number of the most useful books on the subject are listed at the end of this report. The book by Williams (1986) is an excellent elementary introduction. The work of Oppenheim and Schafer (1989) is very comprehensive, at a more advanced level. Hamming's book (1989) comes somewhere in between.

3.1 Definition of Digital Filters

Given a discrete function of time $\{x_n\}$ a *nonrecursive* digital filter is defined by

$$y_n = \sum_{k=-N}^{N} a_k x_{n-k}. \tag{3.1}$$

The output y_n at time $n\Delta t$ depends on both past and future values of x_n, but not on other output values. A *recursive* digital filter is defined by

$$y_n = \sum_{k=K}^{N} a_k x_{n-k} + \sum_{k=1}^{L} b_k y_{n-k}. \tag{3.2}$$

The output y_n at time $n\Delta t$ in this case depends on past and present values of the input (for $K = 0$), and also on previous output values. (Occasionally, future input values are also used ($K < 0$), in which case the recursive filter is *non-causal*). Recursive filters are more powerful than non-recursive ones, but can also be more problematical, as the feedback of the output can give rise to instability.

The response of a nonrecursive filter to an impulse $\delta(n)$ is zero for $|n| > N$, giving rise to the alternative name *finite impulse response* or FIR filter. Recursive filters have longer memories: since the response of a recursive filter to an impulse input can persist indefinitely, it is known as an *infinite impulse response* or IIR filter. For example, the response of the simple filter

$$y_{n+1} = \beta x_n + \alpha y_n \qquad (y_0 = 0)$$

to the unit pulse input $\delta(n)$ is the geometric progression

$$y_1 = \beta, \quad y_2 = \beta\alpha, \quad y_3 = \beta\alpha^2, \quad y_4 = \beta\alpha^3, \quad \cdots .$$

The response continues indefinitely; for $|\alpha| \leq 1$ the response is bounded; for $|\alpha| > 1$ it grows without limit and the filter is unstable.

The frequency response of a digital filter is easily found: let $x_n = \exp(in\theta)$ and assume an output of the form $y_n = H(\theta)\exp(in\theta)$; substituting into (3.2), the transfer function $H(\theta)$ is

$$H(\theta) = \frac{\sum_{k=K}^{N} a_k e^{-ik\theta}}{1 - \sum_{k=1}^{L} b_k e^{-ik\theta}}. \tag{3.3}$$

For nonrecursive filters the denominator reduces to unity. This equation gives the response once the filter coefficients a_k and b_k have been specified. However, what is really required is the opposite: to derive coefficients (and as few as possible) which will yield the desired response function. This *inverse problem* has no unique solution, and a great variety of techniques have been developed. Only the most elementary design techniques will be considered below; for further information see the DSP references.

Recursive filters generally have superior performance to nonrecursive filters with the same total number of coefficients. This may be explained by noting that the transfer function (3.3) can be written

$$H(z) = \frac{\sum_{k=K}^{N} a_k z^{-k}}{1 - \sum_{k=1}^{L} b_k z^{-k}} \tag{3.4}$$

where $z = \exp(i\theta)$. For a nonrecursive filter this is a polynomial in $1/z$; for a recursive function it is a rational function in $1/z$, and is more capable of fitting a specified function having sudden changes or narrow features. Against this, the recursive filter will obviously cause problems if the denominator vanishes. It can be shown that a recursive filter is stable if the roots of the *characteristic polynomial*

$$z^L - \sum_{k=1}^{L} b_k z^{L-k} = 0$$

are inside the unit circle $|z| \leq 1$.

3.2 Transient and Steady-State Solutions

The output of the nonrecursive filter (3.1) for an input $x_n = z^n$ is of the form $y_n = Hz^n$, where H is easily derived by substitution of the assumed forcing and solution:

$$H = \big(a_0 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_N z^{-N}\big).$$

This agrees with the general expression (3.4). For a nonrecursive filter, the response function has only zeros. There are no poles, and the possibility of an unstable response does not arise. No initial values of y are required. There are several design techniques for choosing the coefficients $\{a_n\}$ (see Huang and Lynch, 1992, Appendix A). For a given choice, the degree of filtering depends sensitively upon the span of the filter, which is proportional to the filter length N. The situation for recursive filters is more complicated. Here the effectiveness of the filter is influenced not only by the forced response, but also by transient components whose rate of decay determines when a steady state response is reached.

The recursive filter (3.2) can be written as a difference equation of order L:

$$y_n - \sum_{k=1}^{L} b_k y_{n-k} = \sum_{k=K}^{N} a_k x_{n-k}. \tag{3.5}$$

For simplicity, we restrict attention to the second-order filter

$$y_n - b_1 y_{n-1} - b_2 y_{n-2} = (a_0 x_n + a_1 x_{n-1} + a_2 x_{n-2}\ ) = f_n. \tag{3.6}$$

The general solution can be constructed from two distinct parts, a transient component and a forced component. The transient component is a solution of the homogeneous equation

$$y_n - b_1 y_{n-1} - b_2 y_{n-2} = 0. \tag{3.7}$$

Assuming a solution of the form $y_n = z^n$ leads to the characteristic equation

$$z^2 - b_1 z - b_2 = 0 \tag{3.8}$$

which has a pair of solutions $z = z_1$ and $z = z_2$, either both real or complex conjugates if the coefficients b_1 and b_2 are real. Thus, the transient solution takes the form

$$y_n^T = C_1 z_1^n + C_2 z_2^n. \tag{3.9}$$

Clearly the term *transient* is appropriate only if $|z_1| < 1$ and $|z_2| < 1$. This is identical to the requirement for the filter to be stable, and we assume it to be fulfilled; the transient then decays exponentially with increasing n. Next, consider the case of exponential forcing, $x_n = z^n$, and seek a response of the form $y_n = Hz^n$. Substitution in (3.6) yields

$$H = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - b_1 z^{-1} - b_2 z^{-2}}, \tag{3.10}$$

in agreement with the general expression (3.4). The complete solution is the sum of the transient and forced response

$$y_n = \left(C_1 z_1^n + C_2 z_2^n\right) + H z^n. \tag{3.11}$$

The constants C_1 and C_2 are determined by the starting values y_0 and y_1, which must be supplied.

A filter with response of the form (3.10) is known as a *biquad*, since H is quadratic in both numerator and denominator. These filters are fundamental building blocks, and higher-order filters can be constructed by combining biquad components in cascade (series) or parallel. The response of a cascade is the product of the responses of the components; the response of a parallel combination is the sum:

$$H_{\text{CASCADE}} = \prod_{k=1}^{K} H_k\,, \qquad H_{\text{PARALLEL}} = \sum_{k=1}^{K} H_k.$$

For a stable filter the transient component dies away exponentially. But how quickly? The rate of decay depends on the distance of the poles z_1, z_2 from the unit circle, that is on $|z_1|$ and $|z_2|$. Therefore, *ceteris paribus*, we should arrange for the poles to have the smallest possible moduli; this is the idea which was used to devise the Quick-Start analog filter, and the same approach may be used in the digital domain. In the application to initialization we are interested in the forced response, and would like to eliminate the transient altogether; but this can be done only if we know the initial values which imply $C_1 = C_2 = 0$, and we have no idea how to choose these. One possible solution is to apply the filter iteratively over a short span $0 \le n \le N$, using the same forcing x_n each time but taking the final output y_N as a starting value for the next iteration. This technique will be used in the application below. For a fuller discussion of transient decay, see Hamming (1989, §13.12).

4. NON-RECURSIVE (FIR) DIGITAL FILTERS

4.1 Design of Nonrecursive Filters

Consider a function of time, $f(t)$, with low and high frequency components. To filter out the high frequencies one may proceed as follows:

[1] calculate the Fourier transform $F(\omega)$ of $f(t)$;

[2] set the coefficients of the high frequencies to zero;

[3] calculate the inverse transform.

