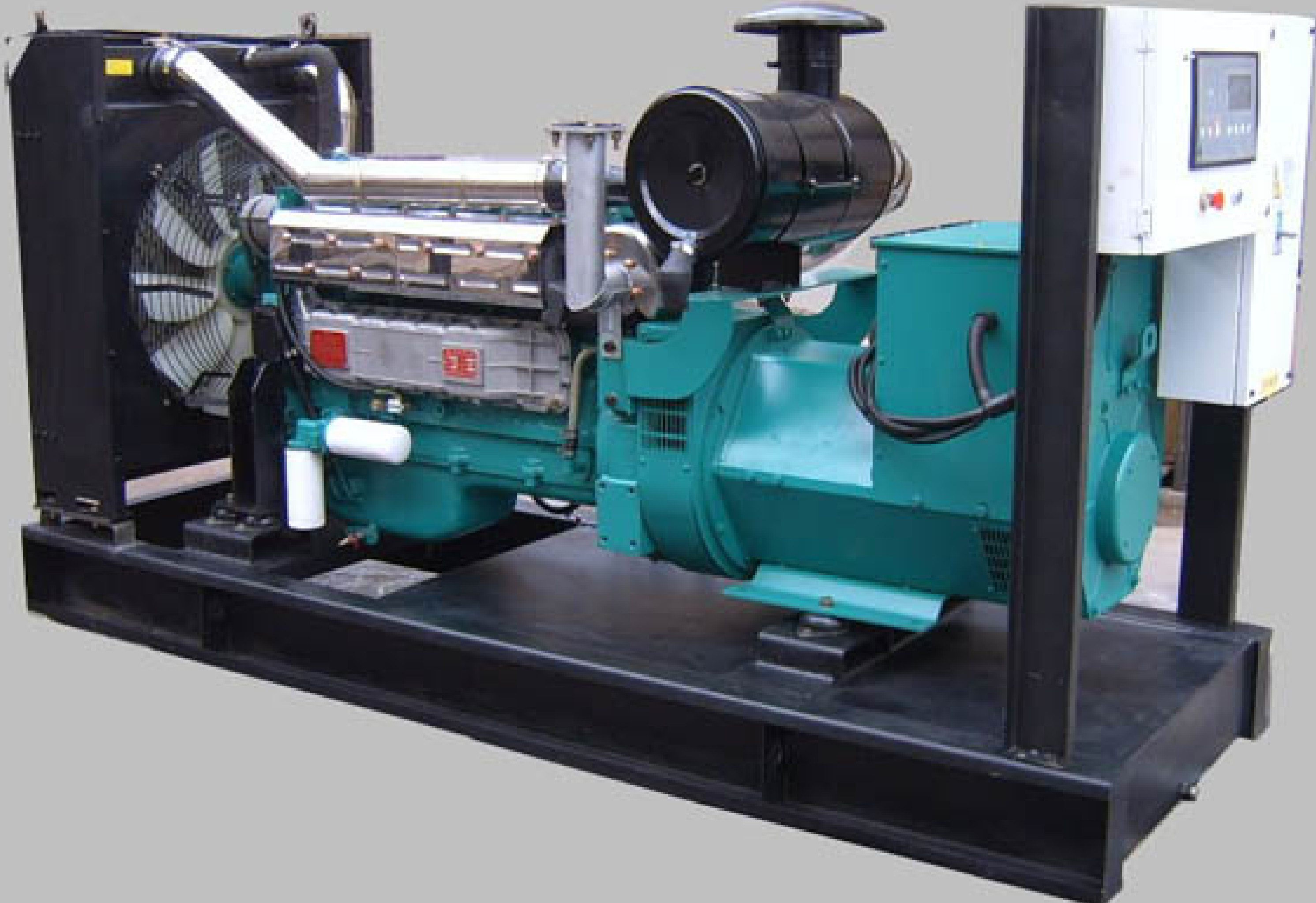
Handbook of Signal Processing Filters



Elaine Lashley

First Edition, 2012

ISBN 978-81-323-3585-6

Published by:
University Publications
4735/22 Prakashdeep Bldg,
Ansari Road, Darya Ganj,
Delhi - 110002
Email: info@wtbooks.com

Table of Contents

Chapter- 1

Filter (Signal Processing)

In signal processing, a **filter** is a device or process that removes from a signal some unwanted component or feature. Filtering is a class of signal processing, the defining feature of filters being the complete or partial suppression of some aspect of the signal. Most often, this means removing some frequencies and not others in order to suppress interfering signals and reduce background noise. However, filters do not exclusively act in the frequency domain; especially in the field of image processing many other targets for filtering exist.

There are many different bases of classifying filters and these overlap in many different ways; there is no simple hierarchical classification. Filters may be:

- analog or digital
- discrete-time (sampled) or continuous-time
- linear or non-linear
- passive or active type of continuous-time filter
- infinite impulse response (IIR) or finite impulse response (FIR) type of discrete-time or digital filter.

Linear continuous-time filters

Linear continuous-time circuit is perhaps the most common meaning for filter in the signal processing world, and simply "filter" is often taken to be synonymous. These are filters that are designed to remove certain frequencies and allow others to pass. Such a filter is, of necessity, a linear filter. Any non-linearity will result in the output signal containing components of frequency which were not present in the input signal.

The modern design methodology for linear continuous-time filters is called network synthesis. Some important filter families designed in this way are:

- Chebyshev filter, has the best approximation to the ideal response of any filter for a specified order and ripple.
- Butterworth filter, has a maximally flat frequency response.
- Bessel filter, has a maximally flat phase delay.
- Elliptic filter, has the steepest cutoff of any filter for a specified order and ripple.

The difference between these filter families is that they all use a different polynomial function to approximate to the ideal filter response. This results in each having a different transfer function.

Another older, less-used methodology is the image parameter method. Filters designed by this methodology are archaically called "wave filters". Some important filters designed by this method are:

- Constant k filter, the original and simplest form of wave filter.
- m-derived filter, a modification of the constant k with improved cutoff steepness and impedance matching.

Terminology

Some terms used to describe and classify linear filters:

- The frequency response can be classified into a number of different bandforms describing which frequencies the filter passes (the passband) and which it rejects (the stopband):
    - Low-pass filter – low frequencies are passed, high frequencies are attenuated.
    - High-pass filter – high frequencies are passed, low frequencies are attenuated.
    - Band-pass filter – only frequencies in a frequency band are passed.
    - Band-stop filter or band-reject filter – only frequencies in a frequency band are attenuated.
    - Notch filter – rejects just one specific frequency - an extreme band-stop filter.
    - Comb filter – has multiple regularly spaced narrow passbands giving the bandform the appearance of a comb.
    - All-pass filter – all frequencies are passed, but the phase of the output is modified.

- Cutoff frequency is the frequency beyond which the filter will not pass signals. It is usually measured at a specific attenuation such as 3dB.
- Roll-off is the rate at which attenuation increases beyond the cut-off frequency.
- Transition band, the (usually narrow) band of frequencies between a passband and stopband.
- Ripple is the variation of the filters insertion loss in the passband.

- The order of a filter is the degree of the approximating polynomial and in passive filters corresponds to the number of elements required to build it. Increasing order increases roll-off and brings the filter closer to the ideal response.

Technologies

Filters can be built in a number of different technologies. The same transfer function can be realised in several different ways, that is the mathematical properties of the filter are the same but the physical properties are quite different. Often the components in different technologies are directly analogous to each other and fulfill the same role in their respective filters. For instance, the resistors, inductors and capacitors of electronics correspond respectively to dampers, masses and springs in mechanics. Likewise, there are corresponding components in distributed element filters.

- Electronic filters were originally entirely passive consisting of resistance, inductance and capacitance. Active technology makes design easier and opens up new possibilities in filter specifications.
- Digital filters operate on signals represented in digital form. The essence of a digital filter is that it directly implements a mathematical algorithm, corresponding to the desired filter transfer function, in its programming or microcode.
- Mechanical filters are built out of mechanical components. In the vast majority of cases they are used to process an electronic signal and transducers are provided to convert this to and from a mechanical vibration. However, examples do exist of filters that have been designed for operation entirely in the mechanical domain.
- Distributed element filters are constructed out of components made from small pieces of transmission line or other distributed elements. There are structures in distributed element filters that directly correspond to the lumped elements of electronic filters, and others that are unique to this class of technology.
- Waveguide filters consist of waveguide components or components inserted in the waveguide. Waveguides are a class of transmission line and many structures of distributed element filters, for instance the stub (electronics), can be implemented in waveguides also.
- Acoustic filters
- Optical filters were originally developed for purposes other than signal processing such as lighting and photography. With the rise of optical fiber technology, however, optical filters increasingly find signal processing applications and signal processing filter terminology, such as longpass and shortpass, are entering the field.

The transfer function

The transfer function $H(s)$ of a filter is the ratio of the output signal $Y(s)$ to that of the input signal $X(s)$ as a function of the complex frequency s:

$$H(s) = \frac{Y(s)}{X(s)}$$

with $s = \sigma + j\omega$.

The transfer function of all linear time-invariant filters generally share certain characteristics:

- For filters which are constructed of discrete components, their transfer function must be the ratio of two polynomials in s, i.e. a rational function of s. The order of the transfer function will be the highest power of $s$encountered in either the numerator or the denominator.
- The polynomials of the transfer function will all have real coefficients. Therefore, the poles and zeroes of the transfer function will either be real or occur in complex conjugate pairs.
- Since the filters are assumed to be stable, the real part of all poles (i.e. zeroes of the denominator) will be negative, i.e. they will lie in the left half-plane in complex frequency space.

Distributed element filters do not, in general, produce rational functions but can often approximate to them.

The proper construction of a transfer function involves the Laplace transform, and therefore it is needed to assume null initial conditions, because

$$\mathcal{L}\left\{\frac{df}{dt}\right\} = s \cdot \mathcal{L}\left\{f(t)\right\} - f(0),$$

And when f(0)=0 we can get rid of the constants and use the usual expression

$$\mathcal{L}\left\{\frac{df}{dt}\right\} = s \cdot \mathcal{L}\left\{f(t)\right\}$$

An alternative to transfer functions is to give the behavior of the filter as a convolution. The convolution theorem, which holds for Laplace transforms, guarantees equivalence with transfer functions.

Classification

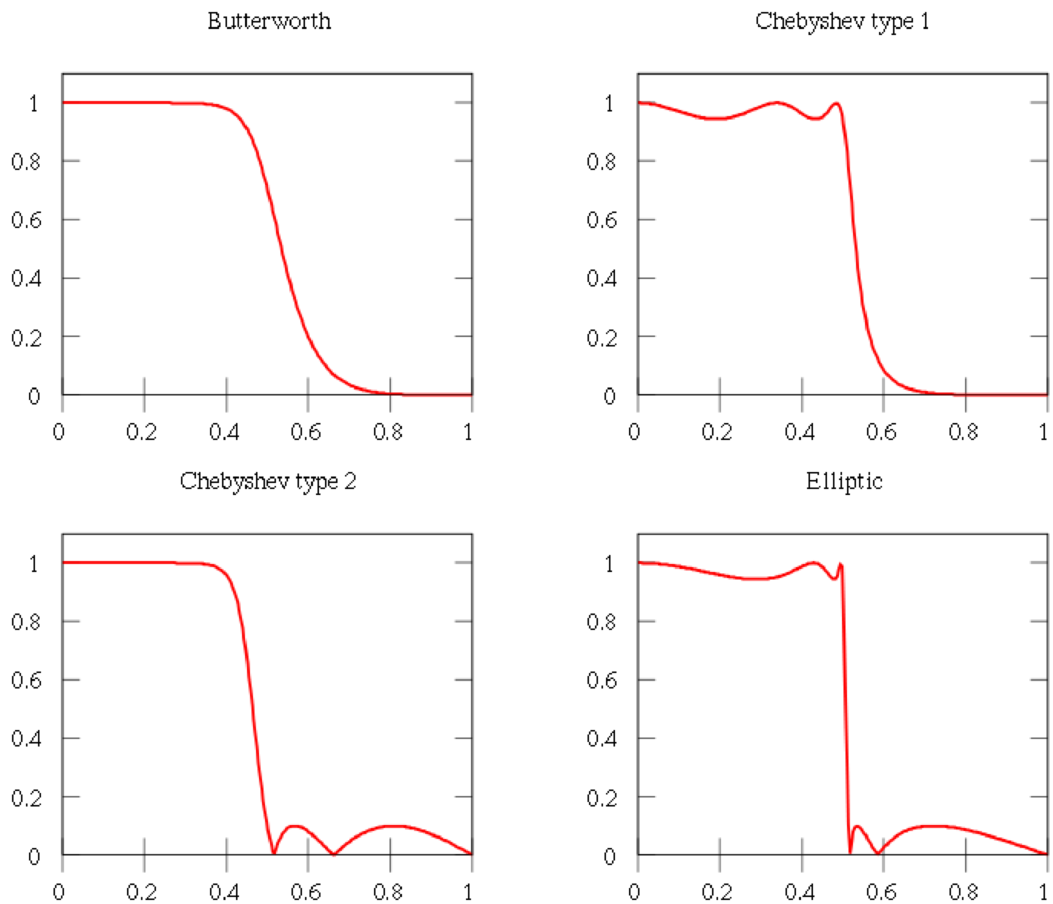
Filters may be specified by family and bandform. A filter's family is specified by the approximating polynomial used and each leads to certain characteristics of the transfer function of the filter. Some common filter families and their particular characteristics are:

- Butterworth filter – no gain ripple in pass band and stop band, slow cutoff

- Chebyshev filter (Type I) – no gain ripple in stop band, moderate cutoff
- Chebyshev filter (Type II) – no gain ripple in pass band, moderate cutoff
- Bessel filter – no group delay ripple, no gain ripple in both bands, slow gain cutoff
- Elliptic filter – gain ripple in pass and stop band, fast cutoff
- Optimum "L" filter
- Gaussian filter – no ripple in response to step function
- Hourglass filter
- Raised-cosine filter

Each family of filters can be specified to a particular order. The higher the order, the more the filter will approach the "ideal" filter; but also the longer the impulse response is and the longer the latency will be. An ideal filter has full transmission in the pass band, complete attenuation in the stop band, and an abrupt transition between the two bands, but this filter has infinite order (i.e., the response cannot be expressed as a linear differential equation with a finite sum) and infinite latency (i.e., its compact support in the Fourier transform forces its time response to be ever lasting).

Here is an image comparing Butterworth, Chebyshev, and elliptic filters. The filters in this illustration are all fifth-order low-pass filters. The particular implementation – analog or digital, passive or active – makes no difference; their output would be the same.



As is clear from the image, elliptic filters are sharper than all the others, but they show ripples on the whole bandwidth.

Any family can be used to implement a particular bandform of which frequencies are transmitted, and which, outside the passband, are more or less attenuated. The transfer function completely specifies the behavior of a linear filter, but not the particular technology used to implement it. In other words, there are a number of different ways of achieving a particular transfer function when designing a circuit. A particular bandform of filter can be obtained by transformation of a prototype filter of that family.

Impedance matching

Impedance matching structures invariably take on the form of a filter, that is, a network of non-dissipative elements. For instance, in a passive electronics implementation, this would likely take the form of a ladder topology of inductors and capacitors. The design of matching networks shares much in common with filters and the design invariably will have a filtering action as an incindental consequence. Although the prime purpose of a matching network is not to filter, it is often the case that both functions are combined in the same circuit. The need for impedance matching does not arise while signals are in the digital domain.

Chapter- 2

Passive Analogue Filter Development

Analogue filters are a basic building block of signal processing much used in electronics. Amongst their many applications are the separation of an audio signal before application to bass, mid-range and tweeter loudspeakers; the combining and later separation of multiple telephone conversations onto a single channel; the selection of a chosen radio station in a radio receiver and rejection of others.

Passive linear electronic analogue filters are those filters which can be described with linear differential equations (linear); they are composed of capacitors, inductors and, sometimes, resistors (passive) and are designed to operate on continuously varying (analogue) signals. There are many linear filters which are not analogue in implementation (digital filter), and there are many electronic filters which may not have a passive topology – both of which may have the same transfer function of the filters described here. Analogue filters are most often used in wave filtering applications, that is, where it is required to pass particular frequency components and to reject others from analogue (continuous-time) signals.

Analogue filters have played an important part in the development of electronics. Especially in the field of telecommunications, filters have been of crucial importance in a number of technological breakthroughs and have been the source of enormous profits for telecommunications companies. It should come as no surprise, therefore, that the early development of filters was intimately connected with transmission lines. Transmission line theory gave rise to filter theory, which initially took a very similar form, and the main application of filters was for use on telecommunication transmission lines. However, the arrival of network synthesis techniques greatly enhanced the degree of control of the designer.

Today, it is often preferred to carry out filtering in the digital domain where complex algorithms are much easier to implement, but analogue filters do still find applications, especially for low-order simple filtering tasks and are often still the norm at higher frequencies where digital technology is still impractical, or at least, less cost effective. Wherever possible, and especially at low frequencies, analogue filters are now implemented in a filter topology which is active in order to avoid the wound components required by passive topology.

It is possible to design linear analogue mechanical filters using mechanical components which filter mechanical vibrations or acoustic waves. While there are few applications for such devices in mechanics per se, they can be used in electronics with the addition of transducers to convert to and from the electrical domain. Indeed some of the earliest ideas for filters were acoustic resonators because the electronics technology was poorly understood at the time. In principle, the design of such filters can be achieved entirely in terms of the electronic counterparts of mechanical quantities, with kinetic energy, potential energy and heat energy corresponding to the energy in inductors, capacitors and resistors respectively.

Historical overview

There are three main stages in the history of **passive analogue filter development**:

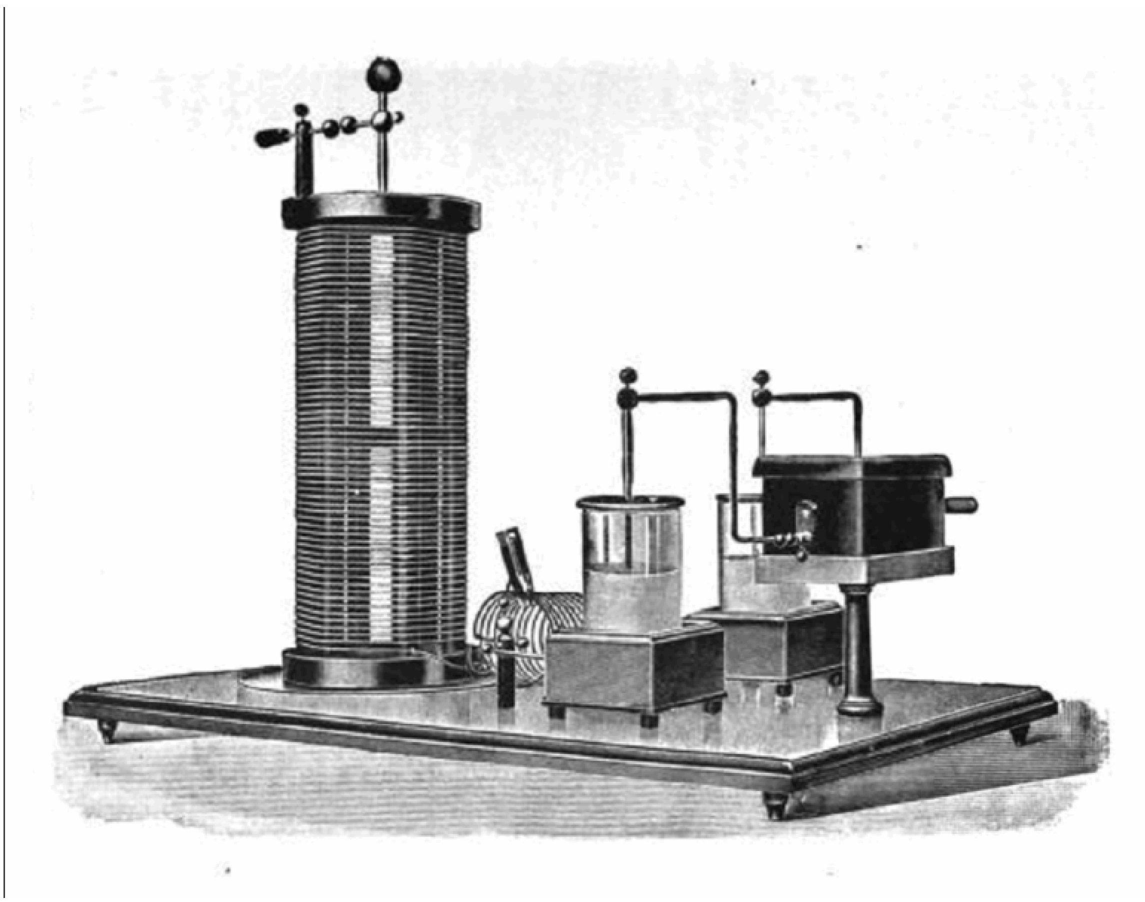
1. **Simple filters**. The frequency dependence of electrical response was known for capacitors and inductors from very early on. The resonance phenomenon was also familiar from an early date and it was possible to produce simple, single-branch filters with these components. Although attempts were made in the 1880s to apply them to telegraphy, these designs proved inadequate for successful frequency division multiplexing. Network analysis was not yet powerful enough to provide the theory for more complex filters and progress was further hampered by a general failure to understand the frequency domain nature of signals.
2. **Image filters**. Image filter theory grew out of transmission line theory and the design proceeded in a similar manner to transmission line analysis. For the first time filters could be produced that had precisely controllable passbands and other parameters. These developments took place in the 1920s and filters produced to these designs were still in widespread use in the 1980s, only declining as the use of analogue telecommunications has declined. Their immediate application was the economically important development of frequency division multiplexing for use on intercity and international telephony lines.
3. **Network synthesis filters**. The mathematical bases of network synthesis were laid in the 1930s and 1940s. After the end of World War II network synthesis became the primary tool of filter design. Network synthesis put filter design on a firm mathematical foundation, freeing it from the mathematically sloppy techniques of image design and severing the connection with physical lines. The essence of network synthesis is that it produces a design that will (at least if implemented with ideal components) accurately reproduce the response originally specified in black box terms.

Throughout the letters R,L and C are used here with their usual meanings to represent resistance, inductance and capacitance, respectively. In particular they are used in combinations, such as LC, to mean, for instance, a network consisting only of inductors and capacitors. Z is used for electrical impedance, any 2-terminal combination of RLC elements and in some sections D is used for the rarely seen quantity elastance, which is the inverse of capacitance.

Resonance

Early filters utilised the phenomenon of resonance to filter signals. Although electrical resonance had been investigated by researchers from a very early stage, it was at first not widely understood by electrical engineers. Consequently, the much more familiar concept of acoustic resonance (which in turn, can be explained in terms of the even more familiar mechanical resonance) found its way into filter design ahead of electrical resonance. Resonance can be used to achieve a filtering effect because the resonant device will respond to frequencies at, or near, to the resonant frequency but will not respond to frequencies far from resonance. Hence frequencies far from resonance are filtered out from the output of the device.

Electrical resonance



A 1915 example of an early type of resonant circuit known as an Oudin coil which uses Leyden jars for the capacitance.

Resonance was noticed early on in experiments with the Leyden jar, invented in 1746. The Leyden jar stores electricity due to its capacitance, and is, in fact, an early form of capacitor. When a Leyden jar is discharged by allowing a spark to jump between the electrodes, the discharge is oscillatory. This was not suspected until 1826, when Felix Savary in France, and later (1842) Joseph Henry in the US noted that a steel needle placed close to the discharge does not always magnetise in the same direction. They both independently drew the conclusion that there was a transient oscillation dying with time.

Hermann von Helmholtz in 1847 published his important work on conservation of energy in part of which he used those principles to explain why the oscillation dies away, that it is the resistance of the circuit which dissipates the energy of the oscillation on each successive cycle. Helmholtz also noted that there was evidence of oscillation from the electrolysis experiments of William Hyde Wollaston. Wollaston was attempting to decompose water by electric shock but found that both hydrogen and oxygen were present at both electrodes. In normal electrolysis they would separate, one to each electrode.

Helmholtz explained why the oscillation decayed but he had not explained why it occurred in the first place. This was left to Sir William Thomson (Lord Kelvin) who, in 1853, postulated that there was inductance present in the circuit as well as the capacitance of the jar and the resistance of the load. This established the physical basis for the phenomenon - the energy supplied by the jar was partly dissipated in the load but also partly stored in the magnetic field of the inductor.

So far, the investigation had been on the natural frequency of transient oscillation of a resonant circuit resulting from a sudden stimulus. More important from the point of view of filter theory is the behaviour of a resonant circuit when driven by an external AC signal: there is a sudden peak in the circuits response when the driving signal frequency is at the resonant frequency of the circuit. James Clerk Maxwell heard of the phenomenon from Sir William Grove in 1868 in connection with experiments on dynamos, and was also aware of the earlier work of Henry Wilde in 1866. Maxwell explained resonance mathematically, with a set of differential equations, in much the same terms that an RLC circuit is described today.

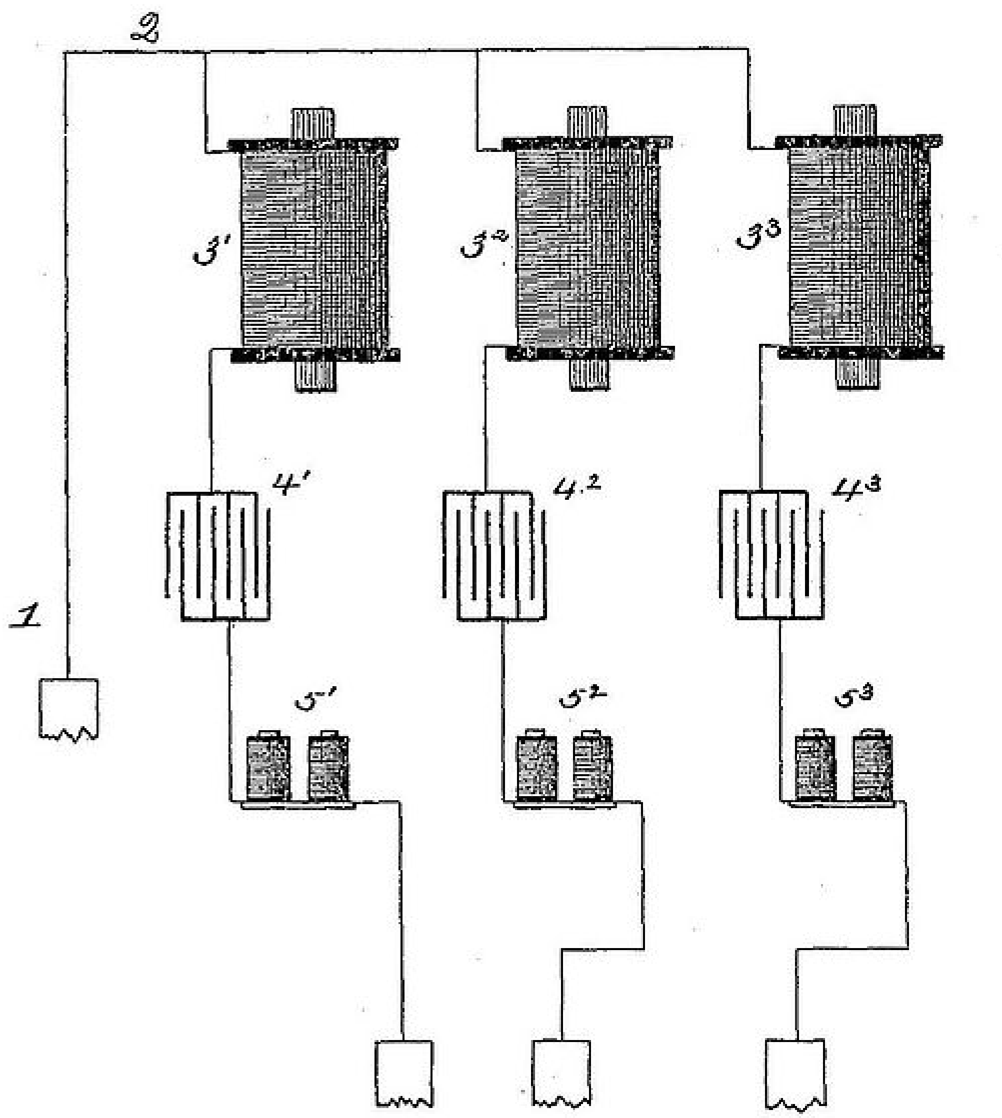
Heinrich Hertz (1887) experimentally demonstrated the resonance phenomena by building two resonant circuits, one of which was driven by a generator and the other was tunable and only coupled to the first electromagnetically (i.e., no circuit connection). Hertz showed that the response of the second circuit was at a maximum when it was in tune with the first. The diagrams produced by Hertz in this paper were the first published plots of an electrical resonant response.

Acoustic resonance

As mentioned earlier, it was acoustic resonance that inspired filtering applications, the first of these being a telegraph system known as the "harmonic telegraph". Versions are due to Elisha Gray, Alexander Graham Bell (1870s), Ernest Mercadier and others. Its purpose was to simultaneously transmit a number of telegraph messages over the same line and represents an early form of frequency division multiplexing (FDM). FDM requires the sending end to be transmitting at different frequencies for each individual communication channel. This demands individual tuned resonators, as well as filters to separate out the signals at the receiving end. The harmonic telegraph achieved this with electromagnetically driven tuned reeds at the transmitting end which would vibrate similar reeds at the receiving end. Only the reed with the same resonant frequency as the transmitter would vibrate to any appreciable extent at the receiving end.

Incidentally, the harmonic telegraph directly suggested to Bell the idea of the telephone. The reeds can be viewed as transducers converting sound to and from an electrical signal. It is no great leap from this view of the harmonic telegraph to the idea that speech can be converted to and from an electrical signal.

Early multiplexing



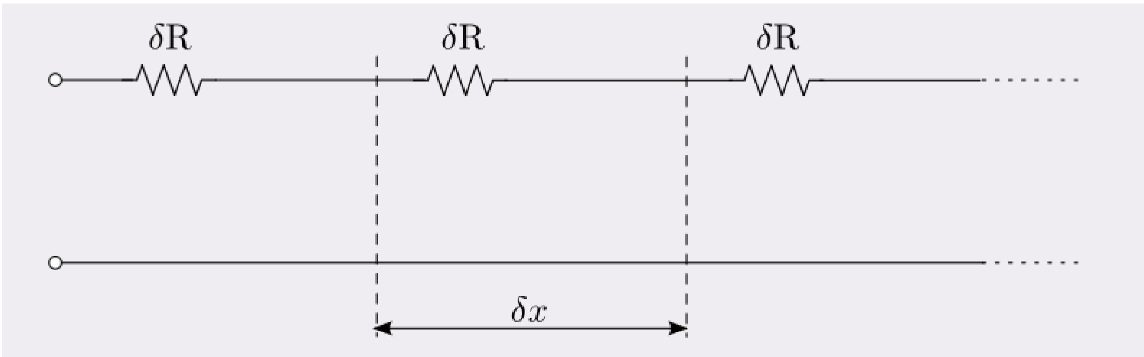
Hutin and Leblanc's multiple telegraph filter of 1891 showing the use of resonant circuits in filtering.

By the 1890s electrical resonance was much more widely understood and had become a normal part of the engineer's toolkit. In 1891 Hutin and Leblanc patented an FDM scheme for telephone circuits using resonant circuit filters. Rival patents were filed in 1892 by Michael Pupin and John Stone Stone with similar ideas, priority eventually being awarded to Pupin. However, no scheme using just simple resonant circuit filters can successfully multiplex (i.e. combine) the wider bandwidth of telephone channels (as opposed to telegraph) without either an unacceptable restriction of speech bandwidth or a channel spacing so wide as to make the benefits of multiplexing uneconomic.

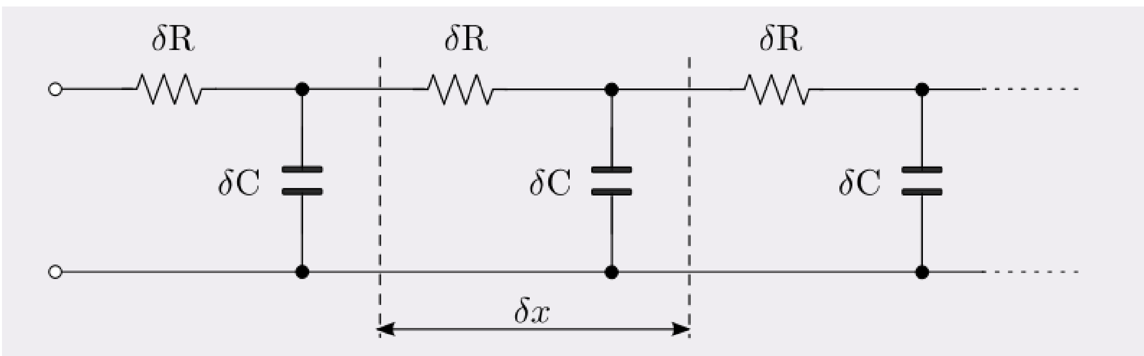
The basic technical reason for this difficulty is that the frequency response of a simple filter approaches a fall of 6 dB/octave far from the point of resonance. This means that if telephone channels are squeezed in side-by-side into the frequency spectrum, there will be crosstalk from adjacent channels in any given channel. What is required is a much more sophisticated filter that has a flat frequency response in the required passband like a low-Q resonant circuit, but that rapidly falls in response (much faster than 6 dB/octave) at the transition from passband to stopband like a high-Q resonant circuit. Obviously, these are contradictory requirements to be met with a single resonant circuit. The solution to these needs was founded in the theory of transmission lines and consequently the necessary filters did not become available until this theory was fully developed. At this early stage the idea of signal bandwidth, and hence the need for filters to match to it, was not fully understood; indeed, it was as late as 1920 before the concept of bandwidth was fully established. For early radio, the concepts of Q-factor, selectivity and tuning sufficed. This was all to change with the developing theory of transmission lines on which image filters are based, as explained in the next section.

At the turn of the century as telephone lines became available, it became popular to add telegraph on to telephone lines with an earth return phantom circuit. An LC filter was required to prevent telegraph clicks being heard on the telephone line. From the 1920s onwards, telephone lines, or balanced lines dedicated to the purpose, were used for FDM telegraph at audio frequencies. The first of these systems in the UK was a Siemens and Halske installation between London and Manchester. GEC and AT&T also had FDM systems. Separate pairs were used for the send and receive signals. The Siemens and GEC systems had six channels of telegraph in each direction, the AT&T system had twelve. All of these systems used electronic oscillators to generate a different carrier for each telegraph signal and required a bank of band-pass filters to separate out the multiplexed signal at the receiving end.

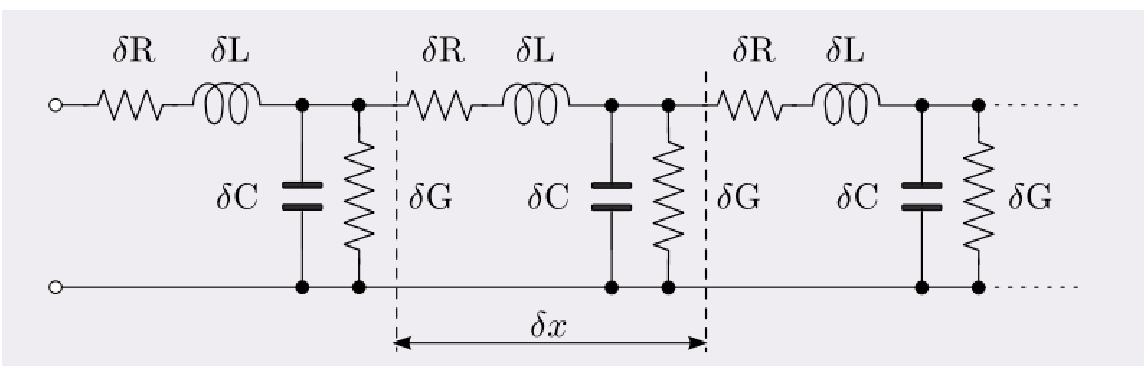
Transmission line theory



Ohm's model of the transmission line was simply resistance.



Lord Kelvin's model of the transmission line accounted for capacitance and the dispersion it caused. The diagram represents Kelvin's model translated into modern terms using infinitesimal elements, but this was not the actual approach used by Kelvin.

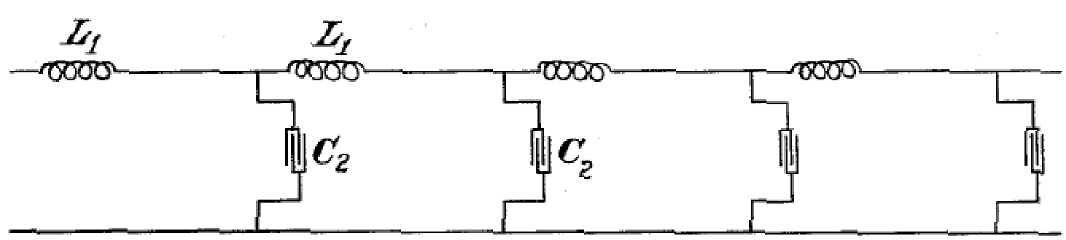


Heaviside's model of the transmission line. L, R, C and G in all three diagrams are the primary line constants. The infinitesimals δL, δR, δC and δG are to be understood as Lδx, Rδx, Cδx and Gδx respectively.

The earliest model of the transmission line was probably described by Georg Ohm (1827) who established that resistance in a wire is proportional to its length. The Ohm model thus included only resistance. Latimer Clark noted that signals were delayed and elongated along a cable, an undesirable form of distortion now called dispersion but then called retardation, and Michael Faraday (1853) established that this was due to the capacitance present in the transmission line. Lord Kelvin (1854) found the correct mathematical description needed in his work on early transatlantic cables; he arrived at an equation identical to the conduction of a heat pulse along a metal bar. This model incorporates only resistance and capacitance, but that is all that was needed in undersea cables dominated by capacitance effects. Kelvin's model predicts a limit on the telegraph signalling speed of a cable but Kelvin still did not use the concept of bandwidth, the limit was entirely explained in terms of the dispersion of the telegraph symbols. The mathematical model of the transmission line reached its fullest development with Oliver Heaviside. Heaviside (1881) introduced series inductance and shunt conductance into the model making four distributed elements in all. This model is now known as the telegrapher's equation and the distributed elements are called the primary line constants.

From the work of Heaviside (1887) it had become clear that the performance of telegraph lines, and most especially telephone lines, could be improved by the addition of inductance to the line. George Campbell at AT&T implemented this idea (1899) by inserting loading coils at intervals along the line. Campbell found that as well as the desired improvements to the line's characteristics in the passband there was also a definite frequency beyond which signals could not be passed without great attenuation. This was a result of the loading coils and the line capacitance forming a low-pass filter, an effect that is only apparent on lines incorporating lumped components such as the loading coils. This naturally led Campbell (1910) to produce a filter with ladder topology, a glance at the circuit diagram of this filter is enough to see its relationship to a loaded transmission line. The cut-off phenomenon is an undesirable side-effect as far as loaded lines are concerned but for telephone FDM filters it is precisely what is required. For this application, Campbell produced band-pass filters to the same ladder topology by replacing the inductors and capacitors with resonators and anti-resonators respectively. Both the loaded line and FDM were of great benefit economically to AT&T and this led to fast development of filtering from this point onwards.

Image filters



Campbell's sketch of the low-pass version of his filter from his 1915 patent showing the now ubiquitous ladder topology with capacitors for the ladder rungs and inductors for the stiles. Filters of more modern design also often adopt the same ladder topology as used by Campbell. It should be understood that although superficially similar, they are really quite different. The ladder construction is essential to the Campbell filter and all the sections have identical element values. Modern designs can be realised in any number of topologies, choosing the ladder topology is merely a matter of convenience. Their response is quite different (better) than Campbell's and the element values, in general, will all be different.

The filters designed by Campbell were named wave filters because of their property of passing some waves and strongly rejecting others. The method by which they were designed was called the image parameter method and filters designed to this method are called image filters. The image method essentially consists of developing the transmission constants of an infinite chain of identical filter sections and then terminating the desired finite number of filter sections in the image impedance. This exactly corresponds to the way the properties of a finite length of transmission line are derived from the theoretical properties of an infinite line, the image impedance corresponding to the characteristic impedance of the line.

From 1920 John Carson, also working for AT&T, began to develop a new way of looking at signals using the operational calculus of Heaviside which in essence is working in the frequency domain. This gave the AT&T engineers a new insight into the way their filters were working and led Otto Zobel to invent many improved forms. Carson and Zobel steadily demolished many of the old ideas. For instance the old telegraph engineers thought of the signal as being a single frequency and this idea persisted into the age of radio with some still believing that frequency modulation (FM) transmission could be achieved with a smaller bandwidth than the baseband signal right up until the publication of Carson's 1922 paper. Another advance concerned the nature of noise, Carson and Zobel (1923) treated noise as a random process with a continuous bandwidth, an idea that was well ahead of its time, and thus limited the amount of noise that it was possible to remove by filtering to that part of the noise spectrum which fell outside the passband. This too, was not generally accepted at first, notably being opposed by Edwin Armstrong (who ironically, actually succeeded in reducing noise with wide-band FM) and was only finally settled with the work of Harry Nyquist whose thermal noise power formula is well known today.

Several improvements were made to image filters and their theory of operation by Otto Zobel. Zobel coined the term constant k filter (or k-type filter) to distinguish Campbell's filter from later types, notably Zobel's m-derived filter (or m-type filter). The particular problems Zobel was trying to address with these new forms were impedance matching into the end terminations and improved steepness of roll-off. These were achieved at the cost of an increase in filter circuit complexity.

A more systematic method of producing image filters was introduced by Hendrik Bode (1930), and further developed by several other investigators including Piloty (1937-1939) and Wilhelm Cauer (1934-1937). Rather than enumerate the behaviour (transfer function, attenuation function, delay function and so on) of a specific circuit, instead a requirement for the image impedance itself was developed. The image impedance can be expressed in terms of the open-circuit and short-circuit impedances of the filter as $Z_i=\sqrt{Z_o Z_s}$. Since the image impedance must be real in the passbands and imaginary in the stopbands according to image theory, there is a requirement that the poles and zeroes of Z_o and Z_s cancel in the passband and correspond in the stopband. The behaviour of the filter can be entirely defined in terms of the positions in the complex plane of these pairs of poles and zeroes. Any circuit which has the requisite poles and zeroes will also have the requisite response. Cauer pursued two related questions arising from this technique: what specification of poles and zeroes are realisable as passive filters; and what realisations are equivalent to each other. The results of this work led Cauer to develop a new approach, now called network synthesis.

This "poles and zeroes" view of filter design was particularly useful where a bank of filters, each operating at different frequencies, are all connected across the same transmission line. The earlier approach was unable to deal properly with this situation, but the poles and zeroes approach could embrace it by specifying a constant impedance for the combined filter. This problem was originally related to FDM telephony but frequently now arises in loudspeaker crossover filters.

Network synthesis filters

The essence of network synthesis is to start with a required filter response and produce a network that delivers that response, or approximates to it within a specified boundary. This is the inverse of network analysis which starts with a given network and by applying the various electric circuit theorems predicts the response of the network. The term was first used with this meaning in the doctoral thesis of Yuk-Wing Lee (1930) and apparently arose out of a conversation with Vannevar Bush. The advantage of network synthesis over previous methods is that it provides a solution which precisely meets the design specification. This is not the case with image filters, a degree of experience is required in their design since the image filter only meets the design specification in the unrealistic case of being terminated in its own image impedance, to produce which would require the exact circuit being sought. Network synthesis on the other hand, takes care of the termination impedances simply by incorporating them into the network being designed.

The development of network analysis needed to take place before network synthesis was possible. The theorems of Gustav Kirchoff and others and the ideas of Charles Steinmetz (phasors) and Arthur Kennelly (complex impedance) laid the groundwork. The concept of a port also played a part in the development of the theory, and proved to be a more useful idea than network terminals. The first milestone on the way to network synthesis was an important paper by Ronald Foster (1924), *A Reactance Theorem*, in which Foster introduces the idea of a driving point impedance, that is, the impedance that is connected to the generator. The expression for this impedance determines the response of the filter and vice versa, and a realisation of the filter can be obtained by expansion of this expression. It is not possible to realise any arbitrary impedance expression as a network. Foster's reactance theorem stipulates necessary and sufficient conditions for realisability: that the reactance must be algebraically increasing with frequency and the poles and zeroes must alternate.

Wilhelm Cauer expanded on the work of Foster (1926) and was the first to talk of realisation of a one-port impedance with a prescribed frequency function. Foster's work considered only reactances (i.e., only LC-kind circuits). Cauer generalised this to any 2-element kind one-port network, finding there was an isomorphism between them. He also found ladder realisations of the network using Thomas Stieltjes' continued fraction expansion. This work was the basis on which network synthesis was built, although Cauer's work was not at first used much by engineers, partly because of the intervention of World War II, partly for reasons explained in the next section and partly because Cauer presented his results using topologies that required mutually coupled inductors and ideal transformers. Although on this last point, it has to be said that transformer coupled double tuned amplifiers are a common enough way of widening bandwidth without sacrificing selectivity.

Image method versus synthesis

Image filters continued to be used by designers long after the superior network synthesis techniques were available. Part of the reason for this may have been simply inertia, but it was largely due to the greater computation required for network synthesis filters, often needing a mathematical iterative process. Image filters, in their simplest form, consist of a chain of repeated, identical sections. The design can be improved simply by adding more sections and the computation required to produce the initial section is on the level of "back of an envelope" designing. In the case of network synthesis filters, on the other hand, the filter is designed as a whole, single entity and to add more sections (i.e., increase the order) the designer would have no option but to go back to the beginning and start over. The advantages of synthesised designs are real, but they are not overwhelming compared to what a skilled image designer could achieve, and in many cases it was more cost effective to dispense with time-consuming calculations. This is simply not an issue with the modern availability of computing power, but in the 1950s it was non-existent, in the 1960s and 1970s available only at cost, and not finally becoming widely available to all designers until the 1980s with the advent of the desktop personal computer. Image filters continued to be designed up to that point and many remained in service into the 21st century.

The computational difficulty of the network synthesis method was addressed by tabulating the component values of a prototype filter and then scaling the frequency and impedance and transforming the bandform to those actually required. This kind of approach, or similar, was already in use with image filters, for instance by Zobel, but the concept of a "reference filter" is due to Sidney Darlington. Darlington (1939), was also the first to tabulate values for network synthesis prototype filters, nevertheless it had to wait until the 1950s before the Cauer-Darlington elliptic filter first came into use.

Once computational power was readily available, it became possible to easily design filters to minimise any arbitrary parameter, for example time delay or tolerance to component variation. The difficulties of the image method were firmly put in the past, and even the need for prototypes became largely superfluous. Furthermore, the advent of active filters eased the computation difficulty because sections could be isolated and iterative processes were not then generally necessary.

Realisability and equivalence

Realisability (that is, which functions are realisable as real impedance networks) and equivalence (which networks equivalently have the same function) are two important questions in network synthesis. Following an analogy with Lagrangian mechanics, Cauer formed the matrix equation,

$$[\mathbf{A}] = s^2[\mathbf{L}] + s[\mathbf{R}] + [\mathbf{D}] = s[\mathbf{Z}]$$

where [**Z**],[**R**],[**L**] and [**D**] are the *n*x*n* matrices of, respectively, impedance, resistance, inductance and elastance of an *n*-mesh network and *s* is the complex frequency operator $s=\sigma+i\omega$. Here [**R**],[**L**] and [**D**] have associated energies corresponding to the kinetic, potential and dissipative heat energies, respectively, in a mechanical system and the already known results from mechanics could be applied here. Cauer determined the driving point impedance by the method of Lagrange multipliers;

$$Z_{\mathrm{p}}(s) = \frac{\det[\mathbf{A}]}{s\, a_{11}}$$

where a_{11} is the complement of the element A_{11} to which the one-port is to be connected. From stability theory Cauer found that [**R**], [**L**] and [**D**] must all be positive-definite matrices for $Z_{\mathrm{p}}(s)$ to be realisable if ideal transformers are not excluded. Realisability is only otherwise restricted by practical limitations on topology. This work is also partly due to Otto Brune (1931), who worked with Cauer in the US prior to Cauer returning to Germany. A well known condition for realisability of a one-port rational impedance due to Cauer (1929) is that it must be a function of *s* that is analytic in the right halfplane ($\sigma>0$), have a positive real part in the right halfplane and take on real values on the real axis. This follows from the Poisson integral representation of these functions. Brune coined the term positive-real for this class of function and proved that it was a necessary and sufficient condition (Cauer had only proved it to be necessary) and they extended the work to LC multiports. A theorem due to Sidney Darlington states that any positive-real function $Z(s)$ can be realised as a lossless two-port terminated in a positive resistor R. No resistors within the network are necessary to realise the specified response.

As for equivalence, Cauer found that the group of real affine transformations,

$$[\mathbf{T}]^T[\mathbf{A}][\mathbf{T}]$$

where,

$$[\mathbf{T}] = \begin{bmatrix} 1 & 0 \cdots 0 \\ T_{21} & T_{22} \cdots T_{2n} \\ \cdot & \cdots \\ T_{n1} & T_{n2} \cdots T_{nn} \end{bmatrix}$$

is invariant in $Z_p(s)$, that is, all the transformed networks are equivalents of the original.

Approximation

The approximation problem in network synthesis is to find functions which will produce realisable networks approximating to a prescribed function of frequency within limits arbitrarily set. The approximation problem is an important issue since the ideal function of frequency required will commonly be unachievable with rational networks. For instance, the ideal prescribed function is often taken to be the unachievable lossless transmission in the passband, infinite attenuation in the stopband and a vertical transition between the two. However, the ideal function can be approximated with a rational function, becoming ever closer to the ideal the higher the order of the polynomial. The first to address this problem was Stephen Butterworth (1930) using his Butterworth polynomials. Independently, Cauer (1931) used Chebyshev polynomials, initially applied to image filters, and not to the now well-known ladder realisation of this filter.

Butterworth filter

Butterworth filters are an important class of filters due to Stephen Butterworth (1930) which are now recognised as being a special case of Cauer's elliptic filters. Butterworth discovered this filter independently of Cauer's work and implemented it in his version with each section isolated from the next with a valve amplifier which made calculation of component values easy since the filter sections could not interact with each other and each section represented one term in the Butterworth polynomials. This gives Butterworth the credit for being both the first to deviate from image parameter theory and the first to design active filters. It was later shown that Butterworth filters could be implemented in ladder topology without the need for amplifiers, possibly the first to do so was William Bennett (1932) in a patent which presents formulae for component values identical to the modern ones. Bennett, at this stage though, is still discussing the design as an artificial transmission line and so is adopting an image parameter approach despite having produced what would now be considered a network synthesis design. He also does not appear to be aware of the work of Butterworth or the connection between them.

Insertion-loss method

The insertion-loss method of designing filters is, in essence, to prescribe a desired function of frequency for the filter as an attenuation of the signal when the filter is inserted between the terminations relative to the level that would have been received were the terminations connected to each other via an ideal transformer perfectly matching them. Versions of this theory are due to Sidney Darlington, Wilhelm Cauer and others all working more or less independently and is often taken as synonymous with network synthesis. Butterworth's filter implementation is, in those terms, an insertion-loss filter, but it is a relatively trivial one mathematically since the active amplifiers used by Butterworth ensured that each stage individually worked into a resistive load. Butterworth's filter becomes a non-trivial example when it is implemented entirely with passive components. An even earlier filter which influenced the insertion-loss method was Norton's dual-band filter where the input of two filters are connected in parallel and designed so that the combined input presents a constant resistance. Norton's design method, together with Cauer's canonical LC networks and Darlington's theorem that only LC components were required in the body of the filter resulted in the insertion-loss method. However, ladder topology proved to be more practical than Cauer's canonical forms.

Darlington's insertion-loss method is a generalisation of the procedure used by Norton. In Norton's filter it can be shown that each filter is equivalent to a separate filter unterminated at the common end. Darlington's method applies to the more straightforward and general case of a 2-port LC network terminated at both ends. The procedure consists of the following steps:

1. determine the poles of the prescribed insertion-loss function,
2. from that find the complex transmission function,
3. from that find the complex reflection coefficients at the terminating resistors,
4. find the driving point impedance from the short-circuit and open-circuit impedances,
5. expand the driving point impedance into an LC (usually ladder) network.

Darlington additionally used a transformation found by Hendrik Bode that predicted the response of a filter using non-ideal components but all with the same *Q*. Darlington used this transformation in reverse to produce filters with a prescribed insertion-loss with non-ideal components. Such filters have the ideal insertion-loss response plus a flat attenuation across all frequencies.

Elliptic filters

Elliptic filters are filters produced by the insertion-loss method which use elliptic rational functions in their transfer function as an approximation to the ideal filter response and the result is called a Chebyshev approximation. This is the same Chebyshev approximation technique used by Cauer on image filters but follows the Darlington insertion-loss design method and uses slightly different elliptic functions. Cauer had some contact with Darlington and Bell Labs before WWII (for a time he worked in the US) but during the war they worked independently, in some cases making the same discoveries. Cauer had disclosed the Chebyshev approximation to Bell Labs but had not left them with the proof. Sergei Schelkunoff provided this and a generalisation to all equal ripple problems. Elliptic filters are a general class of filter which incorporate several other important classes as special cases: Cauer filter (equal ripple in passband and stopband), Chebyshev filter (ripple only in passband), reverse Chebyshev filter (ripple only in stopband) and Butterworth filter (no ripple in either band).

Generally, for insertion-loss filters where the transmission zeroes and infinite losses are all on the real axis of the complex frequency plane (which they usually are for minimum component count), the insertion-loss function can be written as;

$$\frac{1}{1+JF^2}$$

where *F* is either an even (resulting in an antimetric filter) or an odd (resulting in an symmetric filter) function of frequency. Zeroes of *F* correspond to zero loss and the poles of *F* correspond to transmission zeroes. *J* sets the passband ripple height and the stopband loss and these two design requirements can be interchanged. The zeroes and poles of *F* and *J* can be set arbitrarily. The nature of *F* determines the class of the filter;

- if *F* is a Chebyshev approximation the result is a Chebyshev filter,
- if *F* is a maximally flat approximation the result is a passband maximally flat filter,
- if 1/*F* is a Chebyshev approximation the result is a reverse Chebyshev filter,
- if 1/*F* is a maximally flat approximation the result is a stopband maximally flat filter,

A Chebyshev response simultaneously in the passband and stopband is possible, such as Cauer's equal ripple elliptic filter.

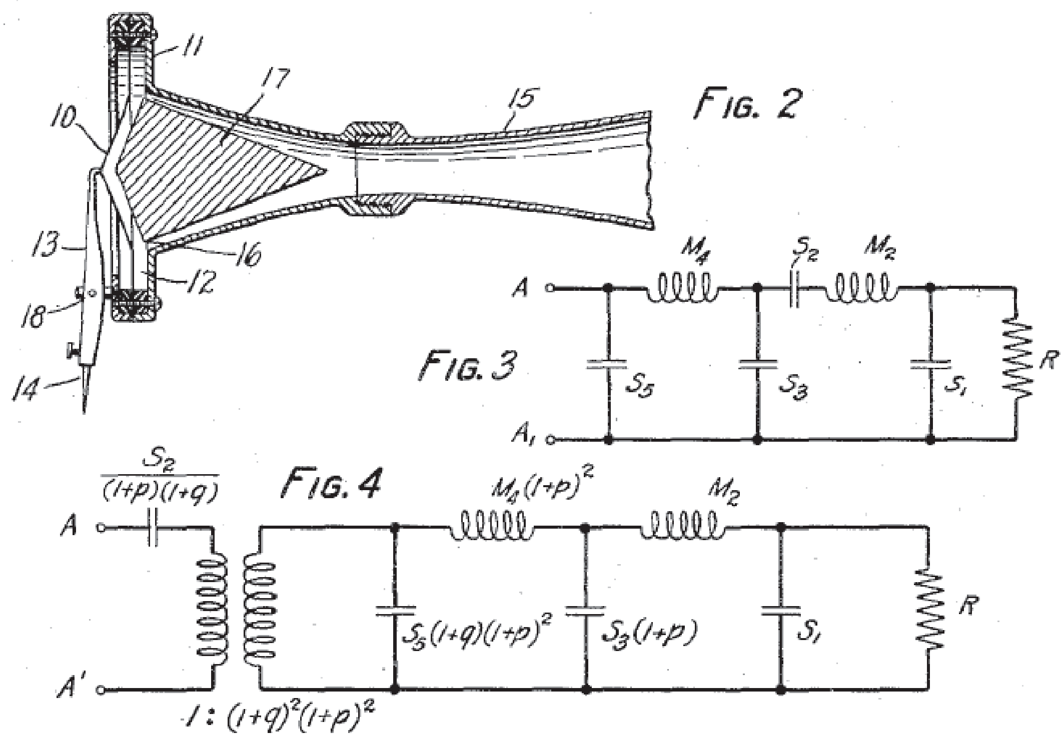
Darlington relates that he found in the New York City library Carl Jacobi's original paper on elliptic functions, published in Latin in 1829. In this paper Darlington was surprised to find foldout tables of the exact elliptic function transformations needed for Chebyshev approximations of both Cauer's image parameter, and Darlington's insertion-loss filters.

Other methods

Darlington considers the topology of coupled tuned circuits to involve a separate approximation technique to the insertion-loss method, but also producing nominally flat passbands and high attenuation stopbands. The most common topology for these is shunt anti-resonators coupled by series capacitors, less commonly, by inductors, or in the case of a two-section filter, by mutual inductance. These are most useful where the design requirement is not too stringent, that is, moderate bandwidth, roll-off and passband ripple.

Other notable developments and applications

Mechanical filters



Norton's mechanical filter together with its electrical equivalent circuit. Two equivalents are shown, "Fig.3" directly corresponds to the physical relationship of the mechanical components; "Fig.4" is an equivalent transformed circuit arrived at by repeated application of a well known transform, the purpose being to remove the series resonant circuit from the body of the filter leaving a simple *LC* ladder network.

Edward Norton, around 1930, designed a mechanical filter for use on phonograph recorders and players. Norton designed the filter in the electrical domain and then used the correspondence of mechanical quantities to electrical quantities to realise the filter using mechanical components. Mass corresponds to inductance, stiffness to elastance and damping to resistance. The filter was designed to have a maximally flat frequency response.

In modern designs it is common to use quartz crystal filters, especially for narrowband filtering applications. The signal exists as a mechanical acoustic wave while it is in the crystal and is converted by transducers between the electrical and mechanical domains at the terminals of the crystal.

Transversal filters

Transversal filters are not usually associated with passive implementations but the concept can be found in a Wiener and Lee patent from 1935 which describes a filter consisting of a cascade of all-pass sections. The outputs of the various sections are summed in the proportions needed to result in the required frequency function. This works by the principle that certain frequencies will be in, or close to antiphase, at different sections and will tend to cancel when added. These are the frequencies rejected by the filter and can produce filters with very sharp cut-offs. This approach did not find any immediate applications, and is not common in passive filters. However, the principle finds many applications as an active delay line implementation for wide band discrete-time filter applications such as television, radar and high-speed data transmission.

Matched filter

The purpose of matched filters is to maximise the signal-to-noise ratio (S/N) at the expense of pulse shape. Pulse shape, unlike many other applications, is unimportant in radar while S/N is the primary limitation on performance. The filters were introduced during WWII (described 1943) by Dwight North and are often eponymously referred to as "North filters".