Step [2] may be performed by multiplying $F(\omega)$ by an appropriate weighting function $H_c(\omega)$. Typically, $H_c(\omega)$ is a step function

$$H_c(\omega) = \begin{cases} 1, & |\omega| \le |\omega_c|; \\ 0, & |\omega| > |\omega_c|, \end{cases} \tag{4.1}$$

where ω_c is a cutoff frequency. These three steps are equivalent to a convolution of $f(t)$ with $h(t) = \sin(\omega_c t)/\pi t$, the inverse Fourier transform of $H_c(\omega)$. This follows from the convolution theorem

$$\mathcal{F}\{(h * f)(t)\} = \mathcal{F}\{h\} \cdot \mathcal{F}\{f\} = H_c(\omega) \cdot F(\omega) \tag{4.2}$$

Thus, to filter $f(t)$ one calculates

$$f^\star(t) = (h * f)(t) = \int_{-\infty}^{+\infty} h(\tau) f(t-\tau) d\tau. \tag{4.3}$$

For simple functions $f(t)$, this integral may be evaluated analytically. In general, some method of approximation must be used.

Suppose now that f is known only at discrete moments $t_n = n\Delta t$, so that the sequence $\{\cdots, f_{-2}, f_{-1}, f_0, f_1, f_2, \cdots\}$ is given. For example, f_n could be the value of some model variable at a particular grid point at time t_n. The shortest period component which can be represented with a time step Δt is $\tau_N = 2\Delta t$, corresponding to a maximum frequency, the so-called Nyquist frequency, $\omega_N = \pi/\Delta t$. The sequence $\{f_n\}$ may be regarded as the Fourier coefficients of a function $F(\theta)$:

$$F(\theta) = \sum_{n=-\infty}^{\infty} f_n e^{-in\theta},$$

where $\theta = \omega\Delta t$ is the *digital frequency* and $F(\theta)$ is periodic, $F(\theta) = F(\theta + 2\pi)$. High frequency components of the sequence may be eliminated by multiplying $F(\theta)$ by a function $H_d(\theta)$ defined by

$$H_d(\theta) = \begin{cases} 1, & |\theta| \le |\theta_c|; \\ 0, & |\theta| > |\theta_c|, \end{cases} \tag{4.4}$$

where the cutoff frequency $\theta_c = \omega_c \Delta t$ is assumed to fall in the Nyquist range $(-\pi, \pi)$ and $H_d(\theta)$ has period 2π. This function may be expanded:

$$H_d(\theta) = \sum_{n=-\infty}^{\infty} h_n e^{-in\theta} \qquad ; \qquad h_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(\theta) e^{in\theta} d\theta. \tag{4.5}$$

The values of the coefficients h_n follow immediately from (4.4) and (4.5):

$$h_n = \frac{\sin n\theta_c}{n\pi}. \tag{4.6}$$

Let $\{f_n^\star\}$ denote the low frequency part of $\{f_n\}$, from which all components with frequency greater than θ_c have been removed. Clearly,

$$H_d(\theta) \cdot F(\theta) = \sum_{n=-\infty}^{\infty} f_n^\star e^{-in\theta}.$$

The convolution theorem for Fourier series now implies that $H_d(\theta) \cdot F(\theta)$ is the transform of the convolution of $\{h_n\}$ with $\{f_n\}$:

$$f_n^\star = (h * f)_n = \sum_{k=-\infty}^{\infty} h_k f_{n-k}. \tag{4.7}$$

This enables the filtering to be performed directly on the given sequence $\{f_n\}$. It is the discrete analogue of (4.3). In practice the summation must be truncated at some finite value of k. Thus, an approximation to the low frequency part of $\{f_n\}$ is given by

$$f_n^\star = \sum_{k=-N}^{N} h_k f_{n-k}. \tag{4.8}$$

Comparing (4.8) with (3.1), it is apparent that the finite approximation to the discrete convolution is formally identical to a nonrecursive digital filter.

As is well known, truncation of a Fourier series gives rise to Gibbs oscillations. These may be greatly reduced by means of an appropriately defined "window" function. The response of the filter is improved if h_n is multiplied by the Lanczos window

$$w_n = \frac{\sin\big(n\pi/(N+1)\big)}{n\pi/(N+1)}.$$

The transfer function $H(\theta)$ of a filter is defined as the function by which a pure sinusoidal oscillation is multiplied when subjected to the filter. For symmetric coefficients, $h_k = h_{-k}$, it is real, implying that the phase is not altered by the filter. Then, if $f_n = \exp(in\theta)$, one may write $f_n^\star = H(\theta) \cdot f_n$, and $H(\theta)$ is easily calculated by substituting f_n in (4.8):

$$H(\theta) = \sum_{k=-N}^{N} h_k e^{-ik\theta} = \left[h_0 + 2\sum_{k=1}^{N} h_k \cos k\theta \right] . \tag{4.9}$$

The transfer functions for a windowed and unwindowed filter are shown in Fig. 4.1. These were calculated for the cutoff period $\tau_c = 6$ hours, span $T_s = 2N\Delta t = 6$ hours and timestep $\Delta t = 360\, s$ (as used in the application to initialization described below). The parameter values are therefore $N = 30$ and $\theta_c = \pi/30 \approx 0.1$. It can be seen that the use of the window decreases the Gibbs oscillations in the stop-band $|\theta| > |\theta_c|$. However, it also has the effect of widening the pass-band beyond the nominal cutoff. For a fuller discussion of windowing see *e.g.* Hamming (1989) or Oppenheim and Schafer (1989).

The simplest design method for nonrecursive filters is the expansion of the desired filtering function, $H(\theta)$, as a Fourier series, and the application of a suitable window function to improve the transfer characteristics. A more sophisticated method uses the Chebyshev alternation theorem to obtain a filter whose maximum error in the pass- and stopbands is minimized. This method yields a filter meeting required specifications with fewer coefficients that the other methods. Such filters are called optimal or equi-ripple filters. The frequency response of four filters is shown in Figure 4.2. Two of the filters are constructed using a Lanczos window and two by means of the optimal design technique (for full details, see Huang and Lynch, 1992). The solid graph is for a Lanczos filter with a nominal cutoff of 6 hours and a span $T_s = 6$ hours (the filter used in L&H). The dotted curve is for the optimal or equiripple filter with a span $T_s = 3$ hours. It can be seen that the behaviour in the transition region is quite similar for these two filters. However, the optimal filter has sidelobes with amplitude of about 0.1 (determined by the value of δ) in the stop-band. The remaining two curves in Figure 4.2 are for the Lanczos filter with nominal cutoff of 6 hours and span $T_s = 3$ hours (dashed line) and the optimal filter with a span $T_s = 2$ hours (dot-dash line). These two filters have similar response in the pass and transition bands. The optimal filter has equiripple oscillations in the stop-band. What is clear from these curves is that, if the stop-band ripples can be tolerated, the optimal filters have the required response in the pass band with shorter span than that required by the Lanczos filters.

The time required for filtering is proportional to the filter span. Thus, a 50% reduction is achieved by replacing the Lanczos filter with 6 hour span by an optimal filter with a 3 hour span. It is shown in Huang and Lynch (1992) that the optimal filter applied to initialization of the HIRLAM model yields results comparable to those obtained with the Lanczos filter.