Filters for control systems

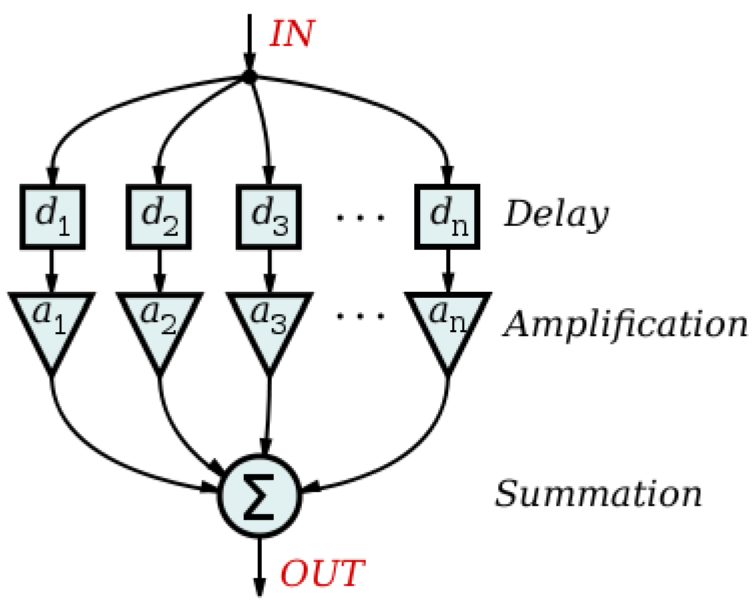
Control systems have a need for smoothing filters in their feedback loops with criteria to maximise the speed of movement of a mechanical system to the prescribed mark and at the same time minimise overshoot and noise induced motions. A key problem here is the extraction of Gaussian signals from a noisy background. An early paper on this was published during WWII by Norbert Wiener with the specific application to anti-aircraft fire control analogue computers. Rudy Kalman (Kalman filter) later reformulated this in terms of state-space smoothing and prediction where it is known as the linear-quadratic-Gaussian control problem. Kalman started an interest in state-space solutions, but according to Darlington this approach can also be found in the work of Heaviside and earlier.

Modern practice

LC passive filters gradually became less popular as active amplifying elements, particularly operational amplifiers, became cheaply available. The reason for the change is that wound components (the usual method of manufacture for inductors) are far from ideal, the wire adding resistance as well as inductance to the component. Inductors are also relatively expensive and are not "off-the-shelf" components. On the other hand, the function of LC ladder sections, LC resonators and RL sections can be replaced by RC components in an amplifier feedback loop (active filters). These components will usually be much more cost effective, and smaller as well. Cheap digital technology, in its turn, has largely supplanted analogue implementations of filters. However, there is still an occasional place for them in the simpler applications such as coupling where sophisticated functions of frequency are not needed.

Chapter- 3

Digital Filter



A general finite impulse response filter with *n* stages, each with an independent delay, d_i, and amplification gain, a_i.

In electronics, computer science and mathematics, a **digital filter** is a system that performs mathematical operations on a sampled, discrete-time signal to reduce or enhance certain aspects of that signal. This is in contrast to the other major type of electronic filter, the analog filter, which is an electronic circuit operating on continuous-time analog signals. An analog signal may be processed by a digital filter by first being digitized and represented as a sequence of numbers, then manipulated mathematically, and then reconstructed as a new analog signal. In an analog filter, the input signal is "directly" manipulated by the circuit.

A digital filter system usually consists of an analog-to-digital converter to sample the input signal, followed by a microprocessor and some peripheral components such as memory to store data and filter coefficients etc. Finally a digital-to-analog converter to complete the output stage. Program Instructions (software) running on the microprocessor implement the digital filter by performing the necessary mathematical operations on the numbers received from the ADC. In some high performance applications, an FPGA or ASIC is used instead of a general purpose microprocessor, or a specialized DSP with specific paralleled architecture for expediting operations such as filtering.

Digital filters may be more expensive than an equivalent analog filter due to their increased complexity, but they make practical many designs that are impractical or impossible as analog filters. Since digital filters use a sampling process and discrete-time processing, they experience latency (the difference in time between the input and the response), which is almost irrelevant in analog filters.

Digital filters are commonplace and an essential element of everyday electronics such as radios, cellphones, and stereo receivers.

Characterization of digital filters

A digital filter is characterized by its transfer function, or equivalently, its difference equation. Mathematical analysis of the transfer function can describe how it will respond to any input. As such, designing a filter consists of developing specifications appropriate to the problem (for example, a second-order low pass filter with a specific cut-off frequency), and then producing a transfer function which meets the specifications.

The transfer function for a linear, time-invariant, digital filter can be expressed as a transfer function in the *Z*-domain; if it is causal, then it has the form:

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots + b_N z^{-N}}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_M z^{-M}}$$

where the order of the filter is the greater of N or M.

This is the form for a recursive filter with both the inputs (Numerator) and outputs (Denominator), which typically leads to an IIR infinite impulse response behaviour, but if the denominator is made equal to unity i.e. no feedback, then this becomes an FIR or finite impulse response filter.

Analysis techniques

A variety of mathematical techniques may be employed to analyze the behaviour of a given digital filter. Many of these analysis techniques may also be employed in designs, and often form the basis of a filter specification.

Typically, one analyzes filters by calculating how the filter will respond to a simple input such as an impulse response. One can then extend this information to visualize the filter's response to more complex signals. Riemann spheres have been used, together with digital video, for this purpose.

Impulse response

The impulse response, often denoted $h[k]$ or h_k, is a measurement of how a filter will respond to the Kronecker delta function. For example, given a difference equation, one would set $x_0 = 1$ and $x_k = 0$ for $k \neq 0$ and evaluate. The impulse response is a characterization of the filter's behaviour. Digital filters are typically considered in two categories: infinite impulse response (IIR) and finite impulse response (FIR). In the case of linear time-invariant FIR filters, the impulse response is exactly equal to the sequence of filter coefficients:

$$y_n = \sum_{k=0}^{n-1} h_k x_{n-k}$$

IIR filters on the other hand are recursive, with the output depending on both current and previous inputs as well as previous outputs. The general form of the an IIR filter is thus:

$$\sum_{m=0}^{M-1} a_m y_{n-m} = \sum_{k=0}^{n-1} b_k x_{n-k}$$

Plotting the impulse response will reveal how a filter will respond to a sudden, momentary disturbance.

Difference equation

In discrete-time systems, the digital filter is often implemented by converting the transfer function to a linear constant-coefficient difference equation (LCCD) via the Z-transform. The discrete frequency-domain transfer function is written as the ratio of two polynomials. For example:

$$H(z) = \frac{(z+1)^2}{(z-\frac{1}{2})(z+\frac{3}{4})}$$

This is expanded:

$$H(z) = \frac{z^2 + 2z + 1}{z^2 + \frac{1}{4}z - \frac{3}{8}}$$

and divided by the highest order of z:

$$H(z)=\frac{1+2z^{-1}+z^{-2}}{1+\frac{1}{4}z^{-1}-\frac{3}{8}z^{-2}}=\frac{Y(z)}{X(z)}$$

The coefficients of the denominator, a_k, are the 'feed-backward' coefficients and the coefficients of the numerator are the 'feed-forward' coefficients, b_k. The resultant linear difference equation is:

$$y[n]=-\sum_{k=1}^{N}a_k y[n-k]+\sum_{k=0}^{M}b_k x[n-k]$$

or, for the example above:

$$\frac{Y(z)}{X(z)}=\frac{1+2z^{-1}+z^{-2}}{1+\frac{1}{4}z^{-1}-\frac{3}{8}z^{-2}}$$

rearranging terms:

$$\Rightarrow (1+\frac{1}{4}z^{-1}-\frac{3}{8}z^{-2})Y(z)=(1+2z^{-1}+z^{-2})X(z)$$

then by taking the inverse z-transform:

$$\Rightarrow y[n]+\frac{1}{4}y[n-1]-\frac{3}{8}y[n-2]=x[n]+2x[n-1]+x[n-2]$$

and finally, by solving for $y[n]$:

$$y[n]=-\frac{1}{4}y[n-1]+\frac{3}{8}y[n-2]+x[n]+2x[n-1]+x[n-2]$$

This equation shows how to compute the next output sample, $y[n]$, in terms of the past outputs, $y[n-p]$, the present input, $x[n]$, and the past inputs, $x[n-p]$. Applying the filter to an input in this form is equivalent to a Direct Form I or II realization, depending on the exact order of evaluation.

Filter design

The design of digital filters is a deceptively complex topic. Although filters are easily understood and calculated, the practical challenges of their design and implementation are significant and are the subject of much advanced research.

There are two categories of digital filter: the recursive filter and the nonrecursive filter. These are often referred to as infinite impulse response (IIR) filters and finite impulse response (FIR) filters, respectively.

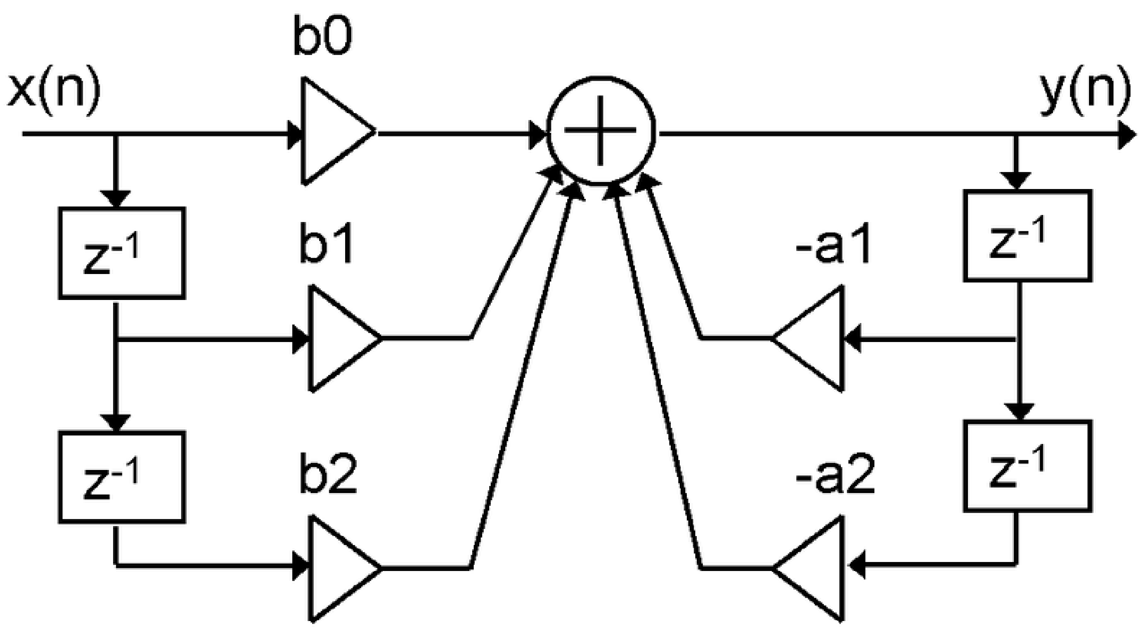
Filter realization

After a filter is designed, it must be *realized* by developing a signal flow diagram that describes the filter in terms of operations on sample sequences.

A given transfer function may be realized in many ways. Consider how a simple expression such as $ax + bx + c$ could be evaluated – one could also compute the equivalent $x(a + b) + c$. In the same way, all realizations may be seen as "factorizations" of the same transfer function, but different realizations will have different numerical properties. Specifically, some realizations are more efficient in terms of the number of operations or storage elements required for their implementation, and others provide advantages such as improved numerical stability and reduced round-off error. Some structures are more optimal for fixed-point arithmetic and others may be more optimal for floating-point arithmetic.

Direct Form I

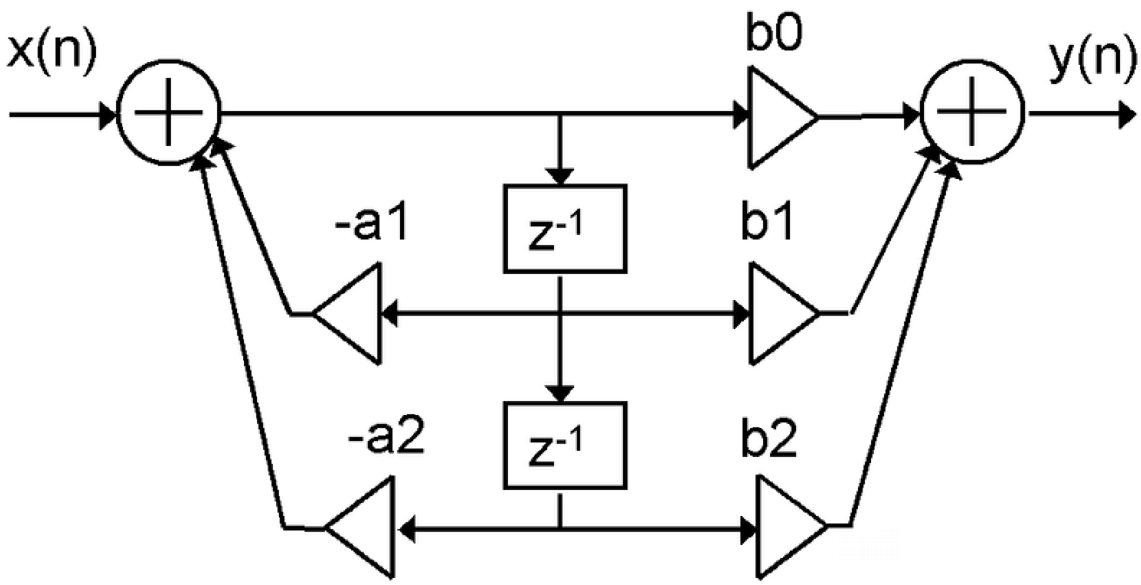
A straightforward approach for IIR filter realization is Direct Form I, where the difference equation is evaluated directly. This form is practical for small filters, but may be inefficient and impractical (numerically unstable) for complex designs. In general, this form requires 2N delay elements (for both input and output signals) for a filter of order N.



Direct Form II

The alternate Direct Form II only needs N delay units, where N is the order of the filter – potentially half as much as Direct Form I. This structure is obtained by reversing the order of the numerator and denominator sections of Direct Form I, since they are in fact two linear systems, and the commutativity property applies. Then, one will notice that there are two columns of delays (z^{-1}) that tap off the center net, and these can be combined since they are redundant, yielding the implementation as shown below.

The disadvantage is that Direct Form II increases the possibility of arithmetic overflow for filters of high Q or resonance. It has been shown that as Q increases, the round-off noise of both direct form topologies increases without bounds. This is because, conceptually, the signal is first passed through an all-pole filter (which normally boosts gain at the resonant frequencies) before the result of that is saturated, then passed through an all-zero filter (which often attenuates much of what the all-pole half amplifies).



Cascaded second-order sections

A common strategy is to realize a higher-order (greater than 2) digital filter as a cascaded series of second-order "biquadratric" (or "biquad") sections. Advantages of this strategy is that the coefficient range is limited. Cascading direct form II sections result in N delay elements for filter order of N. Cascading direct form I sections result in N+2 delay elements since the delay elements of the input of any section (except the first section) are a redundant with the delay elements of the output of the preceding section.

Other Forms

Other forms include:

- Direct Form I and II transpose
- Series/cascade
- Parallel
- Ladder form
- Lattice form
- Coupled normal form
- Multifeedback
- Analog-inspired forms such as Sallen-key and state variable filters
- Systolic arrays

Comparison of analog and digital filters

Digital filters are not subject to the component non-linearities that greatly complicate the design of analog filters. Analog filters consist of imperfect electronic components, whose values are specified to a limit tolerance (e.g. resistor values often have a tolerance of +/- 5%) and which may also change with temperature and drift with time. As the order of an analog filter increases, and thus its component count, the effect of variable component errors is greatly magnified. In digital filters, the coefficient values are stored in computer memory, making them far more stable and predictable.

Because the coefficients of digital filters are definite, they can be used to achieve much more complex and selective designs – specifically with digital filters, one can achieve a lower passband ripple, faster transition, and higher stopband attenuation than is practical with analog filters. Even if the design could be achieved using analog filters, the engineering cost of designing an equivalent digital filter would likely be much lower. Furthermore, one can readily modify the coefficients of a digital filter to make an adaptive filter or a user-controllable parametric filter. While these techniques are possible in an analog filter, they are again considerably more difficult.

Digital filters can be used in the design of finite impulse response filters. Analog filters do not have the same capability, because finite impulse response filters require delay elements.

Digital filters rely less on analog circuitry, potentially allowing for a better signal-to-noise ratio. A digital filter will introduce noise to a signal during analog low pass filtering, analog to digital conversion, digital to analog conversion and may introduce digital noise due to quantization. With analog filters, every component is a source of thermal noise (such as Johnson noise), so as the filter complexity grows, so does the noise.

However, digital filters do introduce a higher fundamental latency to the system. In an analog filter, latency is often negligible; strictly speaking it is the time for an electrical signal to propagate through the filter circuit. In digital filters, latency is a function of the number of delay elements in the system.

Digital filters also tend to be more limited in bandwidth than analog filters. High bandwidth digital filters require expensive ADC/DACs and fast computer hardware for processing.

In very simple cases, it is more cost effective to use an analog filter. Introducing a digital filter requires considerable overhead circuitry, as previously discussed, including two low pass analog filters.

Types of digital filters

Many digital filters are based on the Fast Fourier transform, a mathematical algorithm that quickly extracts the frequency spectrum of a signal, allowing the spectrum to be manipulated (such as to create band-pass filters) before converting the modified spectrum back into a time-series signal.

Another form of a digital filter is that of a state-space model. A well used state-space filter is the Kalman filter published by Rudolf Kalman in 1960.

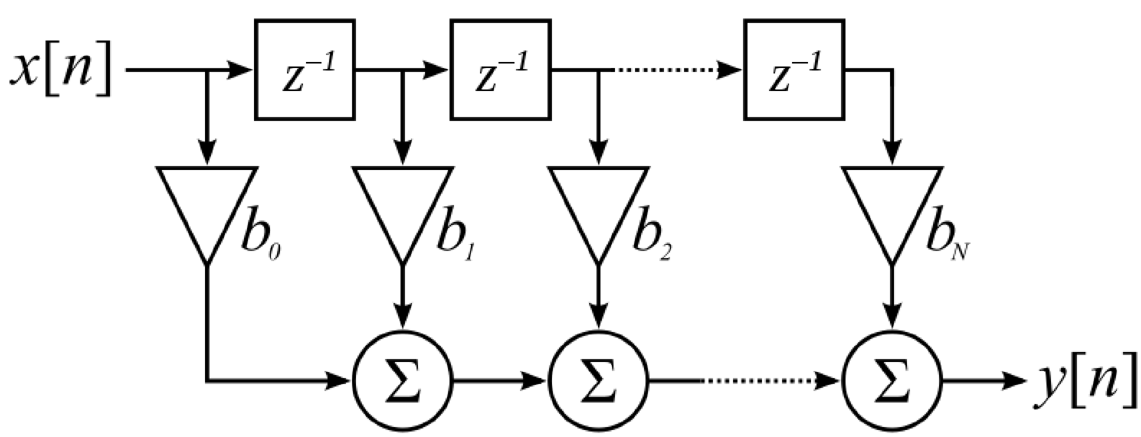
Chapter- 4

Finite Impulse Response

A **finite impulse response (FIR)** filter is a type of a signal processing filter whose impulse response (or response to any finite length input) is of *finite* duration, because it settles to zero in finite time. This is in contrast to infinite impulse response (IIR) filters, which have internal feedback and may continue to respond indefinitely (usually decaying). The impulse response of an Nth-order discrete-time FIR filter (i.e. with a Kronecker delta impulse input) lasts for N+1 samples, and then dies to zero.

FIR filters can be discrete-time or continuous-time, and digital or analog.

Definition



A discrete-time FIR filter of order *N*. The top part is an *N*-stage delay line with *N*+1 taps. Each unit delay is a z^{-1} operator in Z-transform notation.

The output *y* of a linear time invariant system is determined by convolving its input signal *x* with its impulse response *b*.

For a discrete-time FIR filter, the output is a weighted sum of the current and a finite number of previous values of the input. The operation is described by the following equation, which defines the output sequence *y[n]* in terms of its input sequence *x[n]*:

$$y[n] = b_0 x[n] + b_1 x[n-1] + \cdots + b_N x[n-N]$$
$$y[n] = \sum_{i=0}^{N} b_i x[n-i]$$

where:

- $x[n]$ is the input signal,
- $y[n]$ is the output signal,
- b_i are the **filter coefficients**, also known as **tap weights**, that make up the impulse response,
- N is the filter order; an Nth-order filter has $(N + 1)$ terms on the right-hand side. The $x[n - i]$ in these terms are commonly referred to as **taps**, based on the structure of a tapped delay line that in many implementations or block diagrams provides the delayed inputs to the multiplication operations. One may speak of a "5th order/6-tap filter", for instance.

Properties

An FIR filter has a number of useful properties which sometimes make it preferable to an infinite impulse response (IIR) filter. FIR filters:

- Are inherently stable. This is due to the fact that, because there is no feedback, all the poles are located at the origin and thus are located within the unit circle.
- Require no feedback. This means that any rounding errors are not compounded by summed iterations. The same relative error occurs in each calculation. This also makes implementation simpler.
- They can easily be designed to be linear phase by making the coefficient sequence symmetric; linear phase, or phase change proportional to frequency, corresponds to equal delay at all frequencies. This property is sometimes desired for phase-sensitive applications, for example data communications, crossover filters, and mastering.

The main disadvantage of FIR filters is that considerably more computation power in a general purpose processor is required compared to an IIR filter with similar sharpness or selectivity, especially when low frequency (relative to the sample rate) cutoffs are needed. However many digital signal processors provide specialized hardware features to make FIR filters approximately as efficient as IIR for many applications.

Impulse response

The impulse response $h[n]$ can be calculated if we set $x[n] = \delta[n]$in the above relation, where $\delta[n]$ is the Kronecker delta impulse. The impulse response for an FIR filter then becomes the set of coefficients b_n, as follows

$$\begin{aligned} h[n] &= \sum_{i=0}^{N} b_i \delta[n-i] \\ &= b_n. \end{aligned}$$

for $n = 0$to N.

The Z-transform of the impulse response yields the transfer function of the FIR filter

$$\begin{aligned} H(z) &= Z\{h[n]\} \\ &= \sum_{n=-\infty}^{\infty} h[n] z^{-n} \\ &= \sum_{n=0}^{N} b_n\, z^{-n}. \end{aligned}$$

FIR filters are clearly *bounded-input bounded-output* (BIBO) stable, since the output is a sum of a finite number of finite multiples of the input values, so can be no greater than $\sum |b_i|$times the largest value appearing in the input.

Filter design

To design a filter means to select the coefficients such that the system has specific characteristics. The required characteristics are stated in filter specifications. Most of the time filter specifications refer to the frequency response of the filter. There are different methods to find the coefficients from frequency specifications:

1. Window design method
2. Frequency Sampling method
3. Weighted least squares design
4. Parks-McClellan method (also known as the Equiripple, Optimal, or Minimax method). The Remez exchange algorithm is commonly used to find an optimal equiripple set of coefficients. Here the user specifies a desired frequency response, a weighting function for errors from this response, and a filter order N. The algorithm then finds the set of $(N + 1)$ coefficients that minimize the

maximum deviation from the ideal. Intuitively, this finds the filter that is as close as you can get to the desired response given that you can use only $(N + 1)$ coefficients. This method is particularly easy in practice since at least one text includes a program that takes the desired filter and N, and returns the optimum coefficients.

Software packages like MATLAB, GNU Octave, Scilab, and SciPy provide convenient ways to apply these different methods.

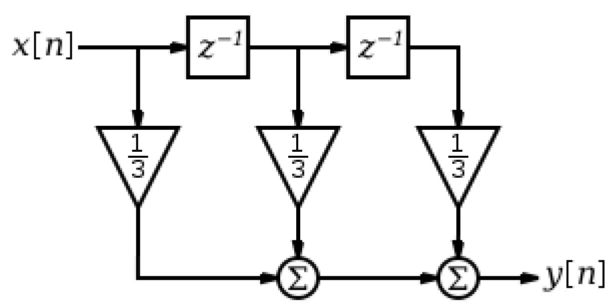
Some filter specifications refer to the time-domain shape of the input signal the filter is expected to "recognize". The optimum matched filter for separating any waveform from white noise is obtained by sampling that shape and using those samples in reverse order as the coefficients of the filter -- giving the filter an impulse response that is the time-reverse of the expected input signal.

Window design method

In the Window Design Method, one designs an ideal IIR filter, then applies a window function to it – in the time domain, multiplying the infinite impulse by the window function. This results in the frequency response of the IIR being convolved with the frequency response of the window function – thus the imperfections of the FIR filter (compared to the ideal IIR filter) can be understood in terms of the frequency response of the window function.

The ideal frequency response of a window is a Dirac delta function, as that results in the frequency response of the FIR filter being identical to that of the IIR filter, but this is not attainable for finite windows, and deviations from this yield differences between the FIR response and the IIR response.

Moving average example



Block diagram of a simple FIR filter (2nd-order/3-tap filter in this case, implementing a moving average)

A moving average filter is a very simple FIR filter. It is sometimes called a *boxcar filter*, especially when followed by decimation. The filter coefficients are found via the following equation:

$$b_i = \frac{1}{N+1} \text{ for } i = 0, 1, \ldots, N$$

To provide a more specific example, we select the filter order:

$$N = 2$$

The impulse response of the resulting filter is:

$$h[n] = \frac{1}{3}\delta[n] + \frac{1}{3}\delta[n-1] + \frac{1}{3}\delta[n-2]$$

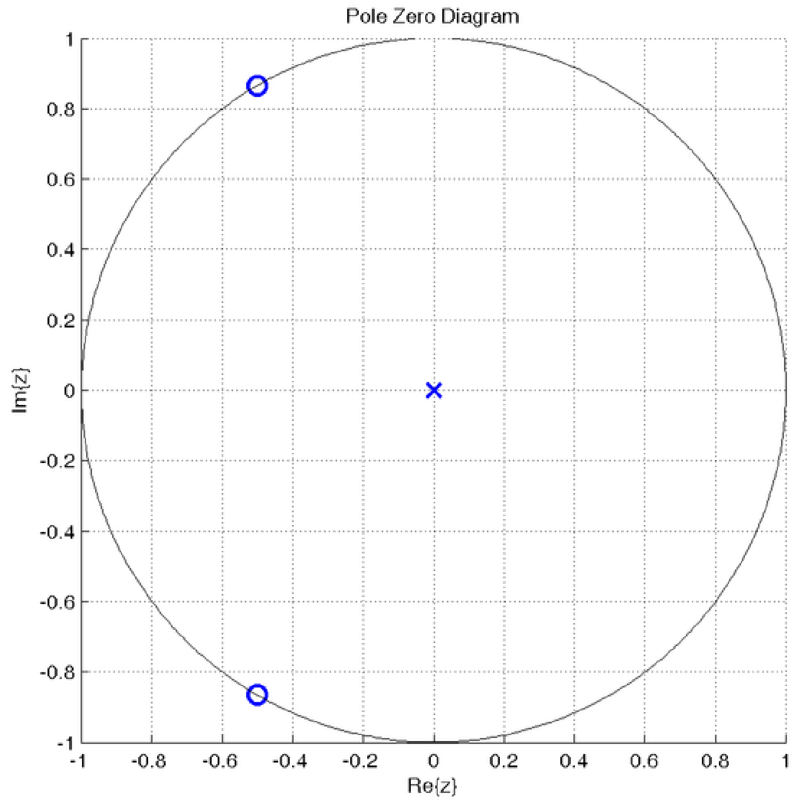
The following figure shows the block diagram of such a 2nd-order moving-average filter.

To discuss stability and spectral topics we take the z-transform of the impulse response:

$$H(z) = \frac{1}{3} + \frac{1}{3}z^{-1} + \frac{1}{3}z^{-2} = \frac{1}{3}\frac{z^2 + z + 1}{z^2}$$

The following figure shows the pole-zero diagram of the filter. Zero frequency (DC) corresponds to (1,0), positive frequencies advancing counterclockwise around the circle to (-1,0) at half the sample frequency.

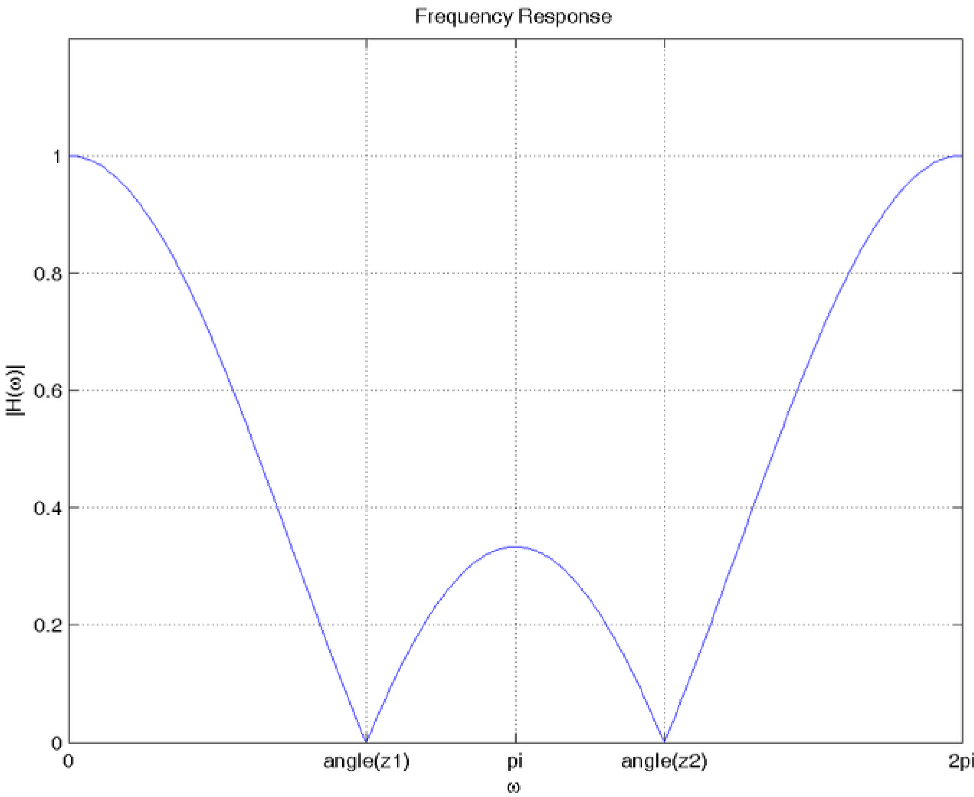
Two poles are located at the origin, and two zeros are located at $z_1 = -\frac{1}{2} + j\frac{\sqrt{3}}{2}$, $z_2 = -\frac{1}{2} - j\frac{\sqrt{3}}{2}$



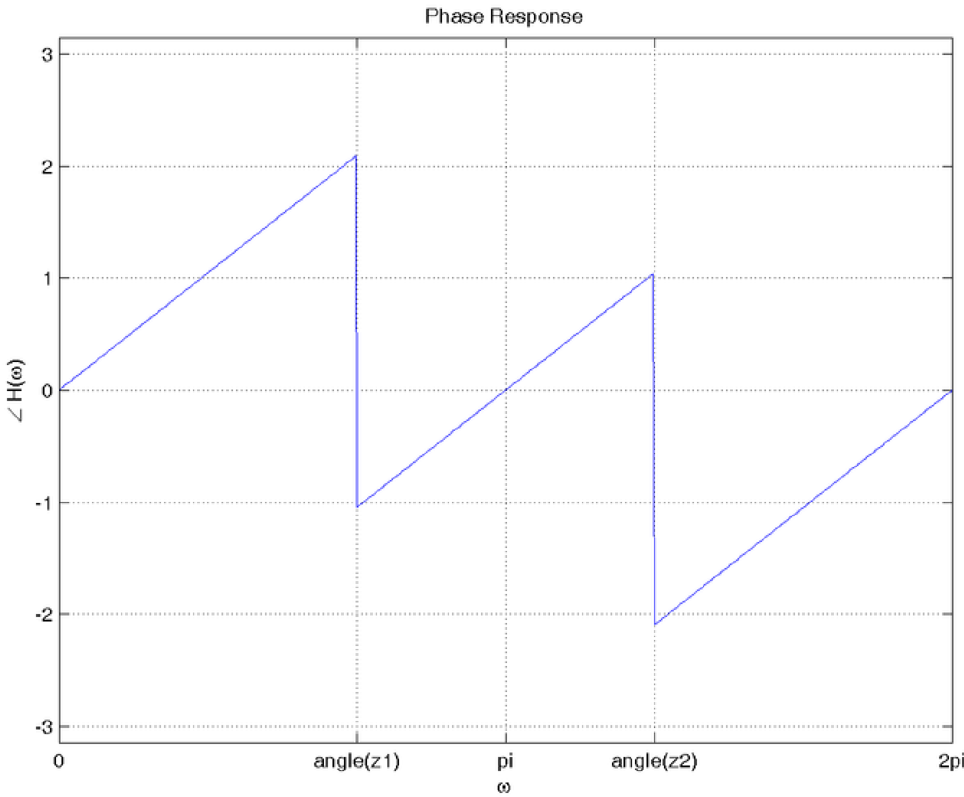
The frequency response, for frequency ω in radians per sample, is:

$$H(e^{j\omega}) = \frac{1}{3} + \frac{1}{3}e^{-j\omega} + \frac{1}{3}e^{-j2\omega}$$

The following figure shows the absolute value of the frequency response. Clearly, the moving-average filter passes low frequencies with a gain near 1, and attenuates high frequencies. This is a typical low-pass filter characteristic. Frequencies above π are aliases of the frequencies below π, and are generally ignored or filtered out if reconstructing a continuous-time signal.

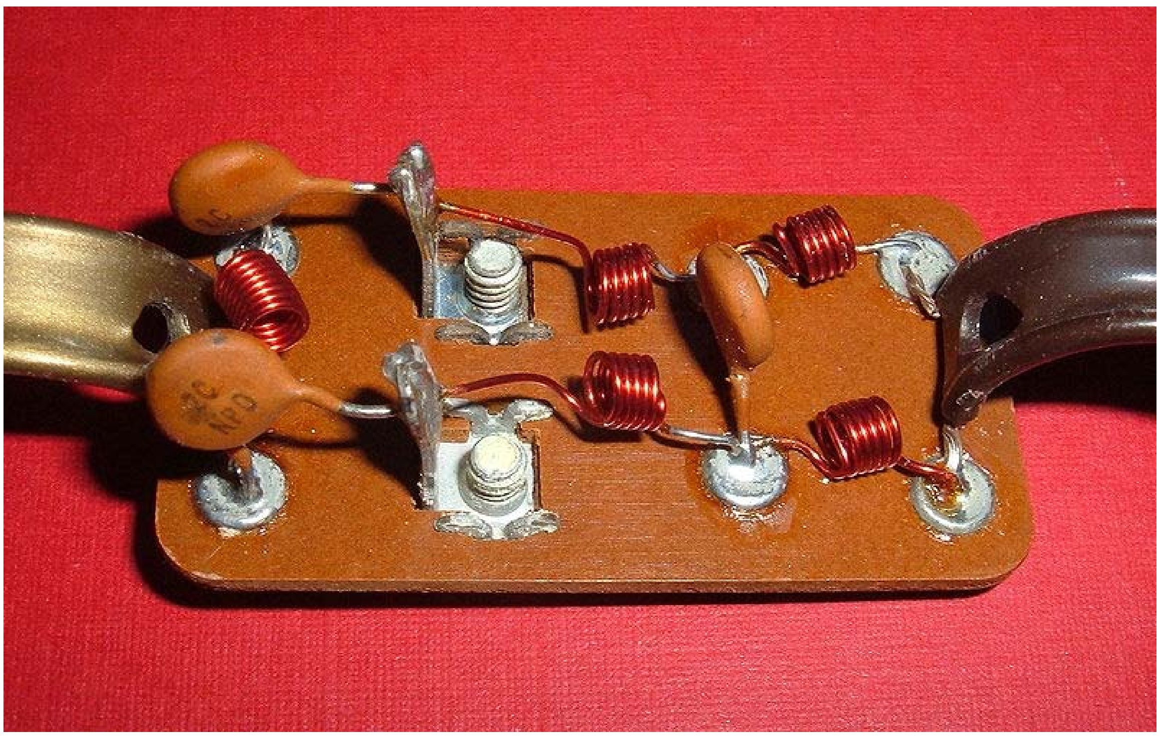


The following figure shows the phase response. Since the phase always follows a straight line except where it has been reduced modulo π radians (should be 2π), the linear phase property is demonstrated.


Phase Response
3
2
1
0
-1
-2
-3
∠ H(ω)
0
angle(z1)
pi
angle(z2)
2pi
ω

Chapter- 5

Electronic Filter



Television signal splitter consisting of a high-pass filter (left) and a low-pass filter (right). The antenna is connected to the screw terminals to the left of center.

Electronic filters are electronic circuits which perform signal processing functions, specifically to remove unwanted frequency components from the signal, to enhance wanted ones, or both. Electronic filters can be:

- passive or active
- analog or digital
- high-pass, low-pass, bandpass, band-reject (band reject; notch), or all-pass.

- discrete-time (sampled) or continuous-time
- linear or non-linear
- infinite impulse response (IIR type) or finite impulse response (FIR type)

The most common types of electronic filters are linear filters, regardless of other aspects of their design.

History

The oldest forms of electronic filters are passive analog linear filters, constructed using only resistors and capacitors or resistors and inductors. These are known as RC and RL single-pole filters respectively. More complex multipole LC filters have also existed for many years, and their operation is well understood.

Hybrid filters are also possible, typically involving a combination of analog amplifiers with mechanical resonators or delay lines. Other devices such as CCD delay lines have also been used as discrete-time filters. With the availability of digital signal processing, active digital filters have become common.

Classification by technology

Passive filters

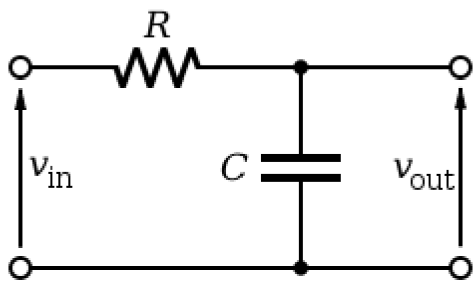
Passive implementations of linear filters are based on combinations of resistors (R), inductors (L) and capacitors (C). These types are collectively known as *passive filters*, because they do not depend upon an external power supply and/or they do not contain active components such as transistors.

Inductors block high-frequency signals and conduct low-frequency signals, while capacitors do the reverse. A filter in which the signal passes through an inductor, or in which a capacitor provides a path to ground, presents less attenuation to low-frequency signals than high-frequency signals and is a *low-pass filter*. If the signal passes through a capacitor, or has a path to ground through an inductor, then the filter presents less attenuation to high-frequency signals than low-frequency signals and is a *high-pass filter*. Resistors on their own have no frequency-selective properties, but are added to inductors and capacitors to determine the *time-constants* of the circuit, and therefore the frequencies to which it responds.

The inductors and capacitors are the reactive elements of the filter. The number of elements determines the order of the filter. In this context, an LC tuned circuit being used in a band-pass or band-stop filter is considered a single element even though it consists of two components.

At high frequencies (above about 100 megahertz), sometimes the inductors consist of single loops or strips of sheet metal, and the capacitors consist of adjacent strips of metal. These inductive or capacitive pieces of metal are called stubs.

Single element types



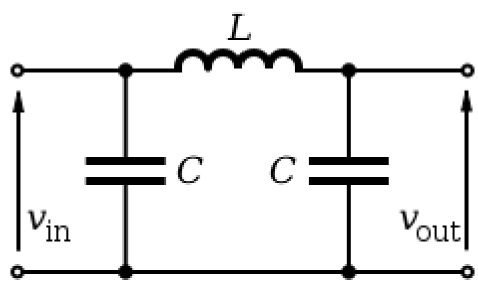
A low-pass electronic filter realised by an RC circuit

The simplest passive filters, RC and RL filters, include only one reactive element, except hybrid LC filter which is characterized by inductance and capacitance integrated in one element.

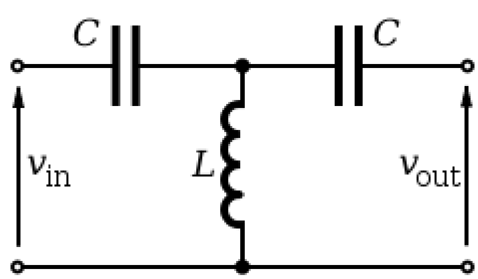
L filter

An L filter consists of two reactive elements, one in series and one in parallel.

T and π filters



Low-pass π filter



High-pass T filter

Three-element filters can have a 'T' or 'π' topology and in either geometries, a low-pass, high-pass, band-pass, or band-stop characteristic is possible. The components can be chosen symmetric or not, depending on the required frequency characteristics. The high-pass T filter in the illustration, has a very low impedance at high frequencies, and a very high impedance at low frequencies. That means that it can be inserted in a transmission line, resulting in the high frequencies being passed and low frequencies being reflected. Likewise, for the illustrated low-pass π filter, the circuit can be connected to a transmission line, transmitting low frequencies and reflecting high frequencies. Using m-derived filter sections with correct termination impedances, the input impedance can be reasonably constant in the pass band.

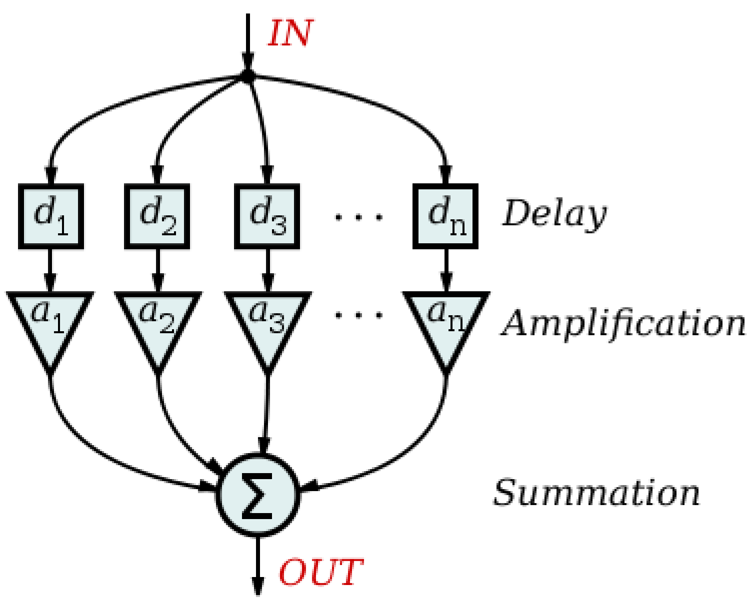
Multiple element types

Multiple element filters are usually constructed as a ladder network. These can be seen as a continuation of the L,T and π designs of filters. More elements are needed when it is desired to improve some parameter of the filter such as stop-band rejection or slope of transition from pass-band to stop-band.

Active filters

Active filters are implemented using a combination of passive and active (amplifying) components, and require an outside power source. Operational amplifiers are frequently used in active filter designs. These can have high Q factor, and can achieve resonance without the use of inductors. However, their upper frequency limit is limited by the bandwidth of the amplifiers used.

Digital filters



A general finite impulse response filter with *n* stages, each with an independent delay, d_i and amplification gain, a_i.

Digital signal processing allows the inexpensive construction of a wide variety of filters. The signal is sampled and an analog-to-digital converter turns the signal into a stream of numbers. A computer program running on a CPU or a specialized DSP (or less often running on a hardware implementation of the algorithm) calculates an output number stream. This output can be converted to a signal by passing it through a digital-to-analog converter. There are problems with noise introduced by the conversions, but these can be controlled and limited for many useful filters. Due to the sampling involved, the input signal must be of limited frequency content or aliasing will occur.

Other filter technologies

Quartz filters and piezoelectrics

In the late 1930s, engineers realized that small mechanical systems made of rigid materials such as quartz would acoustically resonate at radio frequencies, i.e. from audible frequencies (sound) up to several hundred megahertz. Some early resonators were made of steel, but quartz quickly became favored. The biggest advantage of quartz is that it is piezoelectric. This means that quartz resonators can directly convert their own mechanical motion into electrical signals. Quartz also has a very low coefficient of thermal expansion which means that quartz resonators can produce stable frequencies over a wide temperature range. Quartz crystal filters have much higher quality factors than LCR filters. When higher stabilities are required, the crystals and their driving circuits may be mounted in a "crystal oven" to control the temperature. For very narrow band filters, sometimes several crystals are operated in series.

Engineers realized that a large number of crystals could be collapsed into a single component, by mounting comb-shaped evaporations of metal on a quartz crystal. In this scheme, a "tapped delay line" reinforces the desired frequencies as the sound waves flow across the surface of the quartz crystal. The tapped delay line has become a general scheme of making high-*Q* filters in many different ways.

SAW filters

SAW (surface acoustic wave) filters are electromechanical devices commonly used in radio frequency applications. Electrical signals are converted to a mechanical wave in a device constructed of a piezoelectric crystal or ceramic; this wave is delayed as it propagates across the device, before being converted back to an electrical signal by further electrodes. The delayed outputs are recombined to produce a direct analog implementation of a finite impulse response filter. This hybrid filtering technique is also found in an analog sampled filter. SAW filters are limited to frequencies up to 3 GHz.

BAW filters

BAW (Bulk Acoustic Wave) filters are electromechanical devices. BAW filters can implement ladder or lattice filters. BAW filters typically operate at frequencies from around 2 to around 16 GHz, and may be smaller or thinner than equivalent SAW filters.

Two main variants of BAW filters are making their way into devices, Thin film bulk acoustic resonator or FBAR and Solid Mounted Bulk Acoustic Resonators.

Garnet filters

Another method of filtering, at microwave frequencies from 800 MHz to about 5 GHz, is to use a synthetic single crystal yttrium iron garnet sphere made of a chemical combination of yttrium and iron (**YIGF**, or **yttrium iron garnet filter**). The garnet sits on a strip of metal driven by a transistor, and a small loop antenna touches the top of the sphere. An electromagnet changes the frequency that the garnet will pass. The advantage of this method is that the garnet can be tuned over a very wide frequency by varying the strength of the magnetic field.

Atomic filters

For even higher frequencies and greater precision, the vibrations of atoms must be used. Atomic clocks use caesium masers as ultra-high *Q* filters to stabilize their primary oscillators. Another method, used at high, fixed frequencies with very weak radio signals, is to use a ruby maser tapped delay line.

The transfer function

The transfer function $H(s)$ of a filter is the ratio of the output signal $Y(s)$ to that of the input signal $X(s)$ as a function of the complex frequency s:

$$H(s) = \frac{Y(s)}{X(s)}$$

with $s = \sigma + j\omega$.

The transfer function of all linear time-invariant filters, when constructed of discrete components, will be the ratio of two polynomials in s, i.e. a rational function of s. The order of the transfer function will be the highest power of s encountered in either the numerator or the denominator.

Classification by topology

Electronic filters can be classified by the technology used to implement them. Filters using passive filter and active filter technology can be further classified by the particular electronic filter topology used to implement them.

Any given filter transfer function may be implemented in any electronic filter topology.

Some common circuit topologies are:

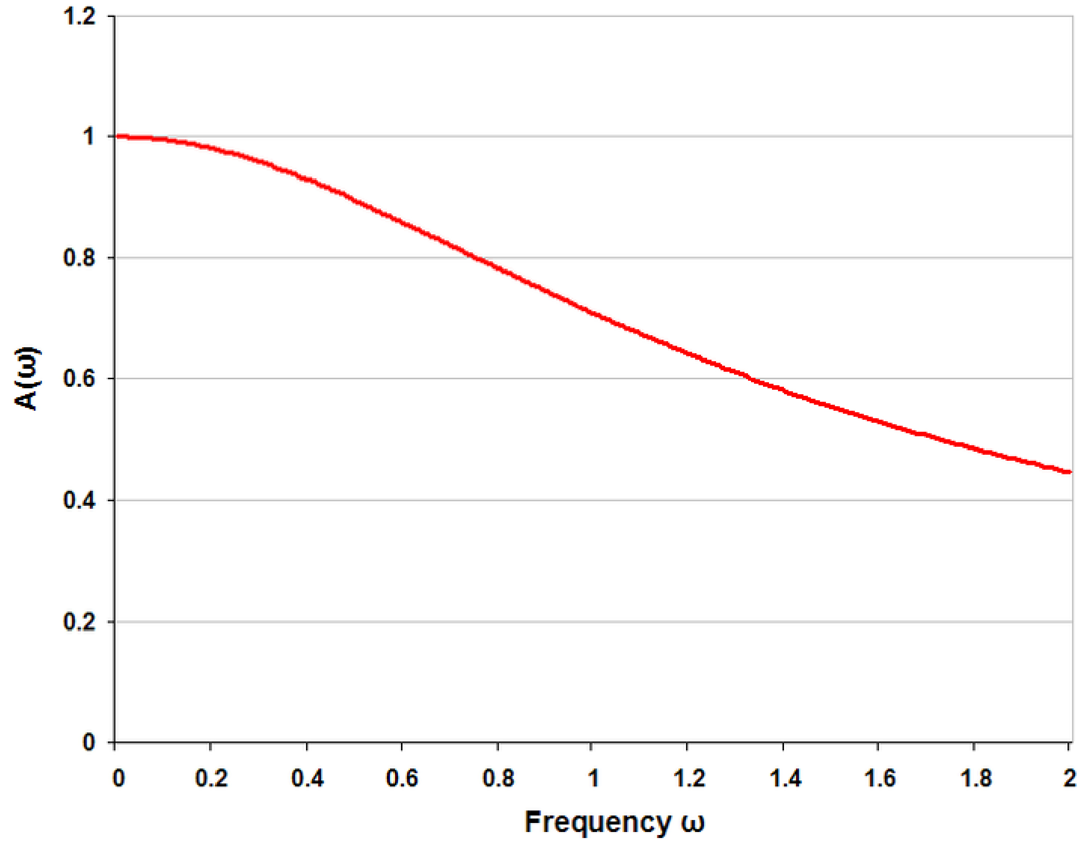
- Cauer topology - Passive
- Sallen Key topology - Active
- Multiple Feedback topology - Active
- State Variable Topology - Active
- Biquadratic topology biquad filter - Active

Classification by design methodology

Historically, linear analog filter design has evolved through three major approaches. The oldest designs are simple circuits where the main design criterion was the Q factor of the circuit. This reflected the radio receiver application of filtering as Q was a measure of the frequency selectivity of a tuning circuit. From the 1920s filters began to be designed from the image point of view, mostly being driven by the requirements of telecommunications. After World War II the dominant methodology was network synthesis. The higher mathematics used originally required extensive tables of polynomial coefficient values to be published but modern computer resources have made that unnecessary.

Direct circuit analysis

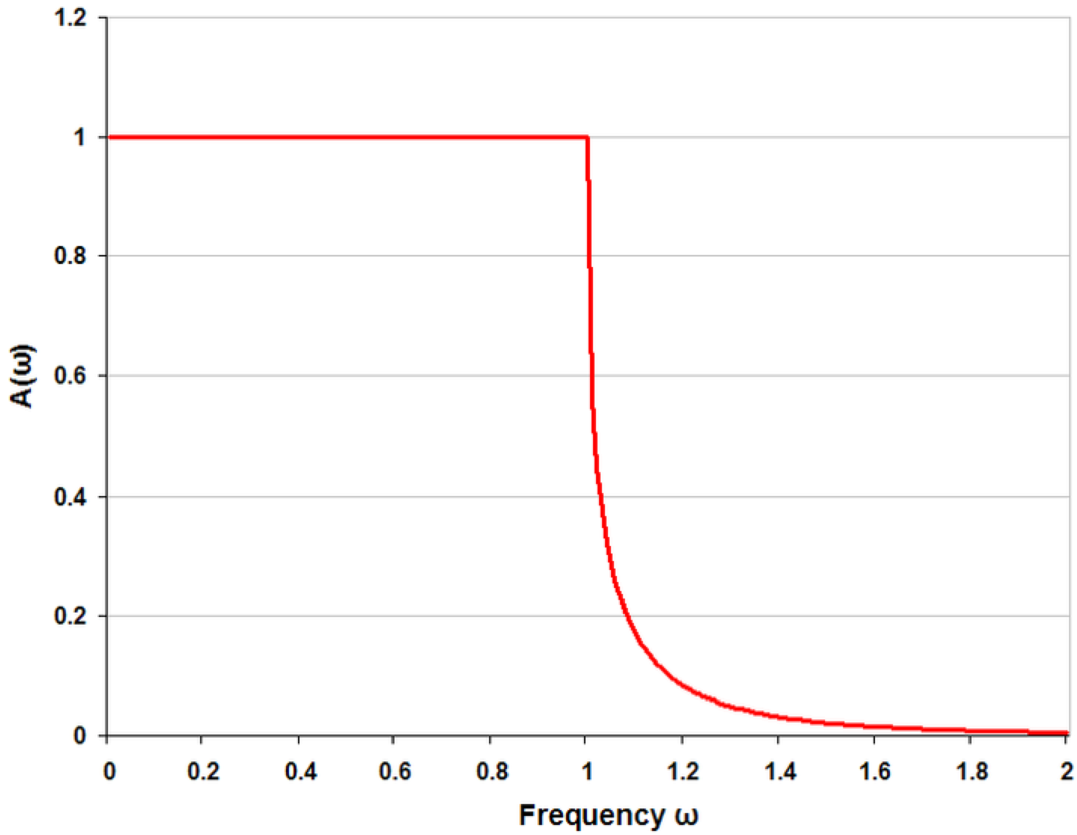
Low order filters can be designed by directly applying basic circuit laws such as Kirchoff's laws to obtain the transfer function. This kind of analysis is usually only carried out for simple filters of 1st or 2nd order.



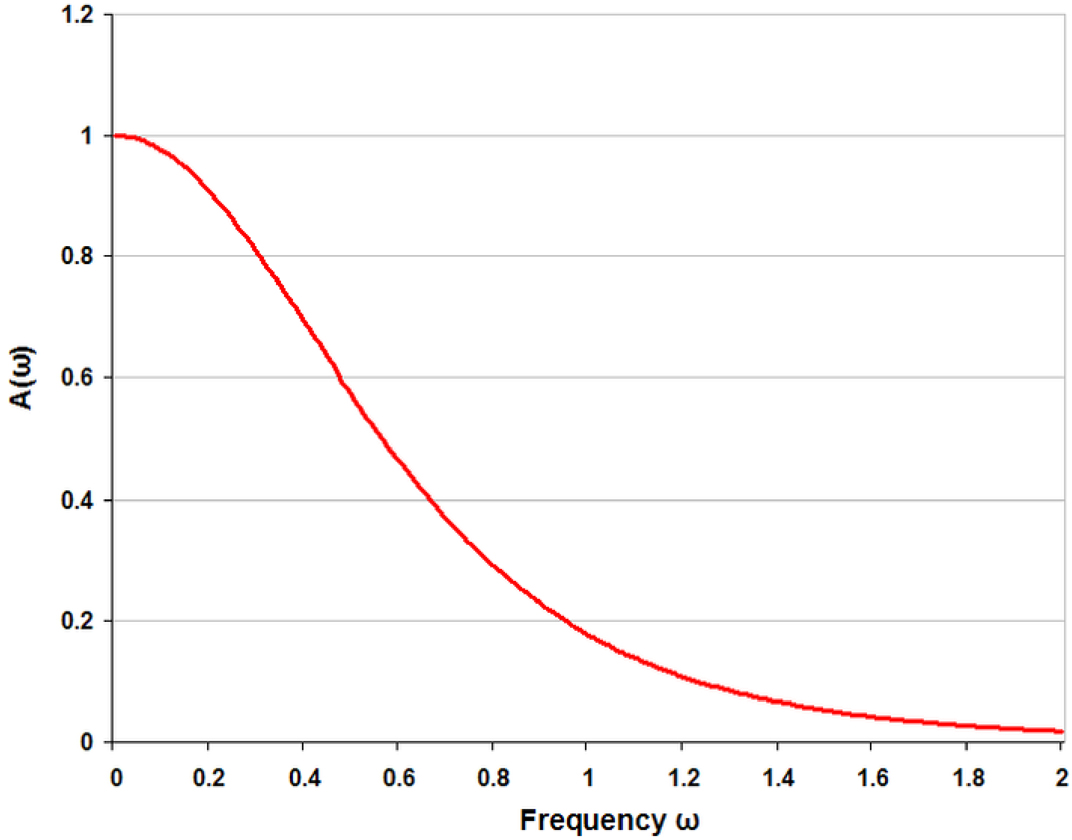
RL filter frequency response

Image impedance analysis

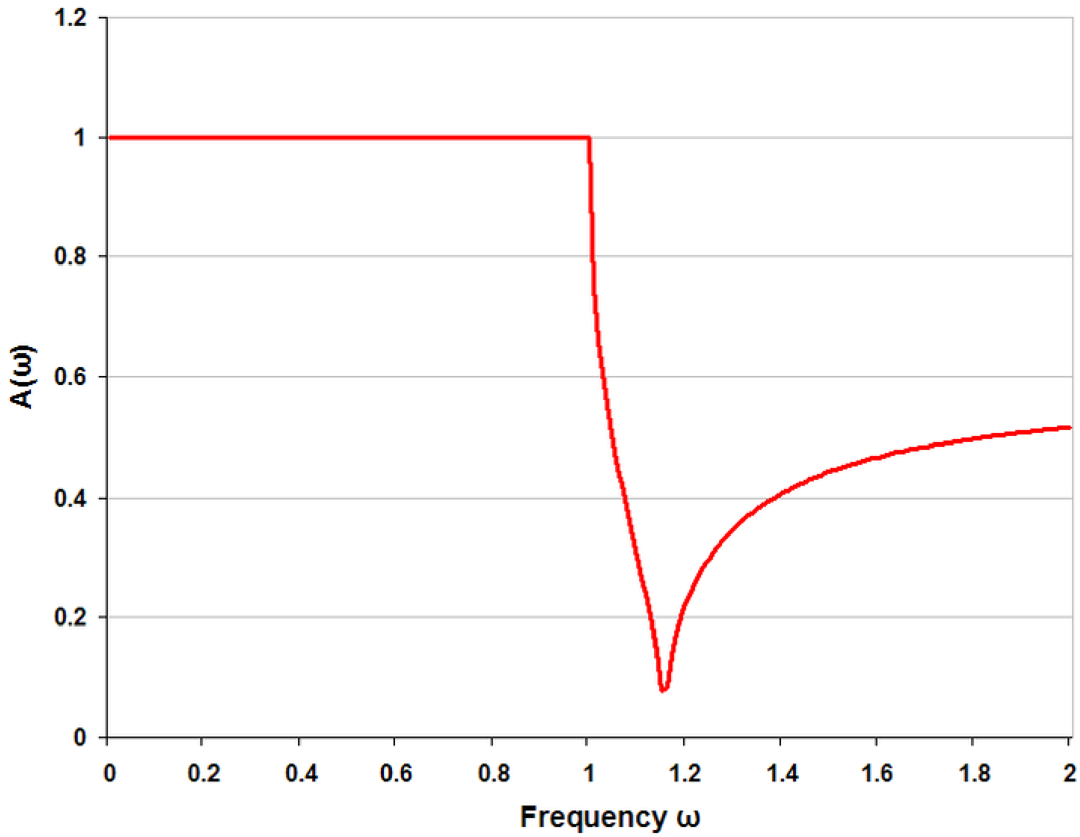
This approach analyses the filter sections from the point of view of the filter being in an infinite chain of identical sections. It has the advantages of simplicity of approach and the ability to easily extend to higher orders. It has the disadvantage that accuracy of predicted responses relies on filter terminations in the image impedance, which is usually not the case.