4.2 Application of an FIR: Initialization using a Nonrecursive Filter

An initialization scheme using a nonrecursive digital filter has been developed by Lynch and Huang (1991) for the HIRLAM model. The value chosen for the cutoff frequency corresponded to a period $\tau_c = 6$ hours. With the time step $\Delta t = 6$ minutes used in the model, this corresponds to a (digital) cutoff frequency $\theta_c = \pi/30$. The coefficients were derived by Fourier expansion of a step-function, truncated at $N = 30$, with application of a Lanczos window, and are given by

$$h_n = \left[\frac{\sin\big(n\pi/(N+1)\big)}{n\pi/(N+1)}\right]\left(\frac{\sin(n\theta_c)}{n\pi}\right).$$

The frequency response was depicted in Fig. 4.1. The central lobe of the coefficient function spans a period of six hours, from $t = -3$ hours to $t = +3$ hours. The summation in (4.8) was calculated over this range, with the coefficients normalized to have unit sum over the span. Thus, the application of the technique involved computation equivalent to sixty time steps, or a six hour adiabatic integration.

The uninitialized fields of surface pressure, temperature, humidity and winds were first integrated forward for three hours, and running sums of the form

$$f_F^\star(0) = \tfrac{1}{2}h_0 f_0 + \textstyle\sum_{n=1}^{N} h_{-n} f_n, \tag{4.10}$$

where $f_n = f(n\Delta t)$, were calculated for each field at each gridpoint and on each model level. These were stored at the end of the three hour forecast. The original fields were then used to make a three hour 'hindcast', during which running sums of the form

$$f_B^\star(0) = \tfrac{1}{2}h_0 f_0 + \textstyle\sum_{n=-1}^{-N} h_{-n} f_n \tag{4.11}$$

were accumulated for each field, and stored as before. The two sums were then combined to form the required summations:

$$f^\star(0) = f_F^\star(0) + f_B^\star(0). \tag{4.12}$$

These fields correspond to the application of the digital filter (4.8) to the original data, and will be referred to as the filtered data.

Fig. 4.3 shows the evolution of N_1, the areally averaged surface pressure tendency, for a 24 hour forecast starting from uninitialized data (solid curve) and from the digitally filtered fields (dashed curve). For comparison, the result of applying a normal mode initialization is also shown (dotted curve). It is clear that the filtering of the initial fields results in removal of the spuriously large tendencies which are found in the uninitialized run. The difference between the two forecasts was remarkably small: the rms surface pressure difference was only $0.07\,hPa$. For further details, see Lynch and Huang (1992).

In Fig. 4.4 the results of applying four different filters to diabatic initialization are shown. The result of NMI is also depicted. It is seen that the evolution of the quantity N_1 is less noisy for all these filters than in the case of normal mode initialization. In particular, the results for the optimal filter with three hour span (dotted curve) are acceptably close to that obtained with the six hour span Lanczos filter, enabling the integration time to be reduced by 50%.

Diabatic initialization with a nonrecursive filter is implemented by performing a backward adiabatic integration for half the span, and using the resulting fields as starting values for a forward diabatic integration for the full span. This is necessary due to the irreversibility of the diabatic processes. The results of Huang and Lynch (1992) show that the technique works satisfactorily. However, the forward diabatic integration will not coincide, at time $t = 0$, with the original analysis. Thus, we are filtering values on a trajectory which does not pass through the initial data. Naïve attempts to compensate for this (minor) effect were unsuccessful. What is required is a filtering method which enables us to deduce initialized starting values from a one-sided diabatic integration, *i.e.* from a forward integration starting from the analysis time. This is possible with a recursive filter.

5. RECURSIVE (IIR) DIGITAL FILTERS

5.1 Design of Recursive Filters

The design of recursive or IIR filters is more difficult than that of nonrecursive or FIR filters. Several techniques are described in the DSP references at the end of this report. Only one such method will be described below: in this approach, a classical analog lowpass response is specified; a transformation of variables then converts this to discrete time in such a way that the required filter coefficients may be deduced.

5.1.1 *Classical Analog Filter Response*

There are numerous classical filter functions which may be used as a basis for digital filter design. They are determined by the manner in which the ideal lowpass filter response is approximated in the pass- and stop-bands. If a Taylor series approximation truncated at N terms is applied at $\omega = 0$ and $\omega = \infty$, the result is a Butterworth filter. The Chebyshev (type I) filter uses a minimax approximation across the passband and a Taylor series at $\omega = \infty$. The inverse, or type II, Chebyshev filter uses a Taylor series expansion at $\omega = 0$ and a minimax approximation across the stopband. The elliptic filter involves a minimax approximation in both the passband and the stopband. The Bessel filter is derived by continued fraction approximation of an exponential, to achieve a phase response which is as near to linear as possible.

It is important to choose the type of filter appropriate to the problem. All the classical types are optimal in one sense or another. The Butterworth filter has an amplitude response which is as flat as possible at the low-frequency limit $\omega = 0$. The Chebyshev filter gives the smallest maximum error over the passband of any filter having similar Taylor series accuracy at $\omega = \infty$. In a complimentary way, the inverse Chebyshev filter minimizes the maximum deviation from zero in the stopband. The elliptic filter involves four parameters (passband ripple, transition width, stopband ripple and filter order) and for given values of any three minimizes the fourth. The Bessel filter has a group delay that is as near constant as possible for a given filter order.

The transfer function for the Butterworth filter of order N has a particularly simple form; for cutoff frequency ω_c it is

$$\left|H(i\omega)\right|^2 = \frac{1}{1+(\omega/\omega_c)^{2N}}. \tag{5.1}$$

As a function of s, the parameter appearing in the Laplace transform, the response of the prototype filter (with cutoff frequency $\omega_c = 1$) is such that

$$H(s)\cdot H(-s) = \left|H(s)\right|^2 = \frac{1}{1+(-s^2)^N}. \tag{5.2}$$

This function has $2N$ poles evenly spaced around the unit circle. To ensure stability of the filter, the N poles in the left half-plane are selected for $H(s)$; $H(-s)$ will then have the remaining poles. There is a simple formula for the poles $s_k = u_k + iv_k$:

$$u_k = -\sin\left[\frac{(2k+1)\pi}{2N}\right] \quad , \quad v_k = \cos\left[\frac{(2k+1)\pi}{2N}\right] \tag{5.3}$$

with $k = 0, 1, 2, \ldots, N-1$. The Butterworth filter is called *maximally flat*, since the first $2N-1$ derivatives vanish at $\omega = 0$.

The response function of the N-th–order Bessel filter takes the form

$$H(s) = \frac{B_N(0)}{B_N(s)} \tag{5.4}$$

where $B_N(s)$ is a Bessel polynomial, defined by the recursion formula

$$B_0 = 1 \qquad B_1 = s+1 \quad \cdots \quad B_n = (2n-1)B_{n-1} + s^2 B_{n-2}\ .$$

In particular, the second-order Bessel filter is

$$H(s) = \frac{3}{s^2+3s+3}, \tag{5.5}$$

and the normalised prototype version (with cutoff frequency $\omega_c = 1$ and $H(0) = 1$) is

$$H(s) = \frac{\gamma}{s^2 + \sqrt{3\gamma}\, s + \gamma} \tag{5.6}$$

where $\gamma = (1+\sqrt{5})/2$. This filter has a phase response which is as near linear as possible, and thus a delay which is maximally flat, for a second-order filter. We also refer to it as the Flat-Delay filter.

The N-pole filter whose transient solution decays most rapidly corresponds to a critically damped oscillator; it has all its poles on the negative real axis, at $s_p = -\sigma$:

$$H(s) = \frac{G}{(s+\sigma)^N}. \tag{5.7}$$

The prototype version ($\omega_c = 1$) corresponds to the value $\sigma = \sqrt{1/(2^{1/N}-1)}$. This is what we propose to call the N-th–order Quick-Start filter.