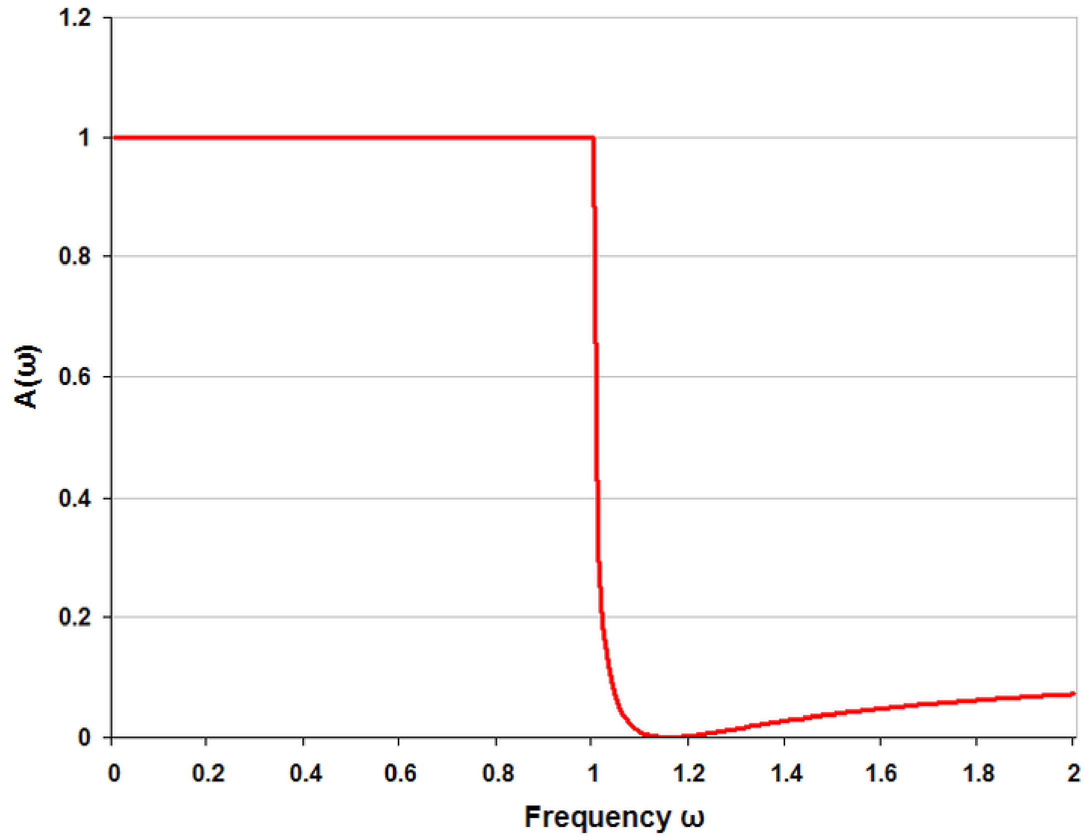
Constant k filter response with 5 elements



Zobel network (constant R) filter, 5 sections



m-derived filter response, m=0.5, 2 elements

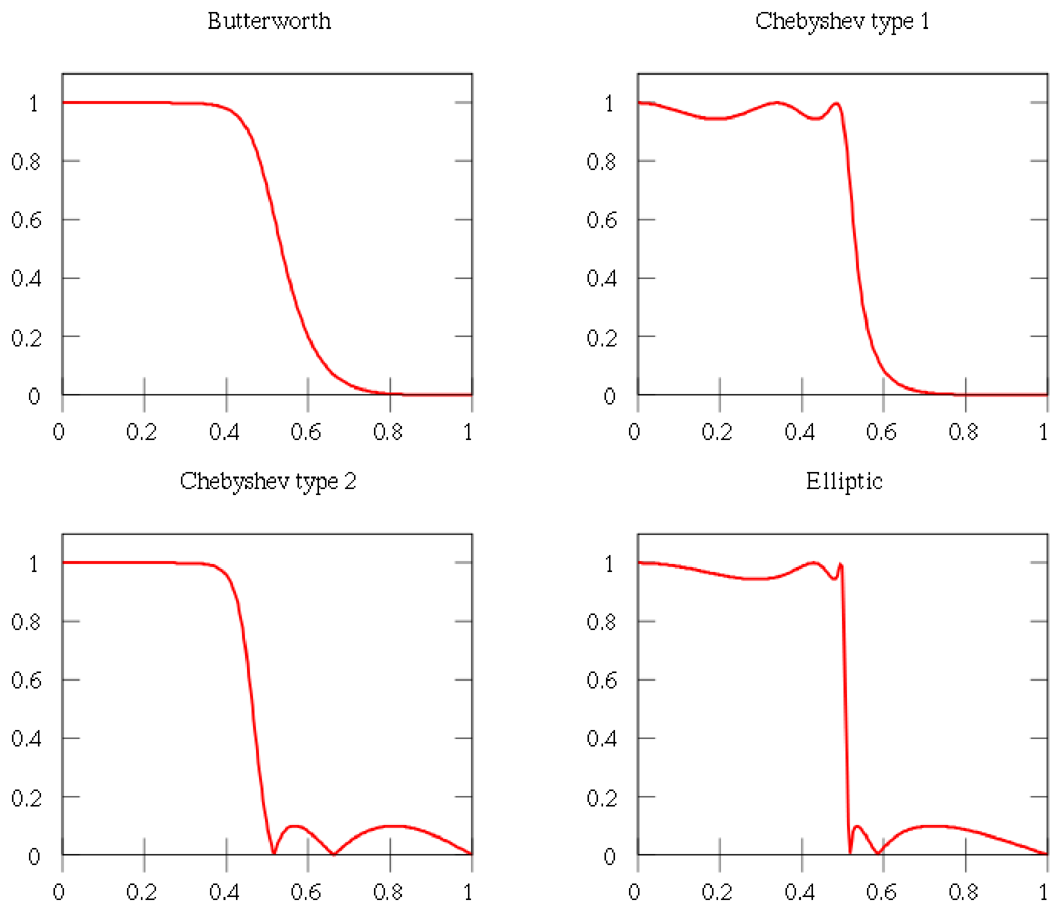


m-derived filter response, m=0.5, 5 elements

Network synthesis

The network synthesis approach starts with a required transfer function and then expresses that as a polynomial equation of the input impedance of the filter. The actual element values of the filter are obtained by continued-fraction or partial-fraction expansions of this polynomial. Unlike the image method, there is no need for impedance matching networks at the terminations as the effects of the terminating resistors are included in the analysis from the start.

Here is an image comparing Butterworth, Chebyshev, and elliptic filters. The filters in this illustration are all fifth-order low-pass filters. The particular implementation – analog or digital, passive or active – makes no difference; their output would be the same.



As is clear from the image, elliptic filters are sharper than all the others, but they show ripples on the whole bandwidth.

Chapter- 6

Equalization Filter, Band-Pass Filter & Band-Stop Filter

Equalization filter

An **equalization** (**EQ**) **filter** is a filter, usually adjustable, designed to compensate for the unequal frequency response of some other signal processing circuit or system. In audio engineering, the EQ filter is more often used creatively to alter the frequency response characteristics of a musical source or a sound mix.

An EQ filter typically allows the user to adjust one or more parameters that determine the overall shape of the filter's transfer function. It is generally used to improve the fidelity of sound, to emphasize certain instruments, to remove undesired noises, or to create completely new and different timbres.

Equalizers may be designed with peaking filters, shelving filters, bandpass filters, or high-pass and low-pass filters. Dynamic range circuitry can be linked with an EQ filter to make timbre changes only after a signal passes an amplitude threshold, or to dynamically increase or reduce amplitude based on the level of a frequency band. Such circuitry is involved in de-essing and in pop-filtering.

There are three primary types of equalizers with peaking filters:

- parametric equalizers
- graphic equalizers
- notch filters

Parametric EQ

Sometimes called **presence filters**; equalizers with peaking filters normally have three user parameters:

- Center Frequency - Equalizers built on peaking filters use a form of bell curve (not precisely the same as the gaussian function) which allows the equalizer to operate smoothly across a range of frequencies. The parameter controls the center frequency or resonant frequency which occurs at the peak (or dip) of the bell curve and is the frequency most affected by equalization. It is often notated as f_c and expressed in Hz.

- *Q* - A parameter that controls the width of the bell curve and is related to the Q factor of resonant circuits and systems. Higher *Q* results in narrower bandwidth and are related by

  $$Q = \frac{\sqrt{2^B}}{2^B - 1},$$

  where *B* is the bandwidth in octaves.
  A high *Q* means that only a narrow frequency range around the center frequency, f_c, is affected by the equalizer, whereas a low *Q* affects a wide frequency range.

- Boost/Cut gain - A parameter that controls the amount, in decibels (dB), that content at the center frequency is boost or reduced in amplitude affecting how much of the selected frequencies should be present. A boost means that those frequencies around f_c will be louder after being equalized, whereas a cut will soften them. A boost or gain of +3 dB will double the sound power of the affected frequencies after equalization. However, a boost of around +10dB is required for the perceived loudness to be twice as loud to the human ear.

A parametric equalizer uses independent parameters for *Q*, center frequency, and boost/cut. Any range of frequencies can be selected and then processed. This is the most powerful EQ because it allows control over all three variables. This EQ is predominantly used in recording and mixing.

A semi-parametric equalizer, also known as a sweepable filter, allows users to control the amplitude and frequency, but uses a fixed bandwidth or *Q*. In some cases, semi-parametric equalizers have a switch that allows the user to select between a wide and a narrow preset bandwidth.

A graphic equalizer uses predetermined *Q* and frequency ranges which are equally spaced according to the musical intervals (equally spaced in log frequency), such as the octave (12-band graphic EQ) or one third of an octave (36-band graphic EQ). Each frequency band can then be independently boosted or cut. This type of EQ is often used for live applications, such as concerts.

A notch filter is an EQ with a very high fixed *Q*. The frequency and boost/cut remain variable. This kind of EQ is useful in multimedia applications and in audio mastering.

Shelving filters

Shelving filters, unlike those described above, boost or cut from a determined frequency until they reach a preset level which is applied to the rest of the frequency spectrum. This kind of filter is usually found on the treble and bass controls of home audio units. These filters are also used in audio mastering.

High pass and low pass filters boost or cut frequencies above or below a selected frequency, called the cutoff frequency. A high pass filter allows only frequencies above the cutoff frequency to pass through unaffected. Frequencies below the cutoff frequency are then attenuated at a constant rate per octave. Low pass filters operate similarly, except that only frequencies below the cutoff are allowed to pass through. Common attenuation rates are 6, 12, and 18 dB per octave. These filters are used to reduce noise and hiss, eliminate pops, and remove rumble. It is common to use a high pass filter (at about 60 to 80 Hz) when recording vocals to eliminate rumble.

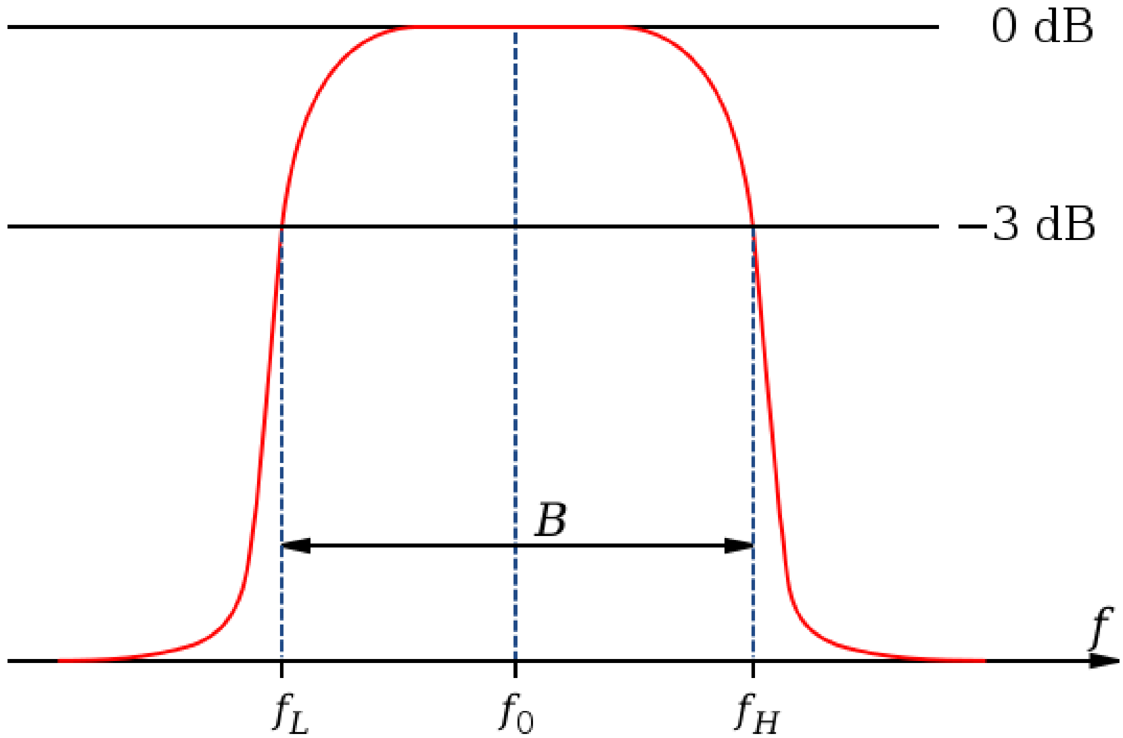
Almost all filters (both analog and digital) induce phase shift on the outgoing audio signal, which can cause a problem in mixing. The higher the value of Q, the more this phase shifting occurs. Therefore, EQ is often used sparingly, unless a particular effect is desired.

Telecommunications lines

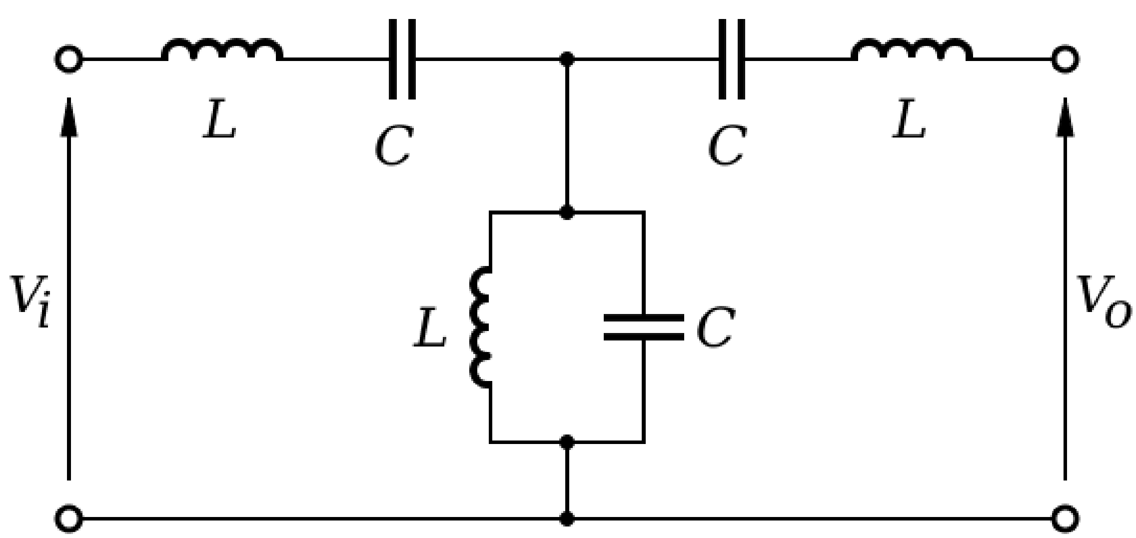
Prior to the widespread use of digital technology, it was common to use equalizers on analog landlines used for trunking. There was a need for the circuitry to be passive and balanced, a requirement for which the Zobel network constant resistance filter was ideally suited, having also the quality of good impedance matching. For lines used for the purpose of broadcast transmissions, phase and delay might also need to be equalized, for which an all-pass filter equalizer would be used.

The first equalizer may have been due to Sally Pero (later Sally Pero Mead) of AT&T Corp. which she designed for use on a submarine telegraph cable receiver. It was a one-port device wired across the line intended to improve the telegraph signalling speed.

Band-pass filter



Bandwidth measured at half-power points (gain -3 dB, √2/2, or about 0.707 relative to peak) on a diagram showing magnitude transfer function versus frequency for a band-pass filter



A medium-complexity example of a band-Pass filter

A **band-pass filter** is a device that passes frequencies within a certain range and rejects (attenuates) frequencies outside that range. An example of an analogue electronic band-pass filter is an RLC circuit (a resistor–inductor–capacitor circuit). These filters can also be created by combining a low-pass filter with a high-pass filter.

Bandpass is an adjective that describes a type of filter or filtering process; it is frequently confused with passband, which refers to the actual portion of affected spectrum. The two words are both compound words that follow the English rules of formation: the primary meaning is the latter part of the compound, while the modifier is the first part. Hence, one may correctly say 'A dual bandpass filter has two passbands'. A *bandpass signal* is a signal containing a band of frequencies away from zero frequency, such as a signal that comes out of a bandpass filter.

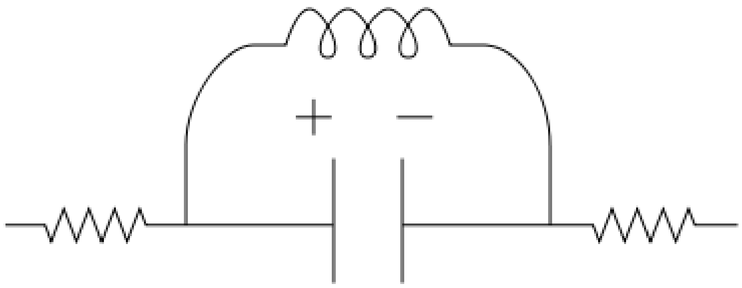
An ideal bandpass filter would have a completely flat passband (e.g. with no gain/attenuation throughout) and would completely attenuate all frequencies outside the passband. Additionally, the transition out of the passband would be instantaneous in frequency. In practice, no bandpass filter is ideal. The filter does not attenuate all frequencies outside the desired frequency range completely; in particular, there is a region just outside the intended passband where frequencies are attenuated, but not rejected. This is known as the filter roll-off, and it is usually expressed in dB of attenuation per octave or decade of frequency. Generally, the design of a filter seeks to make the roll-off as narrow as possible, thus allowing the filter to perform as close as possible to its intended design. Often, this is achieved at the expense of pass-band or stop-band *ripple*.

The bandwidth of the filter is simply the difference between the upper and lower cutoff frequencies. The shape factor is the ratio of bandwidths measured using two different attenuation values to determine the cutoff frequency, e.g., a shape factor of 2:1 at 30/3 dB means the bandwidth measured between frequencies at 30 dB attenuation is twice that measured between frequencies at 3 dB attenuation.

Outside of electronics and signal processing, one example of the use of band-pass filters is in the atmospheric sciences. It is common to band-pass filter recent meteorological data with a period range of, for example, 3 to 10 days, so that only cyclones remain as fluctuations in the data fields.

In neuroscience, visual cortical simple cells were first shown by David Hubel and Torsten Wiesel to have response properties that resemble Gabor filters, which are band-pass.

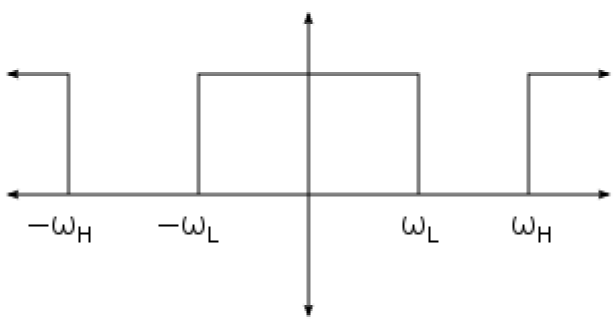
References in popular culture



The schematic showing "Kilroy", which Pynchon said represents a band-pass filter.

In his first novel, *V.*, Thomas Pynchon writes that a schematic for the band pass filter was the origin for the popular graffiti character, Kilroy.

Band-stop filter



A generic ideal band-stop filter, showing both positive and negative angular frequencies

In signal processing, a **band-stop filter** or **band-rejection filter** is a filter that passes most frequencies unaltered, but attenuates those in a specific range to very low levels. It is the opposite of a band-pass filter. A **notch filter** is a band-stop filter with a narrow stopband (high Q factor). Notch filters are used in live sound reproduction (Public Address systems, also known as PA systems) and in instrument amplifier (especially amplifiers or preamplifiers for acoustic instruments such as acoustic guitar, mandolin, bass instrument amplifier, etc.) to reduce or prevent feedback, while having little noticeable effect on the rest of the frequency spectrum. Other names include 'band limit filter', 'T-notch filter', 'band-elimination filter', and 'band-reject filter'.

Typically, the width of the stopband is less than 1 to 2 decades (that is, the highest frequency attenuated is less than 10 to 100 times the lowest frequency attenuated). In the audio band, a notch filter uses high and low frequencies that may be only semitones apart.

Audio example 1: Anti-hum filter for countries using 60 Hz power lines

- Low Freq: 59 Hz
- High Freq: 61 Hz

This means that the filter passes all frequencies, except for the range of 59–61 Hz. This would be used to filter out the mains hum from the 60 Hz power line, though its higher harmonics could still be present. The common European version of the filter would have a 49–51 Hz range.

Audio example 2: Anti-presence filter

- Low Freq: 1 kHz
- High Freq: 4 kHz

RF example 1: Non-linearities of power amplifiers For instance, when measuring non-linearities of power amplifiers a very narrow notch filter could be very useful to avoid the carrier so maximum input power of e.g. a spectrum analyser used to detect spurious content will not be exceeded.

RF example 2: Wave trap A notch filter, usually a simple LC circuit, used to remove a specific interfering frequency. This is a technique used with radio receivers that are so close to a transmitter that it swamps all other signals. The wave trap is used to remove, or greatly reduce, the signal from the local transmitter.

Optical notch filters sometimes rely on destructive interference.

Chapter- 7

Butterworth Filter

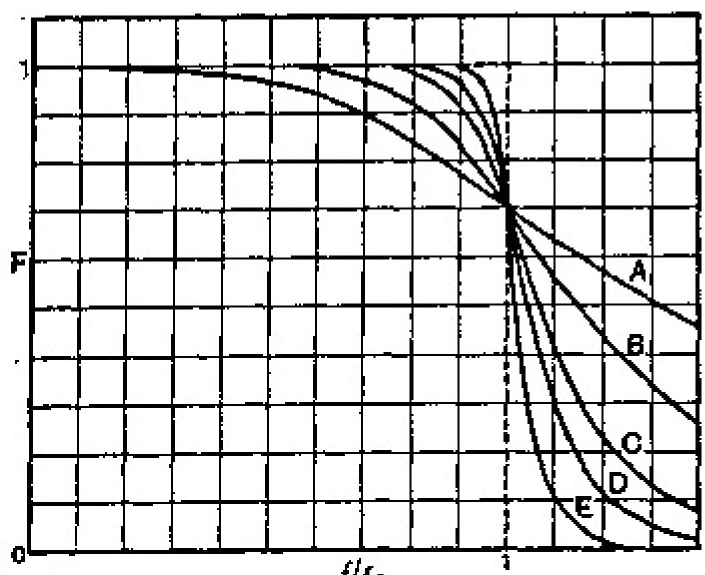


Fig. 3.

The frequency response plot from Butterworth's 1930 paper.

The **Butterworth filter** is a type of signal processing filter designed to have as flat a frequency response as possible in the passband so that it is also termed a **maximally flat magnitude filter**. It was first described by the British engineer Stephen Butterworth in his paper entitled "On the Theory of Filter Amplifiers".

Original paper

Butterworth had a reputation for solving "impossible" mathematical problems. At the time filter design was largely by trial and error because of their mathematical complexity. His paper was far ahead of its time: the filter was not in common use for over 30 years after its publication. Butterworth stated that;

"An ideal electrical filter should not only completely reject the unwanted frequencies but should also have uniform sensitivity for the wanted frequencies."

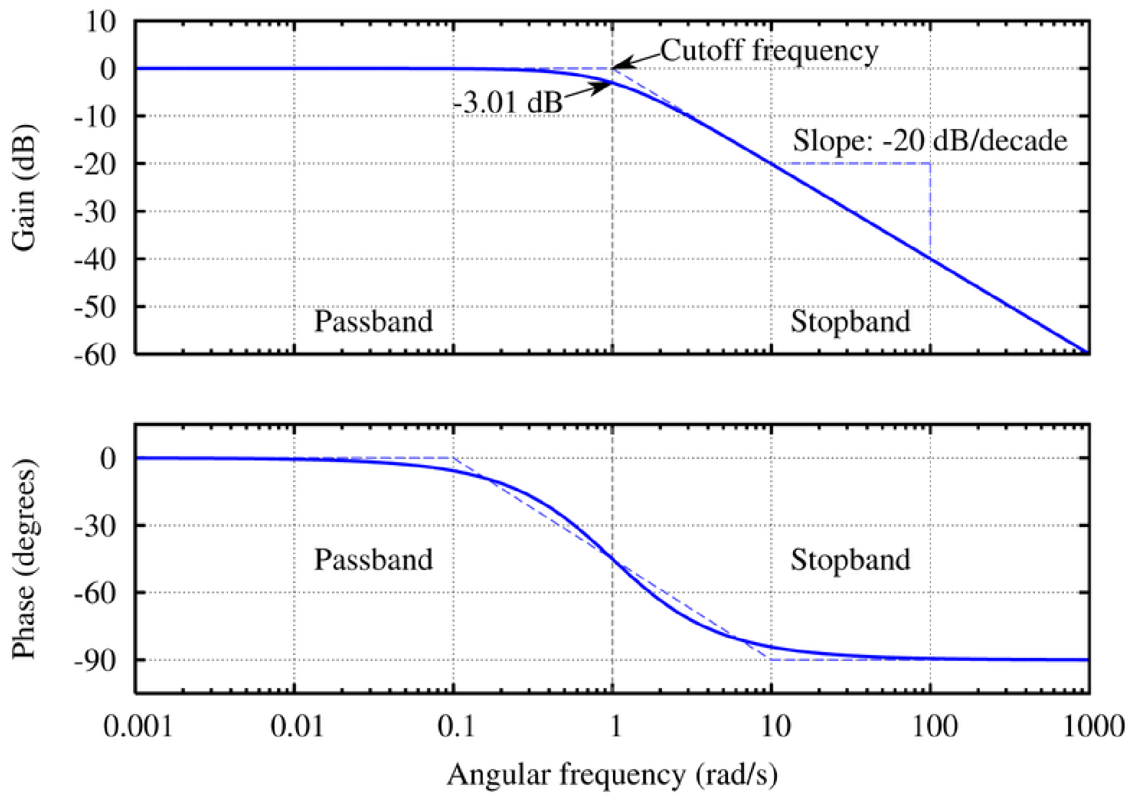
At the time filters generated substantial ripple in the passband and the choice of component values was highly interactive. Butterworth showed that low pass filters could be designed whose frequency response (gain) was;

$$G = \sqrt{\frac{1}{1 + \omega^{2n}}}$$

where ω is the angular frequency in radians per second and n is the number of reactive elements (poles) in the filter. Butterworth only dealt with filters with an even number of poles in his paper: he may have been unaware that such filters could be designed with an odd number of poles. His plot of the frequency response of 2, 4, 6, 8, and 10 pole filters is shown as A, B, C, D, and E in his original graph.

Butterworth solved the equations for two- and four-pole filters, showing how the latter could be cascaded when separated by vacuum tube amplifiers and so enabling the construction of higher-order filters despite inductor losses. In 1930 low-loss core materials such as molypermalloy had not been discovered and air-cored audio inductors were rather lossy. Butterworth discovered that it was possible to adjust the component values of the filter to compensate for the winding resistance of the inductors. He also showed that his basic low-pass filter could be modified to give low-pass, high-pass, band-pass and band-stop functionality.

Overview

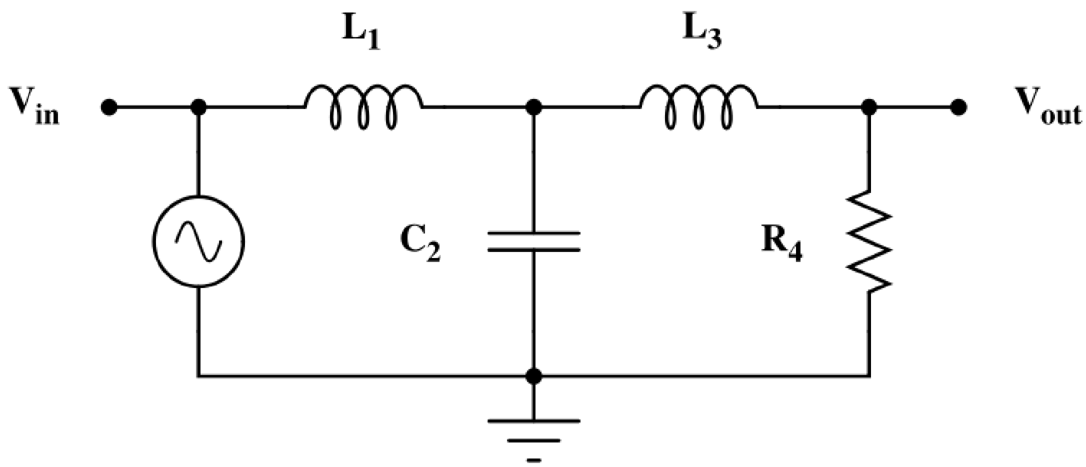


The Bode plot of a first-order Butterworth low-pass filter

The frequency response of the Butterworth filter is maximally flat (has no ripples) in the passband and rolls off towards zero in the stopband. When viewed on a logarithmic Bode plot the response slopes off linearly towards negative infinity. A first-order filter's response rolls off at −6 dB per octave (−20 dB per decade) (all first-order lowpass filters have the same normalized frequency response). A second-order filter decreases at −12 dB per octave, a third-order at −18 dB and so on. Butterworth filters have a monotonically changing magnitude function with ω, unlike other filter types that have non-monotonic ripple in the passband and/or the stopband.

Compared with a Chebyshev Type I/Type II filter or an elliptic filter, the Butterworth filter has a slower roll-off, and thus will require a higher order to implement a particular stopband specification, but Butterworth filters have a more linear phase response in the pass-band than Chebyshev Type I/Type II and elliptic filters can achieve.

A simple example



A third-order low-pass filter (Cauer topology). The filter becomes a Butterworth filter with cutoff frequency ω_c=1 when (for example) C_2=4/3 farad, R_4=1 ohm, L_1=3/2 henry and L_3=1/2 henry.