5.1.2 *Conversion of Analog to Digital Filter*

There are several methods of deriving a digital transfer function from one of the classical analog expressions. They all involve some mapping from the s-plane to the z-plane, chosen to preserve properties such as optimality of the filter. The matched $\mathcal{Z}$-transform design procedure is conceptually the simplest (although not the most effective). Poles and zeros of the transfer function $H(s)$ are mapped directly to poles and zeros of $H(z)$ by a simple substitution. Consider the inverse transform of a simple pole

$$\mathcal{L}^{-1}\left\{\frac{1}{s-\alpha}\right\} = \exp(\alpha t). \tag{5.8}$$

If this is sampled at intervals Δt and the $\mathcal{Z}$-transform calculated, one has:

$$\mathcal{Z}\{\exp(\alpha n\Delta t)\} = \frac{1}{1-e^{\alpha\Delta t}z^{-1}}. \tag{5.9}$$

(the definition and elementary properties of the $\mathcal{Z}$-transform are given in Lynch, 1991a; a comprehensive treatment may be found in Oppenheim and Schafer, 1989). Now assume that $H(s)$ is rational and is split into factors $(s-\alpha)$. The relations (5.8) and (5.9) suggest the substitution $(s-\alpha) \to (1-\exp(\alpha\Delta t)/z)$. Allowance for a general cutoff frequency is made by the change $s \to s/\omega_c$. Combining these, the transformation from prototype analog to general digital transfer function is achieved by

$$(s-\alpha) \longrightarrow (1-e^{\alpha\omega_c\Delta t}z^{-1}) \tag{5.10}$$

mapping the analog pole or zero at $s=\alpha$ to the digital pole or zero at $z=\exp(\alpha\omega_c\Delta t)$. This produces a transfer function $H(z)$ which is a rational function of $1/z$. The filter coefficients can then be ascertained by comparison of $H(z)$ with the general expression for the response function of an IIR filter:

$$H(z) = \frac{\sum_{k=K}^{N} a_k z^{-k}}{1-\sum_{k=1}^{L} b_k z^{-k}} \tag{5.11}$$

The matched $\mathcal{Z}$-transform procedure is very easy to apply, but has the disadvantage that an all-pole analog filter becomes an all-pole digital filter, with no zeros to help shape the frequency response. A more powerful technique is the bilinear transformation. The theory behind this procedure can be found in the literature on digital signal processing; the basic idea was introduced *via* a simple example in §2.4. The definition of the bilinear transform is as follows: the mapping from the s-plane to the z-plane is given by

$$s = \frac{1}{\mu_c}\left[\frac{z-1}{z+1}\right] \tag{5.12}$$

where $\mu_c = \tan(\theta_c/2)$ with $\theta_c = \omega_c \Delta t = 2\pi\Delta t/\tau_c$ defining the cutoff frequency. Under this transformation the imaginary axis $s = i\omega$ of the s-plane maps into the unit circle $z = \exp(i\theta)$ in the z-plane

$$\omega = \frac{\tan(\theta/2)}{\tan(\theta_c/2)}$$

in such a way that $\omega = 1$, the cutoff frequency of the prototype analog filter, corresponds to θ_c (this is the reason for the factor μ_c in (5.12)). Conversion from a prototype analog filter to a digital filter with cutoff frequency θ_c is implemented by the substitution of (5.12) into the transfer function $H(s)$ to obtain a function of z. The coefficients are deduced by comparing the result with the general expression (5.11).

5.2 Some Examples of Recursive Filter Design

The above theory may be made clearer by consideration of a few concrete examples. We first compare the two alternative transformations (5.10) and (5.12), applied to the second order Butterworth filter. From (5.2), this has a squared amplitude response

$$\left|H(s)\right|^2 = \frac{1}{1+s^4} = \left[\frac{1}{s^2+\sqrt{2}s+1}\right]\cdot\left[\frac{1}{s^2-\sqrt{2}s+1}\right] \tag{5.13}$$

The first term has two poles in the left half-plane so, to ensure stability of the filter, $H(s)$ is defined in terms of these:

$$H(s) = \left[\frac{1}{s^2+\sqrt{2}s+1}\right] = \left[\frac{1}{(s-s_1)(s-s_2)}\right] \tag{5.14}$$

where $s_1 = (-1+i)/\sqrt{2}$ and $s_2 = (-1-i)/\sqrt{2}$. If the matched $\mathcal{Z}$-transform approach is used, $H(s)$ is converted to $H(z)$ by means of (5.10), resulting in

$$H(z) = \frac{z^2}{[z-\exp(s_1\theta_c)][z-\exp(s_2\theta_c)]} \tag{5.15}$$

with $\theta_c = \omega_c\Delta t$. This has two poles within the unit circle, and a double zero at the origin. Defining $z_1 = \exp(s_1\theta_c)$ and $z_2 = \exp(s_2\theta_c)$, it may be written

$$H(z) = \left[\frac{z^2}{(z-z_1)(z-z_2)}\right] = \left[\frac{1}{1-(z_1+z_2)z^{-1}-(-z_1z_2)z^{-2}}\right]. \tag{5.16}$$

This response function corresponds to a filter of the form

$$y^{n+1} = a_0 x^{n+1} + \left(b_1 y^n + b_2 y^{n-1}\right)$$

where, by comparing (5.16) with (5.11), the coefficients can be read off:

$$a_0 = 1 \qquad b_1 = (z_1 + z_2) \qquad b_2 = -z_1 z_2.$$

Alternatively, if the bilinear transformation (5.12) is applied to (5.14), the resulting transfer function is

$$H(z) = \frac{\mu_c^2 (z+1)^2}{(z-1)^2 + \sqrt{2}\mu_c (z^2 - 1) + \mu_c^2 (z+1)^2} \tag{5.17}$$

where $\mu_c = \tan(\theta_c/2)$. Since there is a (double) zero at $z = -1$, the highest frequency is completely annihilated, which is a desirable feature for a lowpass filter. The transfer function corresponds to a filter of the form

$$y^{n+1} = \left(a_0 x^{n+1} + a_1 x^n + a_2 x^{n-1}\right) + \left(b_1 y^n + b_2 y^{n-1}\right)$$

where, once again, the coefficients can be deduced by expressing (5.17) in the same form as (5.11), yielding

$$a_0 = \tfrac{1}{2} a_1 = a_2 = \frac{\mu_c^2}{1 + \sqrt{2}\mu_c + \mu_c^2} \qquad b_1 = \frac{2(1 - \mu_c^2)}{1 + \sqrt{2}\mu_c + \mu_c^2} \qquad b_2 = -\frac{1 - \sqrt{2}\mu_c + \mu_c^2}{1 + \sqrt{2}\mu_c + \mu_c^2}.$$

The magnitude of the response for the two filters (5.16) and (5.17) is shown in Fig. 5.1. Both filters have flat response curves for the lowest frequencies, which is a desirable reflection of the character of the analog filter on which they are based (see Fig. 2.3(top)). The bilinear transform filter (Fig. 5.1(a)) ensures complete removal of the highest frequency. For nominal cutoff frequency $\theta_c > \pi/4$ the matched $\mathcal{Z}$-transform filter (Fig. 5.1(b)) amplifies some frequencies (assuming response normalized by $H(0) = 1$); this is generally unacceptable, and is a distortion of the character of the original analog filter. The filter based on the bilinear transform is free from this defect; for this reason and also for its alias-free nature, the bilinear transform is the preferred choice. [Note: Fig. 5.4(c) in Lynch 1991b is erroneous; the correct version appears in Fig. 5.1(a) of the report you are reading.]

The amplitude response and group delay for three filters (BW: Butterworth; FD: Flat Delay; QS: Quick-Start) are shown in Fig. 5.2. These are derived from the prototypes discussed in §2 by means of the bilinear transformation. The parameter values are $\Delta t = 360\,\mathrm{s}$ and $\tau_c = 12\,\mathrm{h}$, which imply $\theta_c = \pi/60$. The close similarity between the analog and digital forms can be seen by comparing Fig 5.2 with Fig. 2.3.