A simple example of a Butterworth filter is the third-order low-pass design shown in the figure on the right, with $C_2 = 4 / 3$ farad, $R_4 = 1$ ohm, $L_1 = 3 / 2$ and $L_3 = 1 / 2$ henry. Taking the impedance of the capacitors C to be $1/Cs$ and the impedance of the inductors L to be Ls, where $s = \sigma + j\omega$ is the complex frequency, the circuit equations yield the transfer function for this device;

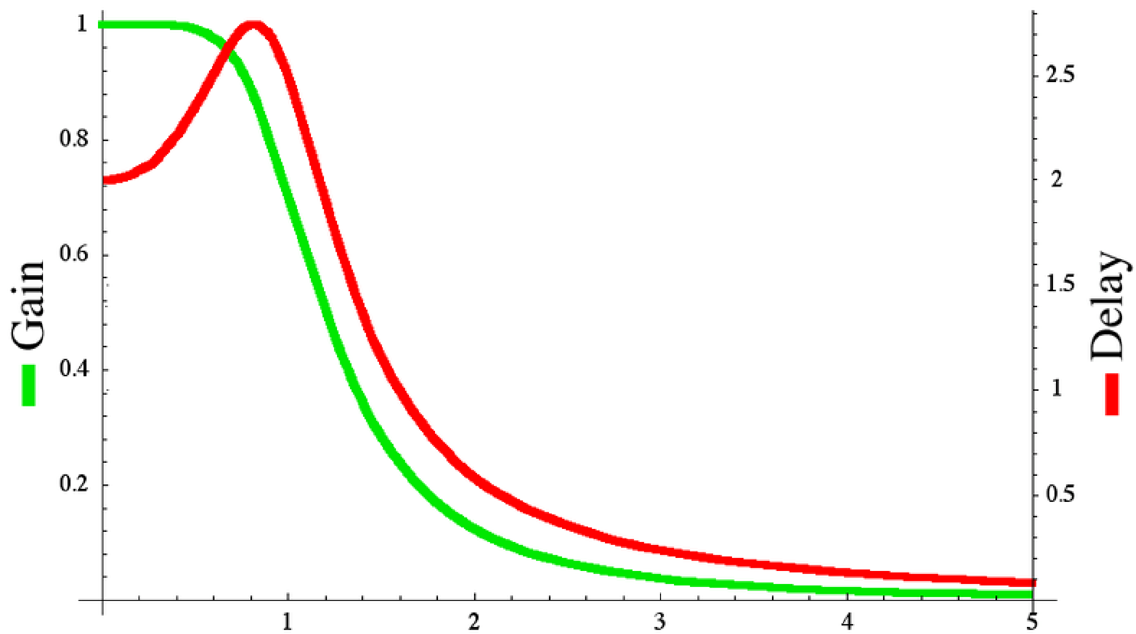
$$H(s) = \frac{V_o(s)}{V_i(s)} = \frac{1}{1 + 2s + 2s^2 + s^3}$$

The magnitude of the frequency response (gain) $G(\omega)$ is given by;

$$G^2(\omega) = |H(j\omega)|^2 = \frac{1}{1 + \omega^6}$$

and the phase is given by;

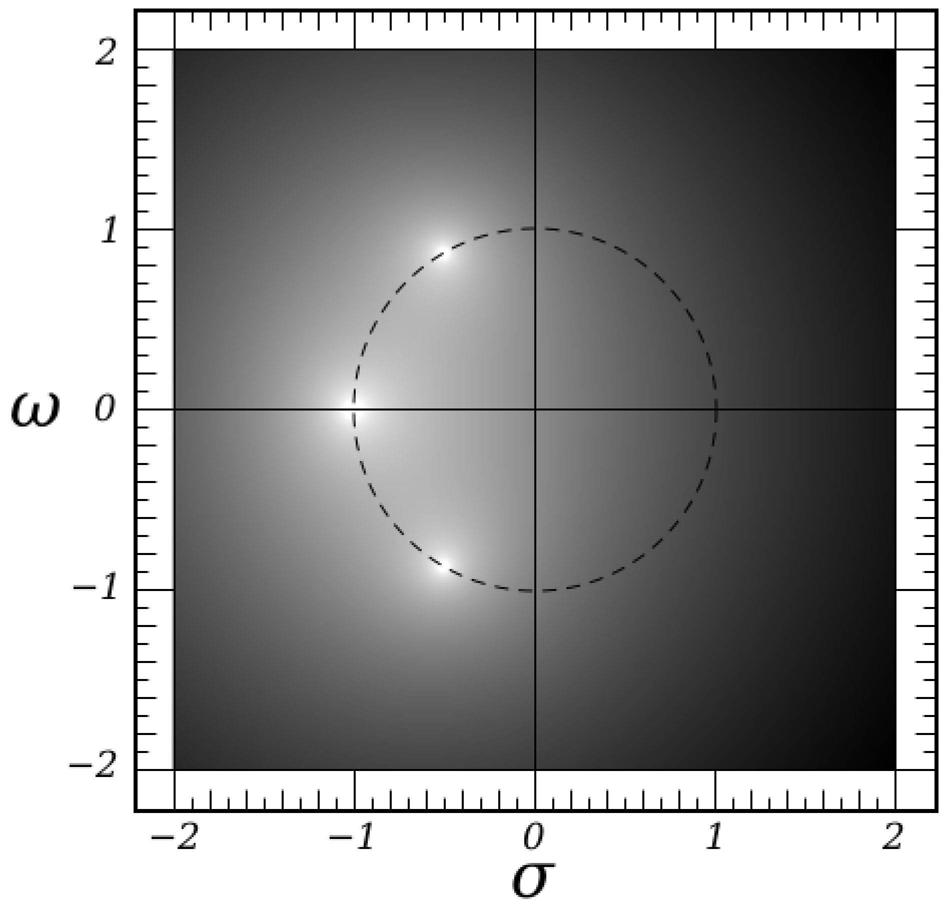
$$\Phi(\omega) = \arg(H(j\omega))$$



Gain and group delay of the third-order Butterworth filter with $\omega_c=1$

The group delay is defined as the derivative of the phase with respect to angular frequency and is a measure of the distortion in the signal introduced by phase differences for different frequencies. The gain and the delay for this filter are plotted in the graph on the left. It can be seen that there are no ripples in the gain curve in either the passband or the stop band.

The log of the absolute value of the transfer function *H(s)* is plotted in complex frequency space in the second graph on the right. The function is defined by the three poles in the left half of the complex frequency plane.



Log density plot of the transfer function H(s) in complex frequency space for the third-order Butterworth filter with ω_c=1. The three poles lie on a circle of unit radius in the left half-plane.

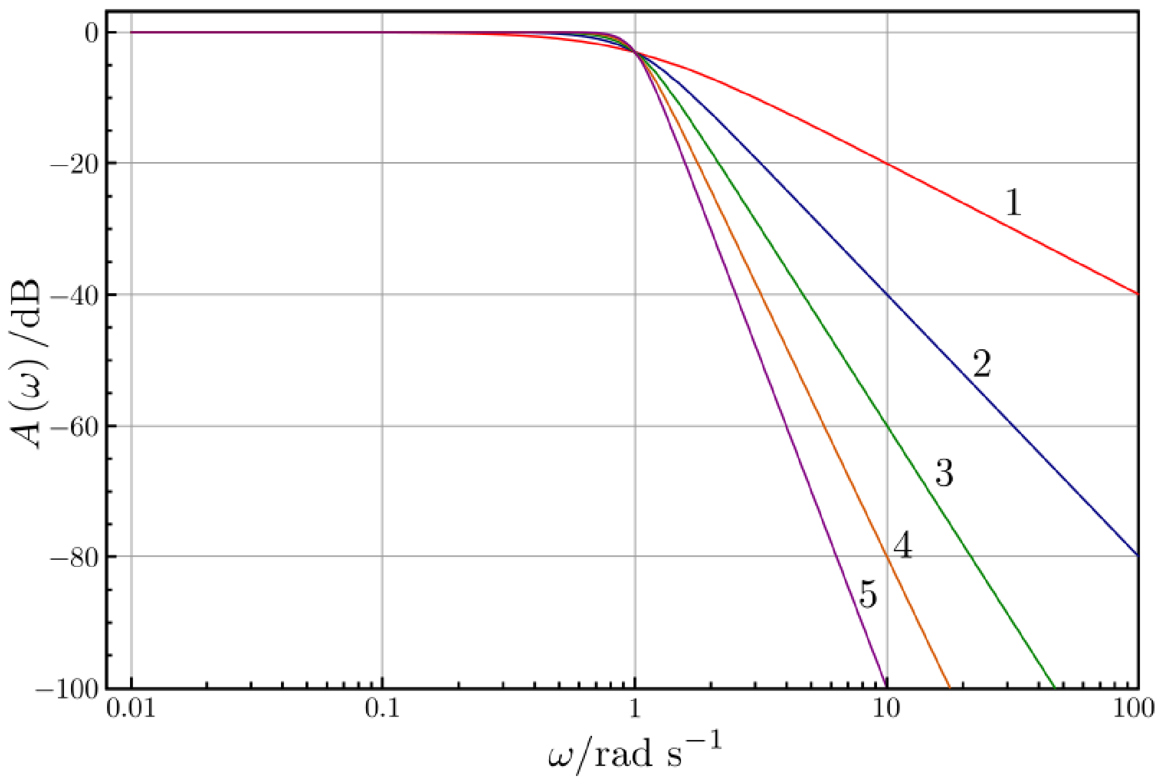
These are arranged on a circle of radius unity, symmetrical about the real *s* axis. The gain function will have three more poles on the right half plane to complete the circle.

By replacing each inductor with a capacitor and each capacitor with an inductor, a high-pass Butterworth filter is obtained.

A band-pass Butterworth filter is obtained by placing a capacitor in series with each inductor and an inductor in parallel with each capacitor to form resonant circuits. The value of each new component must be selected to resonate with the old component at the frequency of interest.

A band-stop Butterworth filter is obtained by placing a capacitor in parallel with each inductor and an inductor in series with each capacitor to form resonant circuits. The value of each new component must be selected to resonate with the old component at the frequency to be rejected.

Transfer function



Plot of the gain of Butterworth low-pass filters of orders 1 through 5, with cutoff frequency $\omega_0 = 1$. Note that the slope is 20*n* dB/decade where *n* is the filter order.

Like all filters, the typical prototype is the low-pass filter, which can be modified into a high-pass filter, or placed in series with others to form band-pass and band-stop filters, and higher order versions of these.

The gain $G(\omega)$ of an *n*-order Butterworth low pass filter is given in terms of the transfer function *H(s)* as;

$$G^2(\omega) = |H(j\omega)|^2 = \frac{{G_0}^2}{1 + \left(\frac{\omega}{\omega_c}\right)^{2n}}$$

where

- n = order of filter
- ω_c = cutoff frequency (approximately the -3dB frequency)
- G_0 is the DC gain (gain at zero frequency)

It can be seen that as n approaches infinity, the gain becomes a rectangle function and frequencies below ω_c will be passed with gain G_0, while frequencies above ω_c will be suppressed. For smaller values of n, the cutoff will be less sharp.

We wish to determine the transfer function $H(s)$ where $s = \sigma + j\omega$ (from Laplace transform). Since $H(s)H(-s)$ evaluated at $s = j\omega$ is simply equal to $|H(j\omega)|^2$, it follows that;

$$H(s)H(-s) = \frac{{G_0}^2}{1 + \left(\frac{-s^2}{\omega_c^2}\right)^n}$$

The poles of this expression occur on a circle of radius ω_c at equally spaced points. The transfer function itself will be specified by just the poles in the negative real half-plane of s. The *k-th* pole is specified by;

$$-\frac{s_k^2}{\omega_c^2} = (-1)^{\frac{1}{n}} = e^{\frac{j(2k-1)\pi}{n}} \qquad \mathrm{k} = 1, 2, 3, \ldots, \mathrm{n}$$

and hence;

$$s_k = \omega_c e^{\frac{j(2k+n-1)\pi}{2n}} \qquad \mathrm{k} = 1, 2, 3, \ldots, \mathrm{n}$$

The transfer function may be written in terms of these poles as;

$$H(s) = \frac{G_0}{\prod_{k=1}^{n}(s - s_k)/\omega_c}$$

The denominator is a Butterworth polynomial in s.

Normalized Butterworth polynomials

The Butterworth polynomials may be written in complex form as above, but are usually written with real coefficients by multiplying pole pairs which are complex conjugates, such as s_1 and s_n. The polynomials are normalized by setting $\omega_c = 1$. The normalized Butterworth polynomials then have the general form;

$$B_n(s) = \prod_{k=1}^{\frac{n}{2}} \left[s^2 - 2s\cos\left(\frac{2k+n-1}{2n}\,\pi\right) + 1\right]_{\text{for n even}}$$

$$B_n(s) = (s+1)\prod_{k=1}^{\frac{n-1}{2}} \left[s^2 - 2s\cos\left(\frac{2k+n-1}{2n}\,\pi\right) + 1\right]_{\text{for n odd}}$$

To four decimal places, they are;

n	Factors of Polynomial $B_n(s)$
1	$(s + 1)$
2	$s^2 + 1.4142s + 1$
3	$(s + 1)(s^2 + s + 1)$
4	$(s^2 + 0.7654s + 1)(s^2 + 1.8478s + 1)$
5	$(s + 1)(s^2 + 0.6180s + 1)(s^2 + 1.6180s + 1)$
6	$(s^2 + 0.5176s + 1)(s^2 + 1.4142s + 1)(s^2 + 1.9319s + 1)$
7	$(s + 1)(s^2 + 0.4450s + 1)(s^2 + 1.2470s + 1)(s^2 + 1.8019s + 1)$
8	$(s^2 + 0.3902s + 1)(s^2 + 1.1111s + 1)(s^2 + 1.6629s + 1)(s^2 + 1.9616s + 1)$

The normalized Butterworth polynomials can be used to determine the transfer function for any low-pass filter cut-off frequency ω_c, as follows

$$H(s) = \frac{G_0}{B_n(a)}, \text{ where } a = \frac{s}{\omega_c}$$

Maximal flatness

Assuming $\omega_c = 1$ and $G_0 = 1$, the derivative of the gain with respect to frequency can be shown to be;

$$\frac{dG}{d\omega} = -nG^3\omega^{2n-1}$$

which is monotonically decreasing for all ω since the gain G is always positive. The gain function of the Butterworth filter therefore has no ripple. Furthermore, the series expansion of the gain is given by;

$$G(\omega) = 1 - \frac{1}{2}\omega^{2n} + \frac{3}{8}\omega^{4n} + \ldots$$

In other words all derivatives of the gain up to but not including the $2n$-th derivative are zero, resulting in "maximal flatness". If the requirement to be monotonic is limited to the passband only and ripples are allowed in the stopband, then it is possible to design a filter of the same order, such as the inverse Chebyshev filter, that is flatter in the passband than the "maximally flat" Butterworth.

High-frequency roll-off

Again assuming $\omega_c = 1$, the slope of the log of the gain for large ω is;

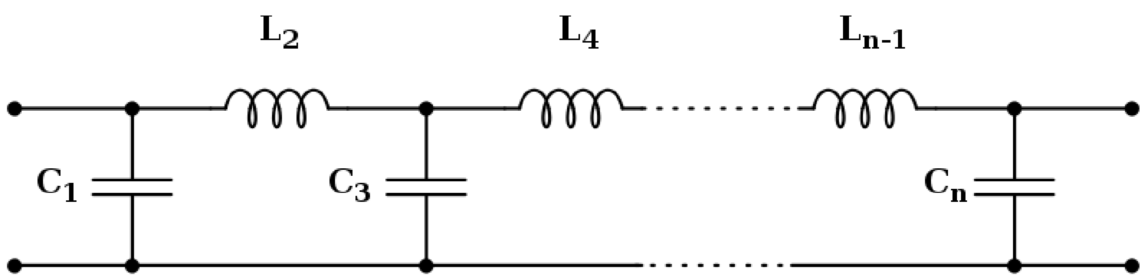
$$\lim_{\omega\to\infty} \frac{d\log(G)}{d\log(\omega)} = -n$$

In decibels, the high-frequency roll-off is therefore 20n dB/decade, or 6n dB/octave (The factor of 20 is used because the power is proportional to the square of the voltage gain.)

Filter design

There are a number of different filter topologies available to implement a linear analogue filter. The most often used topology for a passive realisation is Cauer topology and the most often used topology for an active realisation is Sallen-Key topology.

Cauer topology



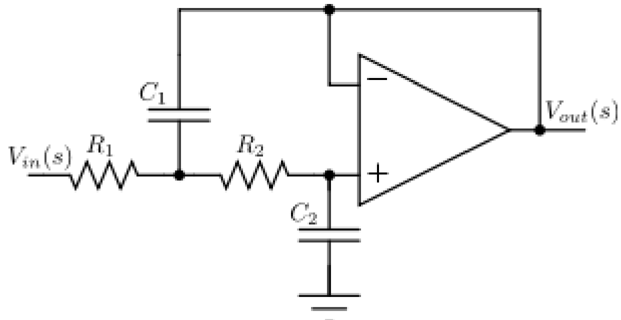
The Cauer topology uses passive components (shunt capacitors and series inductors) to implement a linear analog filter. The Butterworth filter having a given transfer function can be realised using a Cauer 1-form. The kth element is given by;

$$C_k = 2\sin\left[\frac{(2k-1)}{2n}\pi\right]; \text{ k = odd}$$

$$L_k = 2\sin\left[\frac{(2k-1)}{2n}\pi\right]; \text{ k = even}$$

The filter may start with a series inductor if desired, in which case the L_k are k odd and the C_k are k even.

Sallen–Key topology



The Sallen–Key topology uses active and passive components (noninverting buffers, usually op amps, resistors, and capacitors) to implement a linear analog filter. Each Sallen-Key stage implements a conjugate pair of poles; the overall filter is implemented by cascading all stages in series. If there is a real pole (in the case where *n* is odd), this must be implemented separately, usually as an RC circuit, and cascaded with the active stages.

For the second order Sallen-Key circuit shown to the right the transfer function is given by;

$$H(s) = \frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{1 + C_2(R_1 + R_2)s + C_1C_2R_1R_2s^2}$$

We wish the denominator to be one of the quadratic terms in a Butterworth polynomial. Assuming that $\omega_c = 1$, this will mean that;

$$C_1C_2R_1R_2 = 1$$

and;

$$C_2(R_1 + R_2) = -2\cos\left(\frac{2k+n-1}{2n}\pi\right)$$

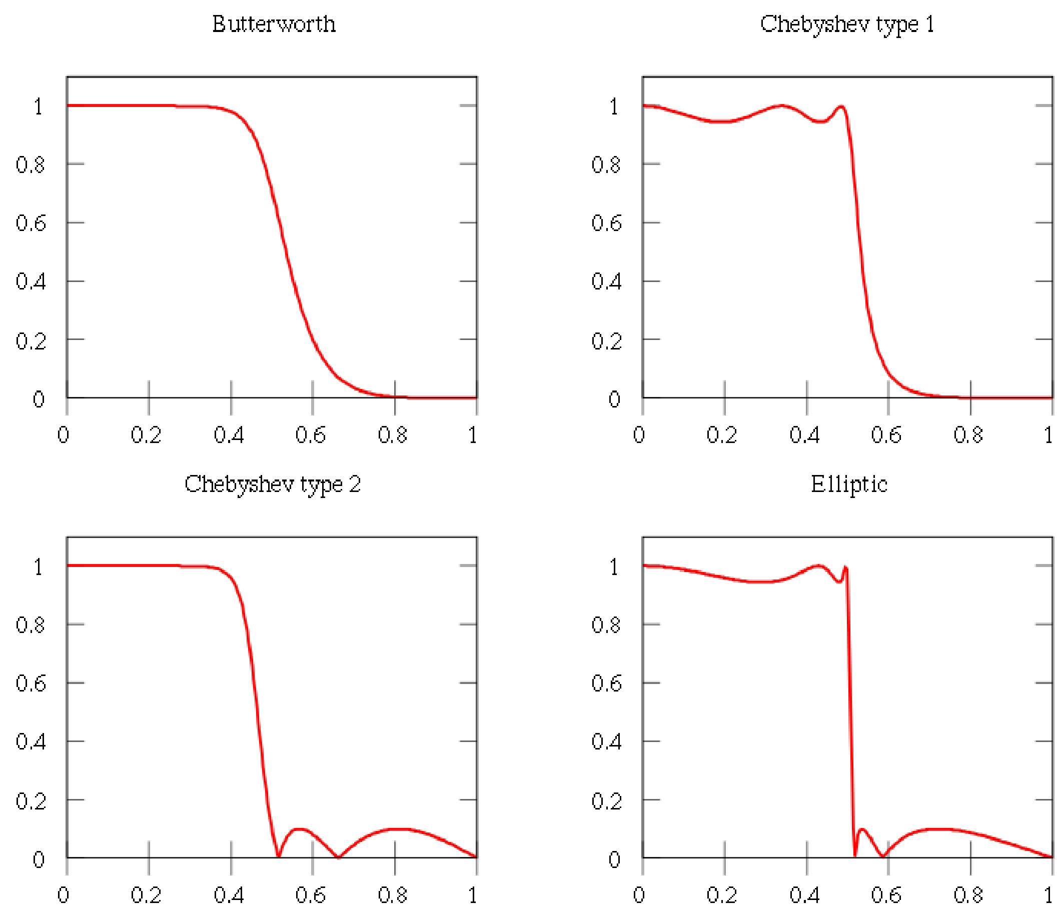
This leaves two undefined component values that may be chosen at will.

Digital implementation

Digital implementations of Butterworth and other filters are often based on the bilinear transform method or the matched Z-transform method, two different methods to discretize an analog filter design. In the case of all-pole filters such as the Butterworth, the matched Z-transform method is equivalent to the impulse invariance method. For higher orders, digital filters are sensitive to quantization errors, so they are often calculated as cascaded biquad sections, plus one first-order or third-order section for odd orders.

Comparison with other linear filters

Here is an image showing the gain of a discrete-time Butterworth filter next to other common filter types. All of these filters are fifth-order.



The Butterworth filter rolls off more slowly around the cutoff frequency than the Chebyshev filter or the Elliptic filter, but without ripple.

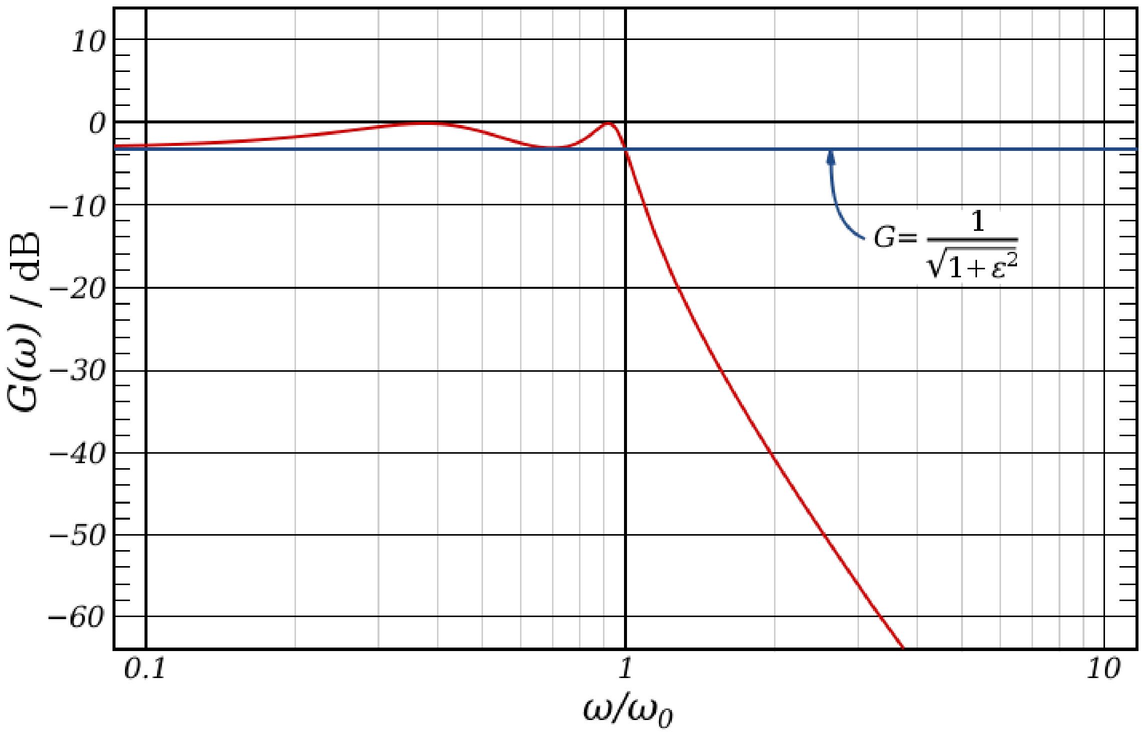
Chapter- 8

Chebyshev Filter

Chebyshev filters are analog or digital filters having a steeper roll-off and more passband ripple (type I) or stopband ripple (type II) than Butterworth filters. Chebyshev filters have the property that they minimize the error between the idealized and the actual filter characteristic over the range of the filter, but with ripples in the passband. This type of filter is named in honor of Pafnuty Chebyshev because their mathematical characteristics are derived from Chebyshev polynomials.

Because of the passband ripple inherent in Chebyshev filters, filters which have a smoother response in the passband but a more irregular response in the stopband are preferred for some applications.

Type I Chebyshev filters



The frequency response of a fourth-order type I Chebyshev low-pass filter with $\varepsilon = 1$

These are the most common Chebyshev filters. The gain (or amplitude) response as a function of angular frequency ω of the *n*th order low pass filter is

$$G_n(\omega) = |H_n(j\omega)| = \frac{1}{\sqrt{1+\varepsilon^2 T_n^2\left(\frac{\omega}{\omega_0}\right)}}$$

where ε is the ripple factor, ω_0 is the cutoff frequency and $T_n()$ is a Chebyshev polynomial of the *n*th order.

The passband exhibits equiripple behavior, with the ripple determined by the ripple factor ε. In the passband, the Chebyshev polynomial alternates between 0 and 1 so the filter gain will alternate between maxima at $G = 1$ and minima at $G = 1/\sqrt{1+\varepsilon^2}$. At the cutoff frequency ω_0 the gain again has the value $1/\sqrt{1+\varepsilon^2}$ but continues to drop into the stop band as the frequency increases. This behavior is shown in the diagram on the right. (*note*: the common definition of the cutoff frequency to −3 dB does *not* hold for Chebyshev filters!)

The order of a Chebyshev filter is equal to the number of reactive components (for example, inductors) needed to realize the filter using analog electronics.

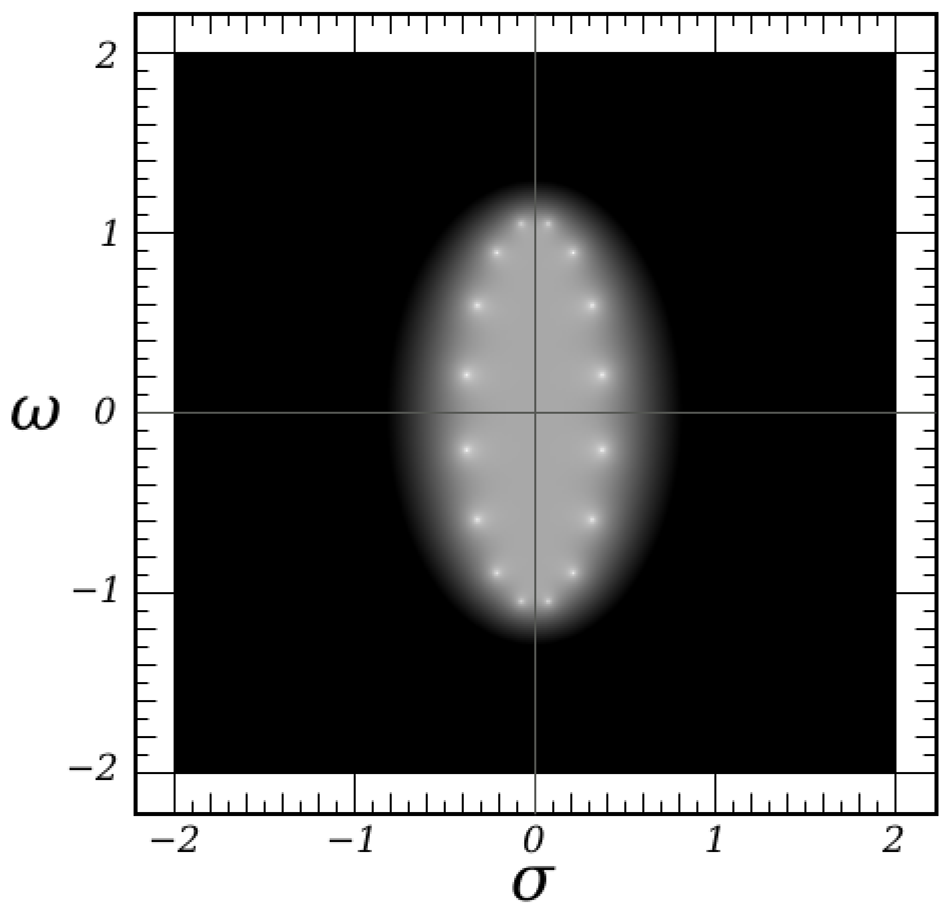
The ripple is often given in dB:

Ripple in dB = $20\log_{10}\frac{1}{\sqrt{1+\varepsilon^2}}$

so that a ripple amplitude of 3 dB results from $\varepsilon = 1.$

An even steeper roll-off can be obtained if we allow for ripple in the stop band, by allowing zeroes on the $j\omega$-axis in the complex plane. This will however result in less suppression in the stop band. The result is called an elliptic filter, also known as Cauer filters.

Poles and zeroes



Log of the absolute value of the gain of an 8th order Chebyshev type I filter in complex frequency space ($s = \sigma + j\omega$) with $\varepsilon = 0.1$ and $\omega_0 = 1$. The white spots are poles and are arranged on an ellipse with a semi-axis of 0.3836... in σ and 1.071... in ω. The transfer function poles are those poles in the left half plane. Black corresponds to a gain of 0.05 or less, white corresponds to a gain of 20 or more.