5.3 Relationship between Analog and Digital Filter Parameters

The definition of the bilinear transformation is such as to ensure that the (digital) cutoff frequency θ_c maps to the value $\omega_c = 1$, the cutoff of the prototype analog filter. The transformation is

$$s = \frac{1}{\mu_c}\left[\frac{z-1}{z+1}\right], \qquad z = \left[\frac{1+\mu_c s}{1-\mu_c s}\right] \tag{5.18}$$

with $\mu_c = \tan(\theta_c/2)$. For pure imaginary values $s = i\omega$ it becomes (with $z = \exp(i\theta)$)

$$\omega = \frac{\tan(\theta/2)}{\mu_c}, \qquad \text{so that} \qquad [\omega = 1] \Rightarrow [\theta = \theta_c]. \tag{5.19}$$

We must also investigate the effect of this mapping on the other filter parameters.

The group delay of the digital filter is defined by

$$\Delta = -\frac{d}{d\theta}(\arg H) = -\frac{d\omega}{d\theta}\frac{d}{d\omega}(\arg H) = \frac{d\omega}{d\theta}\delta.$$

The *delay* Δ_0 is simply the value at the LF limit $\theta = 0$; using (5.19), the (dimensional) value of the delay is

$$\Delta_0 = \frac{\Delta t}{2\mu_c}\delta_0 \approx \frac{\Delta t}{\theta_c}\delta_0 = \left(\frac{\tau_c}{2\pi}\right)\delta_0. \tag{5.20}$$

The start-up time was defined arbitrarily as the time for the transient to decay by a factor $1/e$. It is determined by the position of the pole of greatest modulus of the transfer function. For the second order filters under consideration, the two poles z_p and $\bar{z}_p$ have equal moduli. The pole z_p of the digital filter is related to the pole $s_p = -\sigma_0 + i\omega_0$ of the prototype analog filter by (5.18). We require the value ν such that $|z_p|^\nu = e^{-1}$, which means $\nu = -1/\log|z_p|$. Assuming $|\mu_c s_p| \ll 1$ it is easily shown that $\log|z_p| \approx -2\mu_c\sigma_0$ from which it follows that the (dimensional) start-up time is

$$\nu\Delta t \approx \frac{\Delta t}{2\mu_c\sigma_0} \approx \frac{\Delta t}{\theta_c\sigma_0} = \frac{\tau_c}{2\pi\sigma_0} = \left(\frac{\tau_c}{2\pi}\right)T_0 \tag{5.21}$$

where $T_0 = 1/\sigma_0$ is the start-up for the prototype analog filter.

5.4 Biquads and Higher-Order Filters

In §2 we studied the properties of an *all-pole* filter having a transfer function

$$H(s) = \frac{1}{(s-s_p)(s-\bar{s}_p)}.$$

The bilinear transformation converts this to a digital filter with transfer function

$$H(z) = G\frac{(z+1)^2}{(z-z_p)(z-\bar{z}_p)},$$

where G is a constant chosen so that $H(0)=1$. No increase in complexity of the digital filter results if we consider the more general analog filter

$$H(s) = \frac{(s-s_0)(s-\bar{s}_0)}{(s-s_p)(s-\bar{s}_p)}$$

which becomes, under the bilinear transformation, the *biquad*

$$H(z) = G\frac{(z-z_0)(z-\bar{z}_0)}{(z-z_p)(z-\bar{z}_p)}.$$

This is the most general second-order filter having real coefficients. Components of a particular period τ_0 may be eliminated by setting $z_0 = \exp(i\theta_0)$ where $\theta_0 = 2\pi\Delta t/\tau_0$. If θ_0 is in the stop-band the phase in the pass-band is not changed. Fig. 5.3 shows the amplitude and phase response for two filters with $\Delta t = 360\,\mathrm{s}$ and cutoff period $\tau_c = 12\,\mathrm{h}$. The solid curve derives from an all-pole filter, and is the same as the QS filter response in Fig 5.2. The dashed curve is for a filter chosen to annihilate the component having a period $\tau_0 = 2\,\mathrm{h}$ (the abscissa is $f = \theta/2\pi$, the zero is at $f_0 = \Delta t/\tau_0 = \frac{1}{20}$). Note that the phase in the pass-band is the same for both filters; for the filter with a zero, there is a phase-jump of π as the zero is crossed.

Biquads are important as they may be combined in cascade or parallel to form more complex filters. The poles and zeors of a general filter are combined in complex conjugate pairs to form biquad components. Data to be processed by the general filter are then passed through the biquad components serially (if they are cascaded) or simultaneously (if they are in parallel). If the order of the filter is odd, a single first-order component also occurs. Analysis of a filter into biquad components can result in smaller numerical errors than if the high-order filter is implemented directly.

6. INITIALIZATION WITH A RECURSIVE FILTER

The initialization technique based on a nonrecursive filter, which has been described in §4, has been found to give satisfactory results. However, in the case of diabatic initialization, the diabatic trajectory starting from the backward adiabatic run does not pass through the initial data. A number of modifications of the technique were made to allow for this discrepancy, but they were found to be unsuccessful and even counterproductive. It would be ideal if balanced initial fields could be deduced from a diabatic integration forward from the initial data. Nonrecursive filters have linear phase-error and produce output which is valid at the center of the span; thus, such a filter applied to a forward integration yields data applicable at half the span.

Recursive filters (IIR filters) have more complex phase-errors. Their use in initialization depends upon a simple but crucial idea: *if a filter with a group delay δ is applied for an integration over a time-span $T_s = \delta$, the output at $t = T_s$ may be assumed to apply at the initial time $t = 0$.* This is what is required for initialization, and we will present evidence in the following section in support of this idea.

6.1 The Validity of the Delay Idea

A simple example of the effect of filter delay can be seen in Fig. 6.1. The input is a single component oscillation with a period of 24 hours. The filter used is a second-order Butterworth filter, digitized with the bilinear transformation. In the top panel the cutoff of the filter is $\tau_c = 6\,\mathrm{h}$; in the bottom panel it is $\tau_c = 12\,\mathrm{h}$. In each case the oscillation is passed by the filter without undue attenuation. However, it is subject to a phase change which varies with the cutoff. The delay of the filter with $\tau_c = 6\,\mathrm{h}$ is 1.35 hours; the delay of that with $\tau_c = 12\,\mathrm{h}$ is 2.7 hours (these values follow from (2.15) and (5.20)). It is clear from Fig. 6.1 that the output is delayed relative to the input, and that this delay is greater for the filter with the longer cutoff period. Measurement of the actual delay shows it to be in good agreement with the theoretical value. Note in particular that, for a time equal to the delay, the output is effectively frozen at its initial value. The solution more-or-less *marks time* for a period $t \approx \delta_0$.

To confirm the idea of a delayed solution in a more realistic context, a barotropic model was used to integrate forward from an initial state for various spans, and the difference between the (QS) filter output and the initial data was calculated. The initial field was the output of a 24 hour forecast, which had reached a state of balance and was essentially free from high frequency (HF) components. Thus, since there were no HF components to be filtered out, the only effect of the filter should be a phase shift of the LF input. For comparison, the forecast change for each span was also computed. Let $X(n)$ be the model state at time $n\Delta t$, and $Y(n)$ the filter output at that time. The quantity

$$\Delta_{\text{FORECAST}} = \left[X(n) - X(0)\right]$$

is the forecast change over a span $T_s = n\Delta t$. This is plotted as a function of span in Fig. 6.2(a) and (b) (solid curve, the same in each graph). This forecast change increases approximately in direct proportion to the time. The quantity

$$\Delta_{\text{FILTER}} = \left[Y(n) - X(0)\right]$$

is also plotted, for two values of the filter cutoff. For Fig. 6.2(a) the cutoff is six hours, for Fig. 6.2(b) it is twelve. It is clear that for short spans the output of the filter at time T_s differs little from the initial state. With a 6 hour cutoff (Fig. 6.2, top panel) the rms difference between the filter output and the initial data remains below 2 m for over 1.5 hours. The theoretical delay for this filter is 1.23 h, indicated by the vertical line (this follows from (2.15) and (5.20)). When the filter cutoff is 12 hours (bottom panel) the rms difference does not reach the 2 m level until about 2.5 hours, which is about equal to the theoretical delay, 2.46 h (indicated by the vertical line in Fig. 6.2(b)). This rms difference is much smaller that the rms forecast change occurring during this interval (about 7.25 m). Clearly, provided the span is short enough, the filtered output can reasonably be assumed to be valid at the initial time.