For simplicity, assume that the cutoff frequency is equal to unity. The poles (ω_{pm}) of the gain of the Chebyshev filter will be the zeroes of the denominator of the gain. Using the complex frequency *s*:

$$1 + \varepsilon^2 T_n^2(-js) = 0.$$

Defining $-js = \cos(\theta)$ and using the trigonometric definition of the Chebyshev polynomials yields:

$$1 + \varepsilon^2 T_n^2(\cos(\theta)) = 1 + \varepsilon^2 \cos^2(n\theta) = 0.$$

Solving for θ

$$\theta = \frac{1}{n} \arccos\left(\frac{\pm j}{\varepsilon}\right) + \frac{m\pi}{n}$$

where the multiple values of the arc cosine function are made explicit using the integer index *m*. The poles of the Chebyshev gain function are then:

$$\begin{aligned} s_{pm} &= j\cos(\theta) \\ &= j\cos\left(\frac{1}{n}\arccos\left(\frac{\pm j}{\varepsilon}\right) + \frac{m\pi}{n}\right) \end{aligned}$$

Using the properties of the trigonometric and hyperbolic functions, this may be written in explicitly complex form:

$$\begin{aligned} s_{pm}^{\pm} &= \pm \sinh\left(\frac{1}{n}\operatorname{arsinh}\left(\frac{1}{\varepsilon}\right)\right)\sin(\theta_m) \\ &+ j\cosh\left(\frac{1}{n}\operatorname{arsinh}\left(\frac{1}{\varepsilon}\right)\right)\cos(\theta_m) \end{aligned}$$

where $m = 1, 2, \ldots, n$ and

$$\theta_m = \frac{\pi}{2}\,\frac{2m-1}{n}$$

This may be viewed as an equation parametric in θ_n and it demonstrates that the poles lie on an ellipse in s-space centered at $s = 0$ with a real semi-axis of length $\sinh(\operatorname{arsinh}(1/\varepsilon)/n)$ and an imaginary semi-axis of length of $\cosh(\operatorname{arsinh}(1/\varepsilon)/n)$.

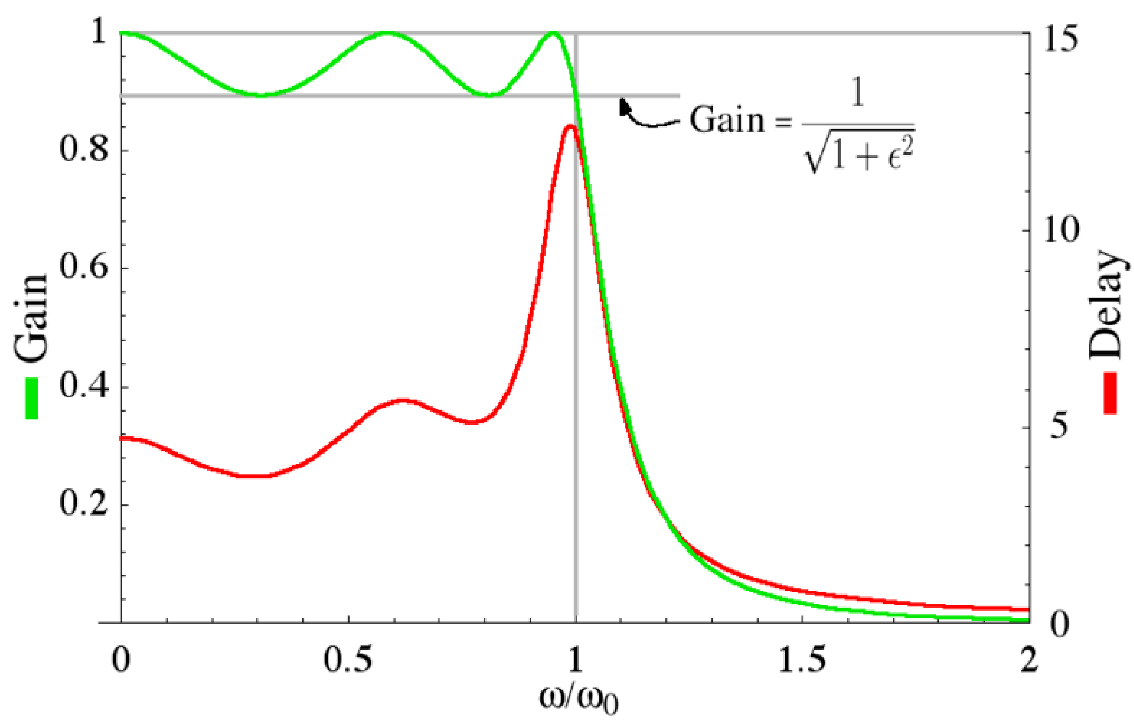
The transfer function

The above expression yields the poles of the gain G. For each complex pole, there is another which is the complex conjugate, and for each conjugate pair there are two more that are the negatives of the pair. The transfer function must be stable, so that its poles will be those of the gain that have negative real parts and therefore lie in the left half plane of complex frequency space. The transfer function is then given by

$$H(s) = \frac{1}{2^{n-1}\varepsilon} \prod_{m=1}^{n} \frac{1}{(s - s_{pm}^{-})}$$

where s_{pm}^{-} are only those poles with a negative sign in front of the real term in the above equation for the poles.

The group delay



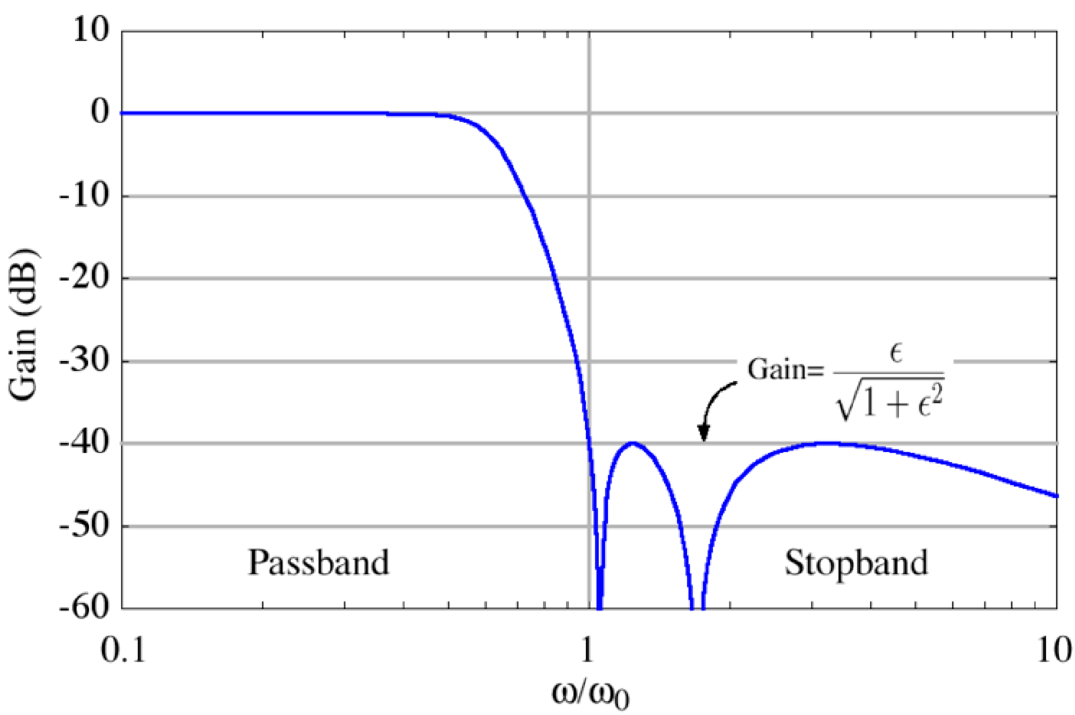
Gain and group delay of a fifth-order type I Chebyshev filter with ε = 0.5.

The group delay is defined as the derivative of the phase with respect to angular frequency and is a measure of the distortion in the signal introduced by phase differences for different frequencies.

$$\tau_g = -\frac{d}{d\omega}\arg(H(j\omega))$$

The gain and the group delay for a fifth order type I Chebyshev filter with ε=0.5 are plotted in the graph on the left. It can be seen that there are ripples in the gain and the group delay in the passband but not in the stop band.

Type II Chebyshev filters



The frequency response of a fifth-order type II Chebyshev low-pass filter with $\varepsilon = 0.01$

Also known as inverse Chebyshev, this type is less common because it does not roll off as fast as type I, and requires more components. It has no ripple in the passband, but does have equiripple in the stopband. The gain is:

$$G_n(\omega, \omega_0) = \frac{1}{\sqrt{1 + \frac{1}{\varepsilon^2 T_n^2(\omega_0/\omega)}}}.$$

In the stop band, the Chebyshev polynomial will oscillate between 0 and 1 so that the gain will oscillate between zero and

$$\frac{1}{\sqrt{1+\frac{1}{\varepsilon^2}}}$$

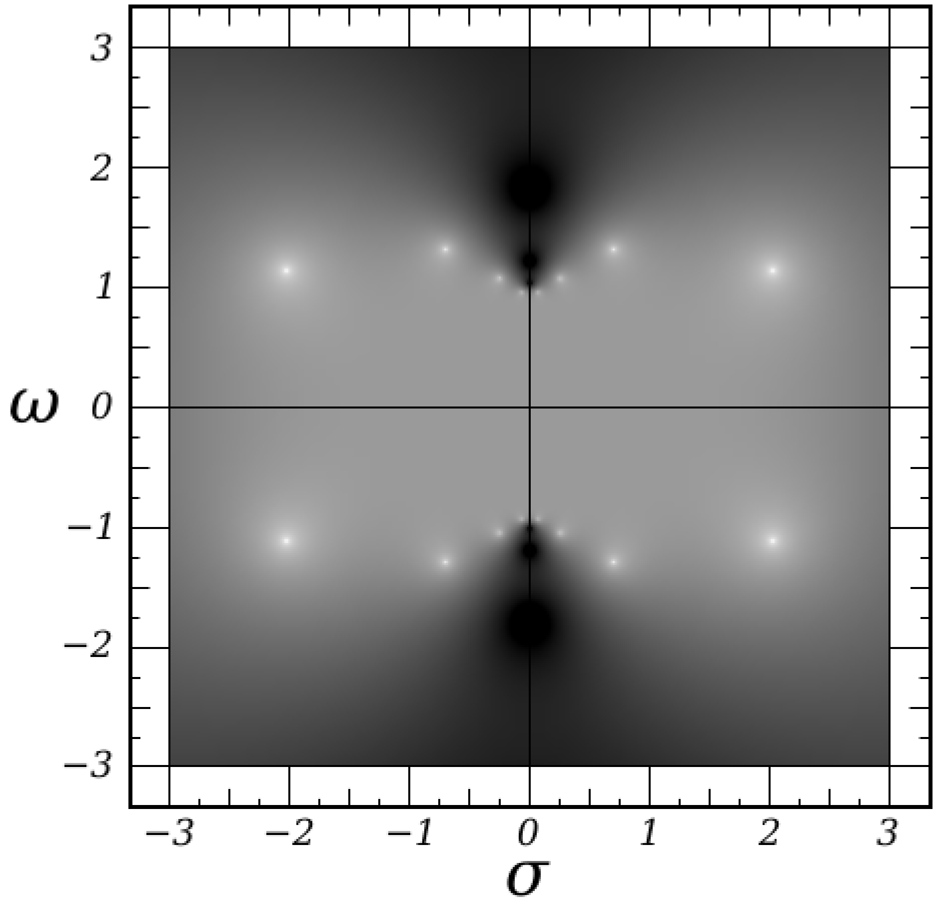
and the smallest frequency at which this maximum is attained will be the cutoff frequency ω_0. The parameter ε is thus related to the stopband attenuation γ in decibels by:

$$\varepsilon = \frac{1}{\sqrt{10^{0.1\gamma}-1}}.$$

For a stopband attenuation of 5dB, $\varepsilon = 0.6801$; for an attenuation of 10dB, $\varepsilon = 0.3333$. The frequency $f_C = \omega_C/2\pi$ is the cutoff frequency. The 3dB frequency f_H is related to f_C by:

$$f_H = \frac{f_C}{\cosh\left(\frac{1}{n}\cosh^{-1}\frac{1}{\varepsilon}\right)}.$$

Poles and zeroes



Log of the absolute value of the gain of an 8th order Chebyshev type II filter in complex frequency space ($s=\sigma+j\omega$) with $\varepsilon = 0.1$ and $\omega_0 = 1$. The white spots are poles and the black spots are zeroes. All 16 poles are shown. Each zero has multiplicity of two, and 12 zeroes are shown and four are located outside the picture, two on the positive ω axis, and two on the negative. The poles of the transfer function will be poles on the left half plane and the zeroes of the transfer function will be the zeroes, but with multiplicity 1. Black corresponds to a gain of 0.05 or less, white corresponds to a gain of 20 or more.

Again, assuming that the cutoff frequency is equal to unity, the poles (ω_{pm}) of the gain of the Chebyshev filter will be the zeroes of the denominator of the gain:

$$1+\varepsilon^2 T_n^2(-1/js_{pm})=0$$

The poles of gain of the type II Chebyshev filter will be the inverse of the poles of the type I filter:

$$\frac{1}{s_{pm}^{\pm}} = \pm\sinh\left(\frac{1}{n}\mathrm{arsinh}\left(\frac{1}{\varepsilon}\right)\right)\sin(\theta_m)$$
$$+j\cosh\left(\frac{1}{n}\mathrm{arsinh}\left(\frac{1}{\varepsilon}\right)\right)\cos(\theta_m)$$

where $m = 1, 2, ..., n$. The zeroes (ω_{zm}) of the type II Chebyshev filter will be the zeroes of the numerator of the gain:

$$\varepsilon^2 T_n^2(-1/js_{zm}) = 0.$$

The zeroes of the type II Chebyshev filter will thus be the inverse of the zeroes of the Chebyshev polynomial.

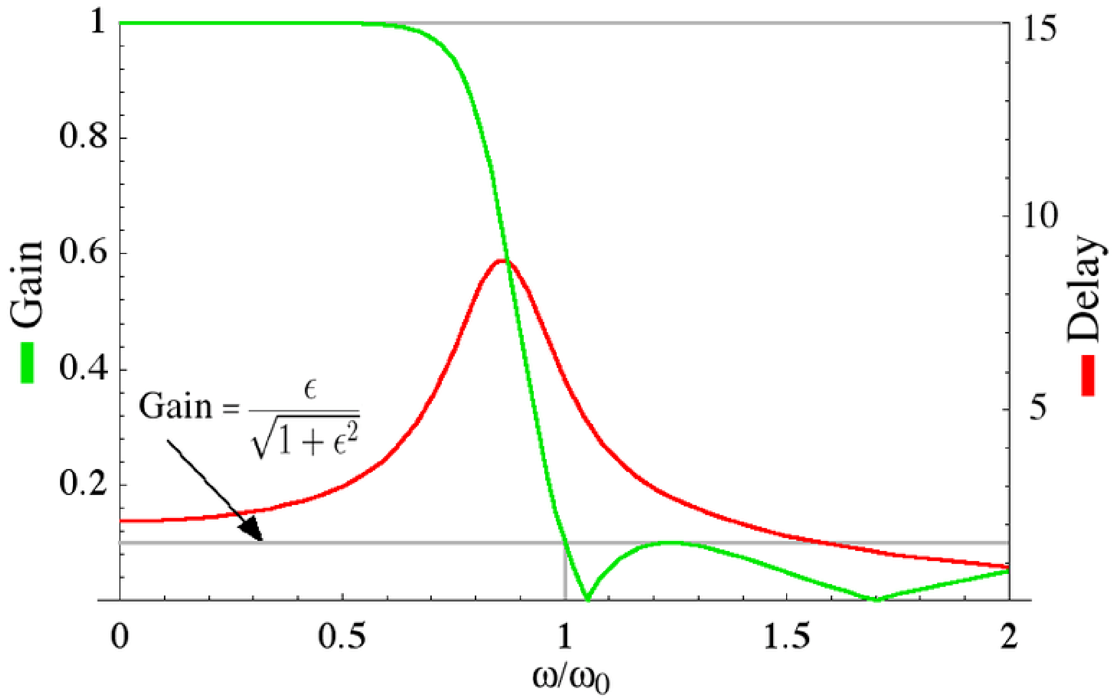
$$1/s_{zm} = -j\cos\left(\frac{\pi}{2}\,\frac{2m-1}{n}\right)$$

for $m = 1, 2, ..., n$.

The transfer function

The transfer function will be given by the poles in the left half plane of the gain function, and will have the same zeroes but these zeroes will be single rather than double zeroes.

The group delay



Gain and group delay of a fifth-order type II Chebyshev filter with ε = 0.1.

The gain and the group delay for a fifth order type II Chebyshev filter with ε=0.1 are plotted in the graph on the left. It can be seen that there are ripples in the gain in the stop band but not in the pass band.

Implementation

Cauer topology

A passive LC Chebyshev low-pass filter may be realized using a Cauer topology. Inductor or capacitor values of a nth-order Chebyshev filter may be calculated from the following equations:

$$G_1 = \frac{2A_1 \cosh(f_H)}{Y}$$
$$G_k = \frac{4A_{k-1}A_k \cosh^2(f_H)}{B_{k-1}G_{k-1}}, \qquad k = 1, 2, 3, \ldots, n$$

G_1, G_k are the capacitor or inductor element values.

f_H, the 3 dB frequency is calculated with: $$f_H = f_C \cosh\left(\frac{1}{n}\cosh^{-1}\frac{1}{\varepsilon}\right)$$

The coefficients A, Y, β, A_k, and B_k may be calculated from the following equations:

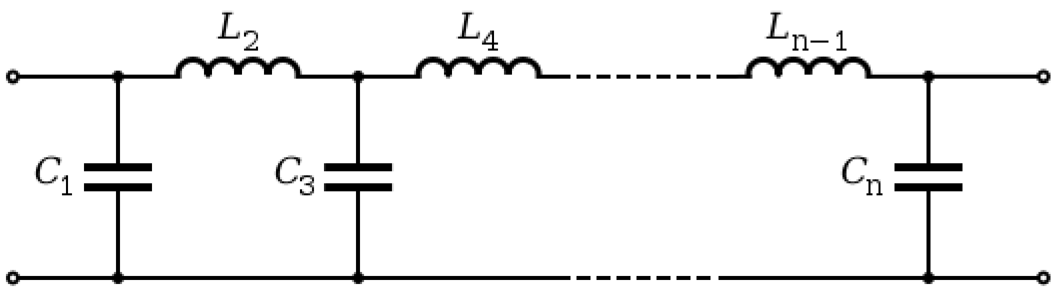
$$Y = \sinh\left(\frac{\beta}{2n}\right)$$

$$\beta = \ln\left(\coth(R_{db}/17.37)\right)$$

$$A_k = \sin\frac{(2k-1)\pi}{2n}, \qquad k = 1, 2, 3, \ldots, n$$

$$B_k = Y^2 + \sin^2\left(\frac{k\pi}{n}\right), \qquad k = 1, 2, 3, \ldots, n$$

where R_{dB} is the passband ripple in decibels.



The calculated G_k values may then be converted into shunt capacitors and top inductors as shown on the right, or they may be converted into top capacitors and shunt inductors.

- For example, $C_{1\text{ shunt}} = G_1$, $L_{2\text{ top}} = G_2$, ...
- or $L_{1\text{ shunt}} = G_1$, $C_{1\text{ top}} = G_2$, ...

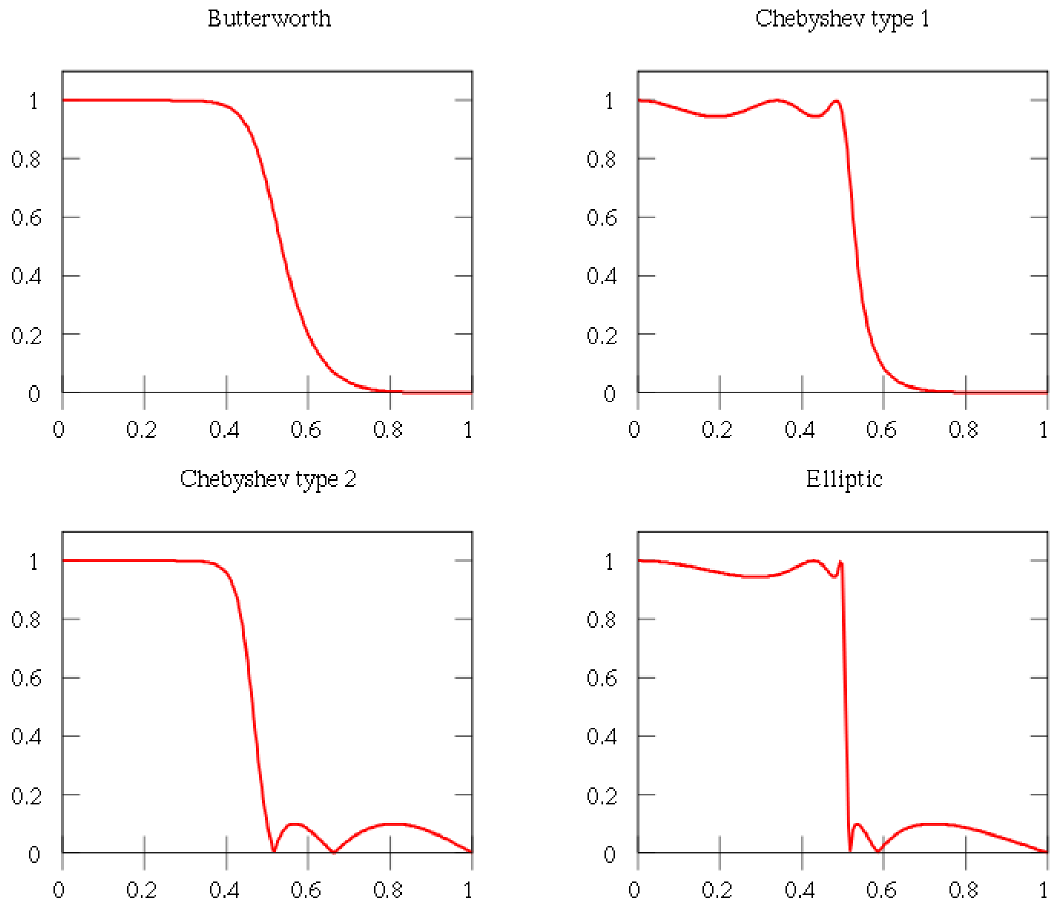
The resulting circuit is a normalized low-pass filter. Using frequency transformations and impedance scaling, the normalized low-pass filter may be transformed into high-pass, band-pass, and band-stop filters of any desired cutoff frequency or bandwidth.

Digital

As with most analog filters, the Chebyshev may be converted to a digital (discrete-time) recursive form via the bilinear transform. However, as digital filters have a finite bandwidth, the response shape of the transformed Chebyshev will be warped. Alternatively, the Matched Z-transform method may be used, which does not warp the response.

Comparison with other linear filters

Here is an image showing the Chebyshev filters next to other common kind of filters obtained with the same number of coefficients (all filters are fifth order):



As is clear from the image, Chebyshev filters are sharper than the Butterworth filter; they are not as sharp as the elliptic one, but they show fewer ripples over the bandwidth.

Chapter- 9

High-Pass Filter

A **high-pass filter**, or HPF, is an LTI filter that passes high frequencies well but attenuates (i.e., reduces the amplitude of) frequencies lower than the filter's cutoff frequency. The actual amount of attenuation for each frequency is a design parameter of the filter. It is sometimes called a **low-cut filter** or **bass-cut filter**.

First-order continuous-time implementation

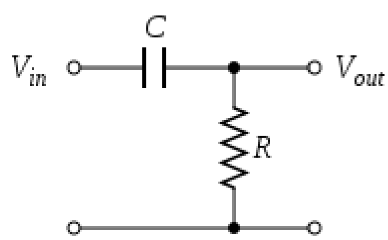


Figure 1: A passive, analog, first-order high-pass filter, realized by an RC circuit

The simple first-order electronic high-pass filter shown in Figure 1 is implemented by placing an input voltage across the series combination of a capacitor and a resistor and using the voltage across the resistor as an output. The product of the resistance and capacitance ($R \times C$) is the time constant (τ); it is inversely proportional to the cutoff frequency f_c, at which the output power is half the input power. That is,

$$f_c = \frac{1}{2\pi\tau} = \frac{1}{2\pi RC},$$

where f_c is in hertz, τ is in seconds, R is in ohms, and C is in farads.

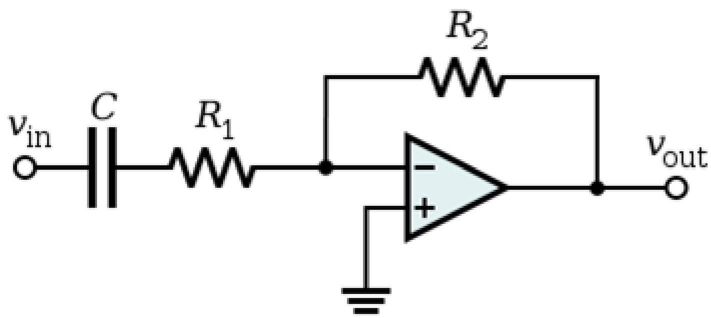


Figure 2: An active high-pass filter

Figure 2 shows an active electronic implementation of a first-order high-pass filter using an operational amplifier. In this case, the filter has a passband gain of $-R_2/R_1$ and has a corner frequency of

$$f_c = \frac{1}{2\pi\tau} = \frac{1}{2\pi R_1 C},$$

Because this filter is active, it may have non-unity passband gain. That is, high-frequency signals are inverted and amplified by R_2/R_1.

Discrete-time realization

Discrete-time high-pass filters can also be designed. However, a simple example comes from the conversion of the continuous-time high-pass filter above to a discrete-time realization. That is, the continuous-time behavior can be discretized.

From the circuit in Figure 1 above, according to Kirchoff's Laws and the definition of capacitance:

$$\begin{cases} V_{\text{out}}(t) = I(t)\,R & \text{(V)} \\ Q_c(t) = C\,\left(V_{\text{in}}(t) - V_{\text{out}}(t)\right) & \text{(Q)} \\ I(t) = \frac{\mathrm{d}Q_c}{\mathrm{d}t} & \text{(I)} \end{cases}$$

where $Q_c(t)$ is the charge stored in the capacitor at time t. Substituting Equation (Q) into Equation (I) and then Equation (I) into Equation (V) gives:

$$V_{\text{out}}(t) = \overbrace{C\,\left(\frac{\mathrm{d}V_{\text{in}}}{\mathrm{d}t} - \frac{\mathrm{d}V_{\text{out}}}{\mathrm{d}t}\right)}^{I(t)}\,R = RC\,\left(\frac{\mathrm{d}V_{\text{in}}}{\mathrm{d}t} - \frac{\mathrm{d}V_{\text{out}}}{\mathrm{d}t}\right)$$

This equation can be discretized. For simplicity, assume that samples of the input and output are taken at evenly-spaced points in time separated by Δ_T time. Let the samples of V_{in} be represented by the sequence $(x_1, x_2, \ldots, x_n)$, and let V_{out} be represented by the sequence $(y_1, y_2, \ldots, y_n)$which correspond to the same points in time. Making these substitutions:

$$y_i = RC\left(\frac{x_i - x_{i-1}}{\Delta_T} - \frac{y_i - y_{i-1}}{\Delta_T}\right)$$

And rearranging terms gives the recurrence relation

$$y_i = \overbrace{\frac{RC}{RC + \Delta_T} y_{i-1}}^{\text{Decaying contribution from prior inputs}} + \overbrace{\frac{RC}{RC + \Delta_T}(x_i - x_{i-1})}^{\text{Contribution from change in input}}$$

That is, this discrete-time implementation of a simple continuous-time RC high-pass filter is

$$y_i = \alpha y_{i-1} + \alpha(x_i - x_{i-1}) \qquad \text{where} \qquad \alpha \triangleq \frac{RC}{RC + \Delta_T}$$

By definition, $0 \leq \alpha \leq 1$. The expression for parameter α yields the equivalent time constant *RC* in terms of the sampling period Δ_T and α:

$$RC = \Delta_T\left(\frac{\alpha}{1 - \alpha}\right)$$

If α = 0.5, then the *RC* time constant equal to the sampling period. If $\alpha \ll 0.5$, then *RC* is significantly smaller than the sampling interval, and $RC \approx \alpha\Delta_T$.

Algorithmic implementation

The filter recurrence relation provides a way to determine the output samples in terms of the input samples and the preceding output. The following pseudocode algorithm will simulate the effect of a high-pass filter on a series of digital samples:

```
// Return RC high-pass filter output samples, given input samples,
// time interval dt, and time constant RC
function highpass(real[0..n] x, real dt, real RC)
  var real[0..n] y
  var real α := RC / (RC + dt)
  y[0] := x[0]
  for i from 1 to n
    y[i] := α * y[i-1] + α * (x[i] - x[i-1])
  return y
```

The loop which calculates each of the *n* outputs can be refactored into the equivalent:

```
for i from 1 to n
  y[i] := α * (y[i-1] + x[i] - x[i-1])
```

However, the earlier form shows how the parameter α changes the impact of the prior output `y[i-1]` and current *change* in input `(x[i] - x[i-1])`. In particular,

- A large α implies that the output will decay very slowly but will also be strongly influenced by even small changes in input. By the relationship between parameter α and time constant *RC* above, a large α corresponds to a large *RC* and therefore a low corner frequency of the filter. Hence, this case corresponds to a high-pass filter with a very narrow stop band. Because it is excited by small changes and tends to hold its prior output values for a long time, it can pass relatively low frequencies. However, a constant input (i.e., an input with `(x[i] - x[i-1])=0`) will always decay to zero, as would be expected with a high-pass filter with a large *RC*.
- A small α implies that the output will decay quickly and will require large changes in the input (i.e., `(x[i] - x[i-1])` is large) to cause the output to change much. By the relationship between parameter α and time constant *RC* above, a small α corresponds to a small *RC* and therefore a high corner frequency of the filter. Hence, this case corresponds to a high-pass filter with a very wide stop band. Because it requires large (i.e., fast) changes and tends to quickly forget its prior output values, it can only pass relatively high frequencies, as would be expected with a high-pass filter with a small *RC*.