6.2 Iteration of the Filter

In order that the output should be applicable at the initial time, the span over which the filter is applied cannot exceed the delay. If the transient response has been adequately damped within this span, the requisite filtering is achieved. It is evident that the Quick-Start filter, whose transient response is as short as possible, should be attractive for this application. Even so, it is found that insufficient damping is achieved within a span equal to the delay. One solution is to iterate the filtering, using the final output values from one iteration as starting values for the next, but with the same input each time. Generally, two or three iterations are sufficient.

Fig. 6.3 shows an input signal comprising a HF oscillation (period $\tau_1 = 1\,\mathrm{h}$) and the output after each iteration of filtering over a span $T_s = N\Delta t = 2.5\,\mathrm{h}$ (cutoff period $\tau_c = 12\,\mathrm{h}$, QS filter (top panel) and BW filter (bottom). The input for each iteration is

$$x_n = \cos\left(\frac{2\pi n\Delta t}{\tau_1}\right).$$

The filter should virtually eliminate this signal, since it period is much shorter than the cutoff period. However, there is a transient response which takes some time to die out. Since we want output valid at the initial time, the filter span is bounded above by the delay. The response after one iteration of the QS filter (dashed line, top panel) is still about 40% of its initial value; thus, it is necessary to iterate the filtering process. The initial values for the first iteration are $y_0 = y_1 = x_0$; for the p-th iteration, the final value y_N^{p-1} of the $(p-1)$-th iteration is used:

$$y_0^p = y_N^{p-1} \qquad\qquad y_1^p = y_N^{p-1}.$$

The process is seen to converge: the initial values for the filter improve with each iteration, and the amplitude of the transient response decreases relentlessly. After three iterations it has fallen to below 10% of its initial amplitude. The same behaviour is found for the Butterworth filter (bottom panel, Fig. 6.3) but the rate of decay of the transient is slower, in accordance with theory.

It is possible to use a first-order filter (with an appropriate cutoff frequency) to deduce y_1 from y_0 and the input values x_n. Then only the single initial value y_0 is required. In practice it was found that such a procedure was no better, and in some cases worse than the simple ruse of putting $y_1 = y_0$. After all, it is the slowly varying solution which interests us, so the first two output values should be close in magnitude. The real problem is the choice of a value y_0 which will induce the smallest possible transient response. This is the purpose of the iterative procedure.

6.3 Application of an IIR Filter to Initialization

The recursive QS filter was used to initialize data for a barotropic forecast model. The model uses a semi-implicit, semi-Lagrangian differencing scheme. The initial data was the 500 hPa analysis for 00Z, 22 Nov., 1982. The uninitialized geopotential analysis and the 24 hour reference forecast from it are shown in Fig. 6.4. For the experiments reported below, the timestep was set at $\Delta t = 360\,\mathrm{s}$, although the numerical scheme used permits a much larger value. The results of two techniques will be described. In the first the filter is applied repeatedly, as described in §6.2. In the second the filter is applied once only, but to the analysis increment rather than to the whole field.

6.3.1 *Initialization by Iterated Filtering*

A short forecast was made and used as input to the filter. The time-series of values of each dependent variable, at each gridpoint and each model level, were filtered independently; the spatial structures of the fields and the balance relationships between them are implicitly taken into account by the filtering process. The cutoff period of the QS filter was set at $\tau_c = 12\,\mathrm{h}$, and the filter was applied over a span $T_s = 2\,\mathrm{h}$. The delay of this filter is $\Delta_0 = 2.46\,\mathrm{h}$ (longer than the span). The output of the filter was assumed to be valid at the initial time, and was used as initial data for a parallel forecast.

It was found that a single application of the filter reduced the noise significantly but insufficiently, so repeated iterations were made as described in §6.2. The noise levels in the forecasts made from data after one, two and three iterations of the filter can be seen in Fig. 6.5. The top panel shows how the geopotential at a central point evolves during the first six forecast hours. The noise level decreases with each additional iteration. The bottom panel in Fig. 6.5 shows the mean absolute divergence, a global measure of high frequency activity. Once again, the level of noise is reduced by repeated filtering. With three iterations the level is more-or-less constant for the six hours.

Table 6.1 lists the rms and maximum changes in height and winds after each iteration (all figures relative to the reference analysis). In Table 6.2 the corresponding figures after 24 hours are shown. The values are, in all cases, much smaller than those in Table 6.1, confirming the convergence between the forecasts. Thus we can see that the filtering succeeds in eliminating the noise without having a major impact on the forecast. However, the rms difference of 2.22 m between the reference forecast and that resulting from the analysis after three iterations of the filter, while not inordinately large, is indicative of some systematic change. To study this, the differences between the 24 hour forecast from filtered data and the twenty-five hour (yes, 25 h!) reference forecast were computed, and are tabulated in Table 6.3. Our suspicions are confirmed by comparing Tables 6.2 and 6.3: three iterations of the filter yield a forecast which differs by only 0.75 m (rms) from the 25 hour reference. The obvious deduction from this is that the output of the thrice-iterated filter is valid not at the analysis time but one hour later. The delay effect discussed in §6.1 is modified by iteration of the filter; the output after three iterations can no longer be assumed to refer to the initial time.

We conclude that the iterated filter is successful in eliminating HF noise, but produces output valid somewhat later than the data time. This is problematic if we wish to have balanced fields valid at the initial time. It is not an obstacle if our aim is to produce a smooth integration: the filter is applied and followed by a slightly shorter forecast (23 hours in the above example).

6.3.2 *Initialization by Incremental Filtering*

In all cases examined it has been found that to adequately eliminate the noise with a single application of the filter the span must exceed the delay. But the consequence of this is that the filter output refers to a time somewhat later than the analysis time. A means of circumventing this problem will now be described: instead of filtering the full fields, only the analysis increments need be filtered, since the preliminary fields (first guess fields) may be assumed to be in balance. This idea springs from the incremental nonlinear normal mode initialization technique developed by Ballish, *et al.*, 1992.

The Quick-Start filter was applied over a short range forecast as before, with the parameter values chosen as follows:

$$\Delta t = 360\,\mathrm{s} \qquad \tau_c = 6\,\mathrm{h} \qquad T_s = 2\,\mathrm{h}.$$

Note that the theoretical delay for these parameters is $\Delta_0 = 1.23\,\mathrm{h}$, which is less than the span. The filter coefficients take the following values:

$$a_0 = .00567 \quad a_1 = .01134 \quad a_2 = .00567 \quad b_1 = 1.69881 \quad b_2 = -0.72149$$

We first consider the effect of filtering the full fields. The evolution of the geopotential height at a central point, for the forecasts from the original analysis X_A (solid curve) and from the filtered data $X_A^\star$ (dashed curve) are shown in Fig. 6.6(a) (the dotted curve will be discussed presently). The HF oscillations are greatly reduced and, in particular, the initial tendency for the filtered forecast is realistic. Corresponding graphs of mean absolute divergence $\|\delta\|_1$ for the two forecasts are shown in Fig. 6.6(b). Clearly, the filtering of the initial data removes the bulk of the noise from the forecast. However, the rms difference of height relative to the reference 24-hour forecast was $\sigma_Z = 3.18\,\mathrm{m}$; rms and maximum differences of height and wind components for full-field initialization are shown in Table 6.4.

The difference in height between the 1-day forecasts from X_A and $X_A^\star$ is shown in Fig. 6.7(a); the maximum of 14 m is larger than would be expected from simple elimination of HF components and suggests that there is a time-shift resulting from the filtering. This is also to be expected as the span exceeds the delay ($T_s = 2\,\mathrm{h}$, $\Delta_0 = 1.23\,\mathrm{h}$). To reduce the phase-error due to filtering, we may apply the filter only to the analysis increment (cf. Ballish, *et al.*, 1992). Let X_F be the first-guess, typically a 6-hour forecast; then the analysis is $X_A = X_F + [X_A - X_F]$. Let $X_A^\star$ be the result of filtering the (full) analysis field; then the incremental filtering is defined as

$$X_A^{\star\star} = X_F + [X_A^\star - X_F^\star]. \tag{6.1}$$

This requires two short integrations, one from the analysis and one from the first-guess. The evolution of the geopotential height at a central point for the forecast from the incrementally filtered data $X_A^{\star\star}$ is shown in Fig. 6.6(a) (dotted curve). It is close to that of the fully filtered case (dashed curve). The noise profile $\|\delta\|_1$ for the forecast from $X_A^{\star\star}$, shown as a dotted line in Fig. 6.6(b), is almost identical to the result for the forecast from $X_A^\star$ (dashed line, Fig. 6.6(b)). The rms difference of height relative to the reference 24-hour forecast is now reduced to $\sigma_Z = 1.75\,\mathrm{m}$; rms and maximum differences of height and wind components for incremental initialization, shown in Table 6.4, may be seen to be conspicuously smaller than in the case of full-field filtering.

The difference in height between the forecasts from X_A and $X_A^{\star\star}$ is shown in Fig. 6.7(b); the maximum is now only 6 m. Thus, filtering of the analysis increment removes the noise without significant phase-error. Results with filter cutoff $\tau_c = 3\,\mathrm{h}$ and span $T_s = 1.5\,\mathrm{h}$ (not shown) were comparable to those reported here. The practical implementation of the incremental initialization technique requires two short-span integrations; however, the prelimianry field X_F is normally available prior to the analysis time, so that $X_F^\star$ may be calculated in advance, and only one short forecast, to compute $X_A^\star$, need be performed in real time.

7. SUMMARY

The elementary theory of digital filters has been reviewed. The two principal types of filter, nonrecursive and recursive, have been considered, and an application of each type to the problem of initialization has been discussed. There are a number of other potential uses for such filters: they may be applied periodically during an integration to suppress noise; they may be used to formulate balance constraints for four-dimensional data assimilation; they may be incorporated into the time-stepping in the manner in which the classical Robert-Asselin filter is used with the leapfrog scheme.

The nonrecursive (FIR) filters have no counterpart in the continuous domain. The most elementary design technique is based on Fourier theory; more sophisticated *optimal* filters are constructed by way of the Chebyshev Alternation Theorem. FIR filters have been successfully used for initialization (Lynch and Huang, 1992, Huang and Lynch, 1992). For this application a backward integration of the model is required in order to obtain output valid at the initial time. The effectiveness of a FIR filter is strongly dependent upon its span: the longer the span, the better the filtering.

The recursive (IIR) filters which we have considered are derived from classical analog filters by means of a bilinear transformation from the continuous (s) to the discrete (z) domain. Direct modelling in the digital domain is also possible. The effectiveness of an IIR filter depends on its frequency response, but also on its transient decay rate, or start-up time. It was argued that output from an IIR filter applied to a short forward integration may be valid at the initial time if the delay exceeds the span. However, to achieve adequate damping it was necessary to employ the filter iteratively, and the repeated application counteracted the delay to some extent. A way around this was found: if the filter is applied only to the analysis increment, it is successful in eliminating HF noise without causing a significant phase-error in the initialized analysis.

Recursive filters may be useful in climate modelling. Although such a filter can be employed continuously, it should be sufficient to apply it periodically during the integration. Some extra storage is required, but this should not be prohibitive. A filter of the Flat-Delay type may be more suitable for this application than the Quick-Start filter used for initialization. The filter may be applied to the full fields, and the time-lag allowed for by integrating past the fiducial time by an amount equal to the delay.

.

Acknowledgements

The work described in this Report was commenced whilst I was a visitor at the Danish Meteorological Institute during July, 1992. I should like to thank the Staff of the Research Division at DMI for their kindness and hospitality during my visit.

It is also a pleasure to thank Xiang-Yu Huang of DMI and Peter Lönnberg of FMI who read the manuscript, pointed out some errors and made a number of suggestions for improvements.

Fig. 2.1. Cassinian ovals: these curves are the loci of points for which the product of the distances r_1 and r_2 from two fixed points has a constant value a^2. Here the fixed points are $(1,0)$ and $(-1,0)$. The graphs shown are for $a^2 = 1, \sqrt{2}, 2$ and 3. The curves were introduced in 1680 by Giovanni Domenico Cassini (1625–1712) in connection with a study of the relative motions of the earth and sun. The case $a^2 = 1$ is called the lemniscate (L. *lemniscus,* ribbon), the figure-of-eight curve first investigated by Jacques Bernoulli in 1694, which played a major rôle in eighteenth century analysis. For $a^2 < 1$ the curves have two separate components. The value $a^2 = \sqrt{2}$ occurs in connection with prototype filters. This curve is also characterised as the locus of points for which the product of the distances r_1 and r_2 from $s = i$ and $s = -i$ is equal to $2r^2$ where r is the distance from the origin. Look out for these Cassinians next time you slice a doughnut.

Fig. 2.2. This peanut-shaped curve is the locus in the s-plane of the poles $s_p = -\sigma_0 \pm \omega_0$ of prototype second-order filters (with cutoff frequency $\omega_c = 1$). The three special cases considered in the text are indicated (BW, FD, QS). The curve is described by the equation $(\sigma_0^2 + \omega_0^2)^2 = 2(\sigma_0^2 - \omega_0^2) + 1$. The unit circle is shown for reference.

Fig. 2.3. Amplitude (top) and Group Delay (bottom) of the three special prototype filters considered in the text (BW: Butterworth, FD: Flat-Delay, QS: Quick-Start).

Fig. 4.1. Transfer functions for the filter defined by the coefficients $h_n = \sin(n\omega_c\Delta t)/n\pi$, where the cutoff period $\tau_c = 2\pi/\omega_c$ is 6 h, with and without modification by a Lanczos window.

Fig. 4.2. Transfer functions for the filter defined as follows: (solid) $h_n = \sin(n\omega_c\Delta t)/n\pi$ with Lanczos window, cutoff $\tau_c = 2\pi/\omega_c$ of 6 h and span $T_s = 6$h; (dashed) same, with span $T_s = 3$h; (dotted) optimal filter with span $T_s = 3$h, $(\tau_p, \tau_s) = (12\,\mathrm{h}, 3\,\mathrm{h})$; (dot-dashed) same, with span $T_s = 2$h, $(\tau_p, \tau_s) = (15\,\mathrm{h}, 3\,\mathrm{h})$.

Fig. 4.3. Mean absolute surface pressure tendency N_1 (units hPa/3h) for a 24 h forecast starting from uninitialized data (solid curve), from digitally filtered fields (dashed curve) and from data after normal mode initialization (dotted curve).

Fig. 4.4. Time evolution of the mean absolute surface pressure tendency N_1 for analyses diabatically initialized by four different digital filters (see text for details) and for adiabatic normal mode initialization

Fig. 5.1. Frequency response functions (magnitude of transfer functions) for two second-order recursive filters derived from the Butterworth analog filter. (A) Digitization using the bi-linear transformation; (B) Digitization using the matched $\mathcal{Z}$-transform.

Fig. 5.2. Amplitude (top) and Group Delay (bottom) of three digital filters derived using the bilinear transformation (BW: Butterworth, FD: Flat-Delay, QS: Quick-Start). Timestep $\Delta t = 360s$, cutoff period $\tau_c = 12h$, cutoff frequency $\theta_c = \pi/60 = 0.05236$, $f_c = 1/120 = 0.008333$.

Fig. 5.3. Amplitude (top) and phase (bottom) of response of two digital QS filters derived using the bilinear transformation with timestep $\Delta t = 360s$ and cutoff period $\tau_c = 12h$. Solid line: zero at $\tau_0 = 2\Delta t$ or $\theta_0 = \pi$; dashed line: zero at $\tau_0 = 2\,\mathrm{h} = 20\Delta t$ or $\theta_0 = \pi/10$ or $f_0 = 1/20$.

Fig. 6.1. Input $\mathbf{X}$ and output $\mathbf{Y}$ for a second-order Butterworth filter. Input is an oscillation with period 24 hours. Output has the same period and (approximately) the same amplitude, but is phase-shifted. Top panel: $\tau_c = 6\,\mathrm{h}$, $\delta_0 = 1.35\,\mathrm{h}$; bottom panel: $\tau_c = 12\,\mathrm{h}$, $\delta_0 = 2.70\,\mathrm{h}$.

Fig. 6.2. Solid Line: Root-mean-square difference (metres) between forecasts of various lengths and the initial state. Dashed Line: rms difference between filter output for various spans and the initial state. Top panel for filter with cutoff period $\tau_c = 6\,\mathrm{h}$ and delay $\delta = 1.228\,\mathrm{h}$; bottom panel for $\tau_c = 12\,\mathrm{h}$ and $\delta = 2.458\,\mathrm{h}$.

Fig. 6.3. Solid Line: Input signal $x_n = \cos(2\pi n\Delta t/\tau_1)$. Output y_n for each of four iterations of filter. Top panel: Quick-Start Filter; bottom panel: Butterworth Filter. For further details, see text.

Fig. 6.4. (a) Uninitialized 500 hPa geopotential analysis for 00Z, 22-11-1982. (b) 24 hour reference forecast of geopotential.

Fig. 6.5. (Top panel) Evolution of geopotential at a central point during the first six forecast hours. Solid: reference forecast; dashed: 1 iteration; dotted: 2 iterations;

dot-dashed: 3 iterations of filter. (Bottom panel) Similar curves for mean absolute divergence.

Fig. 6.6. (a) Evolution of the geopotential height at a central point for the forecasts from the original analysis X_A (solid curve), from the fully filtered data $X_A^\star$ (dashed curve) and from the incrementally filtered data $X_A^{\star\star}$ (dotted curve). (b) Corresponding graphs of mean absolute divergence $\|\delta\|_1$ for the three forecasts.

Fig. 6.7. (a) Difference in height between the 1-day forecasts from X_A and $X_A^\star$ (forecast impact of full-field initialization). (b) Difference in height between the forecasts from X_A and $X_A^{\star\star}$ (forecast impact of incremental initialization).

Table 6.1 The rms and maximum changes in height and winds after each iteration (all figures relative to the reference analysis).

Table 6.2 The rms and maximum differences in height and winds after each iteration (all figures relative to the reference analysis) for the 24 hour forecasts.

Table 6.3 The rms and maximum differences in height and winds after each iteration (all figures relative to the reference analysis) between the 24 hour filtered forecasts and the 25 hour reference.

DSP Books

Bose, N K., 1985: *Digital Filters. Theory and Applications.* North-Holland, 496pp.

Hamming, R.W., 1989: *Digital Filters.* Prentice-Hall International, 284pp.

Higgins, R.J., 1990: *Digital Signal Processing in VLSI.* Prentice-Hall, 575pp.

INMOS Limited, 1989: *Digital Signal Processing.* Prentice-Hall, 266pp.

Kuo, F.F., 1966: *Network Analysis and Synthesis.* John Wiley & Sons, 515pp.

O'Flynn, M. and E.Moriarty, 1987: *Linear Systems: Time Domain and Transform Analysis.* John Wiley & Sons, 500pp.

Oppenheim, A.V., A.S. Willsky and I.T. Young, 1983: *Signals and Systems.* Prentice-Hall Inc., 796pp.

Oppenheim, A.V. and R.W. Schafer, 1989: *Discrete-Time Signal Processing.* Prentice-Hall Intl., Inc., 879pp.

Parks, T.W. and C.S. Burrus, 1987: *Digital Filter Design.* John Wiley & Sons, Inc., 342pp.

Proakis, J.G. and D.G. Manolakis, 1988: *Introduction to Digital Signal Processing.* Macmillan Publ. Co., 944pp.

Rabiner, L.R. and B. Gold, 1975: *Theory and Application of Digital Signal Processing.* Prentice-Hall, Inc., 762pp.

Strum, R.D. and D.E. Kirk, 1988: *Discrete Systems and Digital Signal Processing.* Addison-Wesley, 848pp.

Taylor, F.J. and T. Stouraitis, 1987: *Digital Filter Design Software for the IBM PC.* Marcel Dekker, Inc., 299 pp.

Williams, C.S., 1986: *Designing Digital Filters.* Prentice-Hall, Inc., 349pp.

Williams, A.B. and F.J. Taylor, 1986: *Electronic Filter Design Handbook.* McGraw–Hill Publ. Co., 554pp.

General References

Asselin, R., 1972: Frequency filter for time integrations. *Mon. Weather Rev.,* **100**, 487–490.

Ballish, B., X. Cao, E. Kalnay and M Kanamitsu, 1992: Incremental nonlinear normal-mode initialization. *Mon. Weather Rev.,* **120**, 1723–1734.

Daley, R., 1991: *Atmospheric Data Analysis.* Cambridge, 457pp.

Doetsch, G, 1971: *Guide to the Applications of the Laplace and Z-Transforms.* Van Nostrand Reinhold, 240pp.

Huang, Xiang-Yu and Peter Lynch, 1992: Diabatic digital filtering initialization: application to the HIRLAM model. *Mon. Weather Rev.,* **120**, (December issue).

Lorenc, A., 1992: Iterative analysis using covariance functions and filters. *Q. J. Roy. Met. Soc.*, **118**, 569–591.

Lynch, Peter, 1991a: Filtering integration schemes based on the Laplace and Z transforms. *Mon. Weather Rev.*, **119**, 653–666.

Lynch, Peter, 1991b: Filtered Equations and Filtering Integration Schemes. Proceedings of Seminar on Numerical Methods in Atmospheric Models, ECMWF Reading, U.K., pp 119–159.

Lynch, Peter, 1992: Richardson's barotropic forecast: a reappraisal. *Bull. Amer. Meteor. Soc.,* **73**, 1, 35–47.

Lynch, Peter, and Xiang-Yu Huang, 1992: Initialization of the HIRLAM model using a digital filter. *Mon. Weather Rev.,* **120**, 1019–1034.

McClellan, J.H., T.W. Parks and L.R. Rabiner, 1973: A computer program for designing optimum FIR linear-phase digital filters. *IEEE Trans. on Audio and Electroacoustics,* **AU-21**, 506–526.

Machenhauer, B., 1977: On the dynamics of gravity oscillations in a shallow water model with applications to normal mode initialization. *Beitr. Atmos. Phys.*, **50**, 253–271.

Raymond, W.H. and A. Garder, 1991: A review of recursive and implicit filters. *Mon. Weather Rev.*, **119**, 477–495.