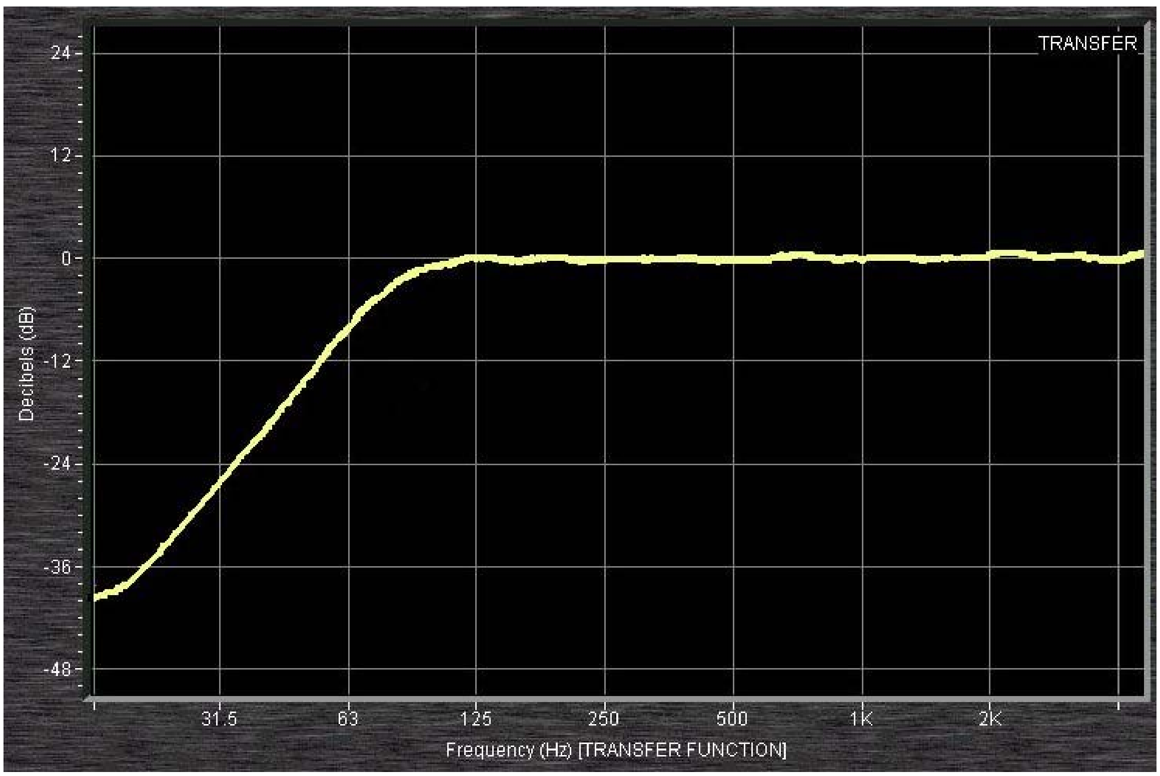
Applications

Audio

High-pass filters have many applications. They are used as part of an audio crossover to direct high frequencies to a tweeter while attenuating bass signals which could interfere with, or damage, the speaker. When such a filter is built into a loudspeaker cabinet it is normally a passive filter that also includes a low-pass filter for the woofer and so often employs both a capacitor and inductor (although very simple high-pass filters for tweeters can consist of a series capacitor and nothing else). An alternative, which provides good quality sound without inductors (which are prone to parasitic coupling, are expensive, and may have significant internal resistance) is to employ bi-amplification with active RC filters or active digital filters with separate power amplifiers for each loudspeaker. Such low-current and low-voltage line level crossovers are called active crossovers.

Rumble filters are high-pass filters applied to the removal of unwanted sounds near to the lower end of the audible range or below. For example, noises (e.g., footsteps, or motor noises from record players and tape decks) may be removed because they are undesired or may overload the RIAA equalization circuit of the preamp.

High-pass filters are also used for AC coupling at the inputs of many audio amplifiers, for preventing the amplification of DC currents which may harm the amplifier, rob the amplifier of headroom, and generate waste heat at the loudspeakers voice coil. One amplifier, the professional audio model DC300 made by Crown International beginning in the 1960s, did not have high-pass filtering at all, and could be used to amplify the DC signal of a common 9-volt battery at the input to supply 18 volts DC in an emergency for mixing console power. However, that model's basic design has been superseded by newer designs such as the Crown Macro-Tech series developed in the late 1980s which included 10 Hz high-pass filtering on the inputs and switchable 35 Hz high-pass filtering on the outputs. Another example is the QSC Audio PLX amplifier series which includes an internal 5 Hz high-pass filter which is applied to the inputs whenever the optional 50 and 30 Hz high-pass filters are turned off.



A 75 Hz "low cut" filter from an input channel of a Mackie 1402 mixing console as measured by Smaart software. This high-pass filter has a slope of 18 dB per octave.

Mixing consoles often include high-pass filtering at each channel strip. Some models have fixed-slope, fixed-frequency high-pass filters at 80 or 100 Hz that can be engaged; other models have 'sweepable HPF'—a high-pass filter of fixed slope that can be set within a specified frequency range, such as from 20 to 400 Hz on the Midas Heritage 3000, or 20 to 20,000 Hz on the Yamaha M7CL digital mixing console. Veteran systems engineer and live sound mixer Bruce Main recommends that high-pass filters be engaged for most mixer input sources, except for those such as kick drum, bass guitar and piano, sources which will have useful low frequency sounds. Main writes that DI unit inputs (as opposed to microphone inputs) do not need high-pass filtering as they are not subject to modulation by low-frequency stage wash—low frequency sounds coming from the subwoofers or the public address system and wrapping around to the stage. Main indicates that high-pass filters are commonly used for directional microphones which have a proximity effect—a low-frequency boost for very close sources. This low frequency boost commonly causes problems up to 200 or 300 Hz, but Main notes that he has seen microphones that benefit from a 500 Hz HPF setting on the console.

Image

High-pass and low-pass filters are also used in digital image processing to perform image modifications, enhancements, noise reduction, etc., using designs done in either the spatial domain or the frequency domain. The unsharp masking, or sharpening, operation used in image editing software is a high-boost filter, a generalization of high-pass.

Chapter- 10

Low-Pass Filter

A **low-pass filter** is a filter that passes low-frequency signals but attenuates (reduces the amplitude of) signals with frequencies higher than the cutoff frequency. The actual amount of attenuation for each frequency varies from filter to filter. It is sometimes called a **high-cut filter**, or **treble cut filter** when used in audio applications. A low-pass filter is the opposite of a high-pass filter, and a band-pass filter is a combination of a low-pass and a high-pass.

Low-pass filters exist in many different forms, including electronic circuits (such as a *hiss filter* used in audio), digital filters for smoothing sets of data, acoustic barriers, blurring of images, and so on. The moving average operation used in fields such as finance is a particular kind of low-pass filter, and can be analyzed with the same signal processing techniques as are used for other low-pass filters. Low-pass filters provide a smoother form of a signal, removing the short-term fluctuations, and leaving the longer-term trend.

Examples of low-pass filters

Acoustic

A stiff physical barrier tends to reflect higher sound frequencies, and so acts as a low-pass filter for transmitting sound. When music is playing in another room, the low notes are easily heard, while the high notes are attenuated.

Electronic

In an electronic low-pass RC filter for voltage signals, high frequencies contained in the input signal are attenuated but the filter has little attenuation below its cutoff frequency which is determined by its RC time constant.

For current signals, a similar circuit using a resistor and capacitor in parallel works in a similar manner.

Electronic low-pass filters are used to drive subwoofers and other types of loudspeakers, to block high pitches that they can't efficiently broadcast.

Radio transmitters use low-pass filters to block harmonic emissions which might cause interference with other communications.

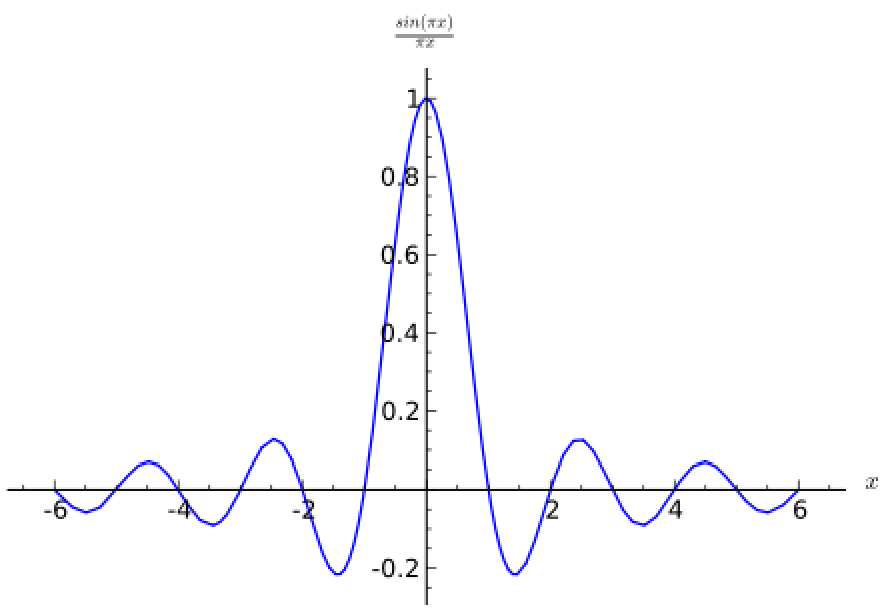
The tone knob found on many electric guitars is a low-pass filter used to reduce the amount of treble in the sound.

An integrator is another example of a single time constant low-pass filter.

Telephone lines fitted with DSL splitters use low-pass and high-pass filters to separate DSL and POTS signals sharing the same pair of wires.

Low-pass filters also play a significant role in the sculpting of sound for electronic music as created by analogue synthesisers.

Ideal and real filters



The sinc function, the impulse response of an ideal low-pass filter.

An ideal low-pass filter completely eliminates all frequencies above the cutoff frequency while passing those below unchanged: its frequency response is a rectangular function, and is a brick-wall filter. The transition region present in practical filters does not exist in an ideal filter. An ideal low-pass filter can be realized mathematically (theoretically) by multiplying a signal by the rectangular function in the frequency domain or, equivalently, convolution with its impulse response, a sinc function, in the time domain.

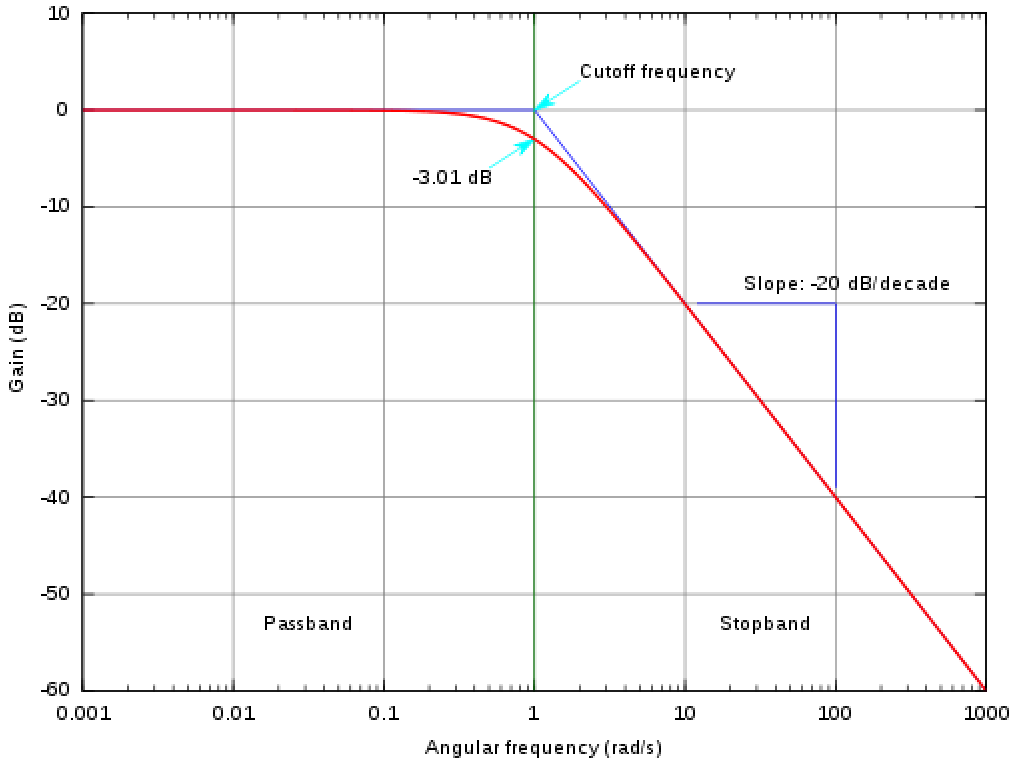
However, the ideal filter is impossible to realize without also having signals of infinite extent in time, and so generally needs to be approximated for real ongoing signals, because the sinc function's support region extends to all past and future times. The filter would therefore need to have infinite delay, or knowledge of the infinite future and past, in order to perform the convolution. It is effectively realizable for pre-recorded digital signals by assuming extensions of zero into the past and future, or more typically by making the signal repetitive and using Fourier analysis.

Real filters for real-time applications approximate the ideal filter by truncating and windowing the infinite impulse response to make a finite impulse response; applying that filter requires delaying the signal for a moderate period of time, allowing the computation to "see" a little bit into the future. This delay is manifested as phase shift. Greater accuracy in approximation requires a longer delay.

An ideal low-pass filter results in ringing artifacts via the Gibbs phenomenon. These can be reduced or worsened by choice of windowing function, and the design and choice of real filters involves understanding and minimizing these artifacts. For example, "simple truncation [of sinc] causes severe ringing artifacts," in signal reconstruction, and to reduce these artifacts one uses window functions "which drop off more smoothly at the edges."

The Whittaker–Shannon interpolation formula describes how to use a perfect low-pass filter to reconstruct a continuous signal from a sampled digital signal. Real digital-to-analog converters use real filter approximations.

Continuous-time low-pass filters



The gain-magnitude frequency response of a first-order (one-pole) low-pass filter. *Power gain* is shown in decibels (i.e., a 3 dB decline reflects an additional half-power attenuation). Angular frequency is shown on a logarithmic scale in units of radians per second.

There are many different types of filter circuits, with different responses to changing frequency. The frequency response of a filter is generally represented using a Bode plot, and the filter is characterized by its cutoff frequency and rate of frequency rolloff. In all cases, at the *cutoff frequency,* the filter attenuates the input power by half or 3 dB. So the **order** of the filter determines the amount of additional attenuation for frequencies higher than the cutoff frequency.

- A **first-order filter**, for example, will reduce the signal amplitude by half (so power reduces by 6 dB) every time the frequency doubles (goes up one octave); more precisely, the power rolloff approaches 20 dB per decade in the limit of high frequency. The magnitude Bode plot for a first-order filter looks like a horizontal line below the cutoff frequency, and a diagonal line above the cutoff frequency. There is also a "knee curve" at the boundary between the two, which smoothly transitions between the two straight line regions. If the transfer function of a first-order low-pass filter has a zero as well as a pole, the Bode plot will flatten out again, at some maximum attenuation of high frequencies; such an effect is caused for example by a little bit of the input leaking around the one-pole filter; this one-pole–one-zero filter is still a first-order low-pass.

- A **second-order filter** attenuates higher frequencies more steeply. The Bode plot for this type of filter resembles that of a first-order filter, except that it falls off more quickly. For example, a second-order Butterworth filter will reduce the signal amplitude to one fourth its original level every time the frequency doubles (so power decreases by 12 dB per octave, or 40 dB per decade). Other all-pole second-order filters may roll off at different rates initially depending on their Q factor, but approach the same final rate of 12 dB per octave; as with the first-order filters, zeroes in the transfer function can change the high-frequency asymptote.

- Third- and higher-order filters are defined similarly. In general, the final rate of power rolloff for an order-*n* all-pole filter is 6*n* dB per octave (i.e., 20*n* dB per decade).

On any Butterworth filter, if one extends the horizontal line to the right and the diagonal line to the upper-left (the asymptotes of the function), they will intersect at exactly the "cutoff frequency". The frequency response at the cutoff frequency in a first-order filter is 3 dB below the horizontal line. The various types of filters – Butterworth filter, Chebyshev filter, Bessel filter, etc. – all have different-looking "knee curves". Many second-order filters are designed to have "peaking" or resonance, causing their frequency response at the cutoff frequency to be *above* the horizontal line.

The meanings of 'low' and 'high' – that is, the cutoff frequency – depend on the characteristics of the filter. The term "low-pass filter" merely refers to the shape of the filter's response; a high-pass filter could be built that cuts off at a lower frequency than any low-pass filter – it is their responses that set them apart. Electronic circuits can be devised for any desired frequency range, right up through microwave frequencies (above 1 GHz) and higher.

Laplace notation

Continuous-time filters can also be described in terms of the Laplace transform of their impulse response in a way that allows all of the characteristics of the filter to be easily analyzed by considering the pattern of poles and zeros of the Laplace transform in the complex plane (in discrete time, one can similarly consider the Z-transform of the impulse response).

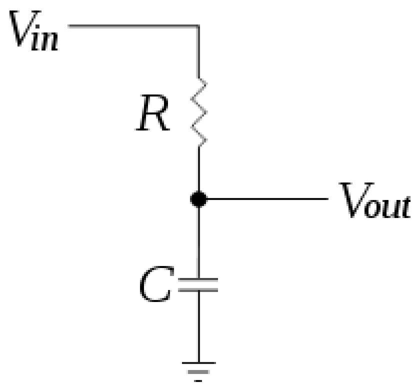
For example, a first-order low-pass filter can be described in Laplace notation as

$$\frac{\text{Output}}{\text{Input}} = K\frac{1}{1 + s\tau}$$

where s is the Laplace transform variable, τ is the filter time constant, and K is the filter passband gain.

Electronic low-pass filters

Passive electronic realization



Passive, first order low-pass RC filter

One simple electrical circuit that will serve as a low-pass filter consists of a resistor in series with a load, and a capacitor in parallel with the load. The capacitor exhibits reactance, and blocks low-frequency signals, causing them to go through the load instead. At higher frequencies the reactance drops, and the capacitor effectively functions as a short circuit. The combination of resistance and capacitance gives you the time constant of the filter $\tau = RC$ (represented by the Greek letter tau). The break frequency, also called the turnover frequency or cutoff frequency (in hertz), is determined by the time constant:

$$f_c = \frac{1}{2\pi\tau} = \frac{1}{2\pi RC}$$

or equivalently (in radians per second):

$$\omega_c = \frac{1}{\tau} = \frac{1}{RC}.$$

One way to understand this circuit is to focus on the time the capacitor takes to charge. It takes time to charge or discharge the capacitor through that resistor:

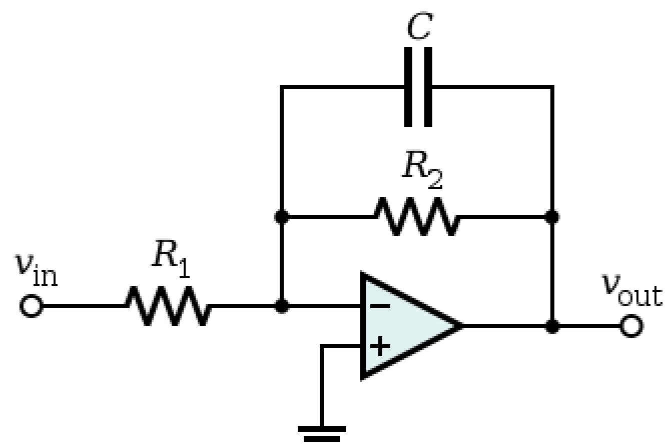
- At low frequencies, there is plenty of time for the capacitor to charge up to practically the same voltage as the input voltage.
- At high frequencies, the capacitor only has time to charge up a small amount before the input switches direction. The output goes up and down only a small fraction of the amount the input goes up and down. At double the frequency, there's only time for it to charge up half the amount.

Another way to understand this circuit is with the idea of reactance at a particular frequency:

- Since DC cannot flow through the capacitor, DC input must "flow out" the path marked V_{out} (analogous to removing the capacitor).
- Since AC flows very well through the capacitor — almost as well as it flows through solid wire — AC input "flows out" through the capacitor, effectively short circuiting to ground (analogous to replacing the capacitor with just a wire).

The capacitor is not an "on/off" object (like the block or pass fluidic explanation above). The capacitor will variably act between these two extremes. It is the Bode plot and frequency response that show this variability.

Active electronic realization



An active low-pass filter

Another type of electrical circuit is an *active* low-pass filter.

In the operational amplifier circuit shown in the figure, the cutoff frequency (in hertz) is defined as:

$$f_c = \frac{1}{2\pi R_2 C}$$

or equivalently (in radians per second):

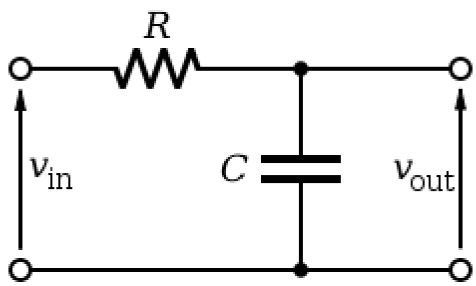
$$\omega_c = \frac{1}{R_2 C}.$$

The gain in the passband is $-R_2/R_1$, and the stopband drops off at −6 dB per octave as it is a first-order filter.

Sometimes, a simple gain amplifier (as opposed to the very-high-gain operational amplifier) is turned into a low-pass filter by simply adding a feedback capacitor *C*. This feedback decreases the frequency response at high frequencies via the Miller effect, and helps to avoid oscillation in the amplifier. For example, an audio amplifier can be made into a low-pass filter with cutoff frequency 100 kHz to reduce gain at frequencies which would otherwise oscillate. Since the audio band (what we can hear) only goes up to 20 kHz or so, the frequencies of interest fall entirely in the passband, and the amplifier behaves the same way as far as audio is concerned.

Discrete-time realization

The effect of a low-pass filter can be simulated on a computer by analyzing its behavior in the time domain, and then discretizing the model.



A simple low-pass RC filter

From the circuit diagram to the right, according to Kirchoff's Laws and the definition of capacitance:

$$v_{\text{in}}(t) - v_{\text{out}}(t) = R\,i(t) \qquad \textbf{(V)}$$
$$Q_c(t) = C\,v_{\text{out}}(t) \qquad \textbf{(Q)}$$
$$i(t) = \frac{\mathrm{d}Q_c}{\mathrm{d}t}\,, \qquad \textbf{(I)}$$

where $Q_c(t)$ is the charge stored in the capacitor at time t. Substituting equation **Q** into equation **I** gives $i(t) = C\frac{\mathrm{d}v_{\text{out}}}{\mathrm{d}t}$, which can be substituted into equation **V** so that:

$$v_{\text{in}}(t) - v_{\text{out}}(t) = RC\frac{\mathrm{d}v_{\text{out}}}{\mathrm{d}t}.$$

This equation can be discretized. For simplicity, assume that samples of the input and output are taken at evenly-spaced points in time separated by Δ_T time. Let the samples of v_{in} be represented by the sequence $(x_1, x_2, \ldots, x_n)$, and let v_{out} be represented by the sequence $(y_1, y_2, \ldots, y_n)$which correspond to the same points in time. Making these substitutions:

$$x_i - y_i = RC\,\frac{y_i - y_{i-1}}{\Delta_T}.$$

And rearranging terms gives the recurrence relation

$$y_i = \overbrace{x_i \left(\frac{\Delta_T}{RC + \Delta_T} \right)}^{\text{Input contribution}} + \overbrace{y_{i-1} \left(\frac{RC}{RC + \Delta_T} \right)}^{\text{Inertia from previous output}} .$$

That is, this discrete-time implementation of a simple RC low-pass filter is the exponentially-weighted moving average

$$y_i = \alpha x_i + (1 - \alpha) y_{i-1} \qquad \text{where} \qquad \alpha \triangleq \frac{\Delta_T}{RC + \Delta_T}.$$

By definition, the *smoothing factor* $0 \leq \alpha \leq 1$. The expression for α yields the equivalent time constant *RC* in terms of the sampling period Δ_T and smoothing factor α:

$$RC = \Delta_T \left(\frac{1 - \alpha}{\alpha} \right).$$

If α = 0.5, then the *RC* time constant is equal to the sampling period. If $\alpha \ll 0.5$, then *RC* is significantly larger than the sampling interval, and $\Delta_T \approx \alpha RC$.

Algorithmic implementation

The filter recurrence relation provides a way to determine the output samples in terms of the input samples and the preceding output. The following pseudocode algorithm will simulate the effect of a low-pass filter on a series of digital samples:

```
// Return RC low-pass filter output samples, given input samples,
// time interval dt, and time constant RC
function lowpass(real[0..n] x, real dt, real RC)
  var real[0..n] y
  var real α := dt / (RC + dt)
  y[0] := x[0]
  for i from 1 to n
      y[i] := α * x[i] + (1-α) * y[i-1]
  return y
```

The loop which calculates each of the *n* outputs can be refactored into the equivalent:

```
  for i from 1 to n
      y[i] := y[i-1] + α * (x[i] - y[i-1])
```

That is, the change from one filter output to the next is proportional to the difference between the previous output and the next input. This exponential smoothing property matches the exponential decay seen in the continuous-time system. As expected, as the time constant *RC* increases, the discrete-time smoothing parameter α decreases, and the output samples $(y_1, y_2, \ldots, y_n)$respond more slowly to a change in the input samples $(x_1, x_2, \ldots, x_n)$ – the system will have more *inertia*.

Chapter- 11

Wiener Filter

In signal processing, the **Wiener filter** is a filter proposed by Norbert Wiener during the 1940s and published in 1949. Its purpose is to reduce the amount of noise present in a signal by comparison with an estimation of the desired noiseless signal. The discrete-time equivalent of Wiener's work was derived independently by Kolmogorov and published in 1941. Hence the theory is often called the **Wiener-Kolmogorov** filtering theory. The Wiener-Kolmogorov was the first statistically designed filter to be proposed and subsequently gave rise to many others including the famous Kalman filter. A Wiener filter is not an adaptive filter because the theory behind this filter assumes that the inputs are stationary.

Description

The goal of the Wiener filter is to filter out noise that has corrupted a signal. It is based on a statistical approach.

Typical filters are designed for a desired frequency response. However, the design of the Wiener filter takes a different approach. One is assumed to have knowledge of the spectral properties of the original signal and the noise, and one seeks the linear time-invariant filter whose output would come as close to the original signal as possible. Wiener filters are characterized by the following:

1. Assumption: signal and (additive) noise are stationary linear stochastic processes with known spectral characteristics or known autocorrelation and cross-correlation
2. Requirement: the filter must be physically realizable/causal (this requirement can be dropped, resulting in a non-causal solution)
3. Performance criterion: minimum mean-square error (MMSE)

This filter is frequently used in the process of deconvolution; for this application.

Wiener filter problem setup

The input to the Wiener filter is assumed to be a signal, $s(t)$, corrupted by additive noise, $n(t)$. The output, $\hat{s}(t)$, is calculated by means of a filter, $g(t)$, using the following convolution:

$$\hat{s}(t) = g(t) * [s(t) + n(t)]$$

where

- $s(t)$is the original signal (not exactly known; to be estimated)
- $n(t)$is the noise
- $\hat{s}(t)$is the estimated signal (the intention is to equal $s(t+\alpha)$)
- $g(t)$is the Wiener filter's impulse response

The error is defined as

$$e(t) = s(t+\alpha) - \hat{s}(t)$$

where

- $\alpha$is the delay of the Wiener filter (since it is causal)

In other words, the error is the difference between the estimated signal and the true signal shifted by α.

The squared error is

$$e^2(t) = s^2(t+\alpha) - 2s(t+\alpha)\hat{s}(t) + \hat{s}^2(t)$$

where

- $s(t+\alpha)$is the desired output of the filter
- $e(t)$is the error

Depending on the value of α, the problem can be described as follows:

- If $\alpha > 0$then the problem is that of prediction (error is reduced when $\hat{s}(t)$is similar to a later value of *s*)
- If $\alpha = 0$then the problem is that of filtering (error is reduced when $\hat{s}(t)$is similar to $s(t)$)
- If $\alpha < 0$then the problem is that of smoothing (error is reduced when $\hat{s}(t)$is similar to an earlier value of *s*)

Writing $\hat{s}(t)$as a convolution integral:

$$\hat{s}(t) = \int_{-\infty}^{\infty} g(\tau)\left[s(t-\tau) + n(t-\tau)\right]\, d\tau.$$

Taking the expected value of the squared error results in

$$E(e^2) = R_s(0) - 2\int_{-\infty}^{\infty} g(\tau) R_{xs}(\tau+\alpha)\, d\tau + \iint_{[-\infty,-\infty]}^{[\infty,\infty]} g(\tau) g(\theta) R_x(\tau-\theta)\, d\tau\, d\theta$$

where

- $x(t) = s(t) + n(t)$ is the observed signal
- R_s is the autocorrelation function of $s(t)$
- R_x is the autocorrelation function of $x(t)$
- R_{xs} is the cross-correlation function of $x(t)$ and $s(t)$

If the signal $s(t)$ and the noise $n(t)$ are uncorrelated (i.e., the cross-correlation R_{sn} is zero), then this means that

- $R_{xs} = R_s$
- $R_x = R_s + R_n$

For many applications, the assumption of uncorrelated signal and noise is reasonable.

The goal is to minimize $E(e^2)$, the expected value of the squared error, by finding the optimal $g(t)$, the Wiener filter impulse response function.

Wiener filter solutions

The Wiener filter problem has solutions for three possible cases: one where a non-causal filter is acceptable (requiring an infinite amount of both past and future data), the case where a causal filter is desired (using an infinite amount of past data), and the FIR case where a finite amount of past data is used. The first case is simple to solve but is not suited for real-time applications. Wiener's main accomplishment was solving the case where the causality requirement is in effect, and in an appendix of Wiener's book Levinson gave the FIR solution.

Noncausal solution

$$G(s) = \frac{S_{x,s}(s)}{S_x(s)} e^{\alpha s}.$$

Provided that $g(t)$ is optimal, then the minimum mean-square error equation reduces to

$$E(e^2) = R_s(0) - \int_{-\infty}^{\infty} g(\tau) R_{x,s}(\tau + \alpha)\, d\tau,$$

and the solution $g(t)$ is the inverse two-sided Laplace transform of $G(s)$.

Causal solution

$$G(s) = \frac{H(s)}{S_x^+(s)},$$

where

- $H(s)$ consists of the causal part of $\frac{S_{x,s}(s)}{S_x^-(s)} e^{\alpha s}$ (that is, that part of this fraction having a positive time solution under the inverse Laplace transform)
- $S_x^+(s)$ is the causal component of $S_x(s)$ (i.e., the inverse Laplace transform of $S_x^+(s)$ is non-zero only for $t \geq 0$)
- $S_x^-(s)$ is the anti-causal component of $S_x(s)$ (i.e., the inverse Laplace transform of $S_x^-(s)$ is non-zero only for $t < 0$)

This general formula is complicated and deserves a more detailed explanation. To write down the solution $G(s)$ in a specific case, one should follow these steps:

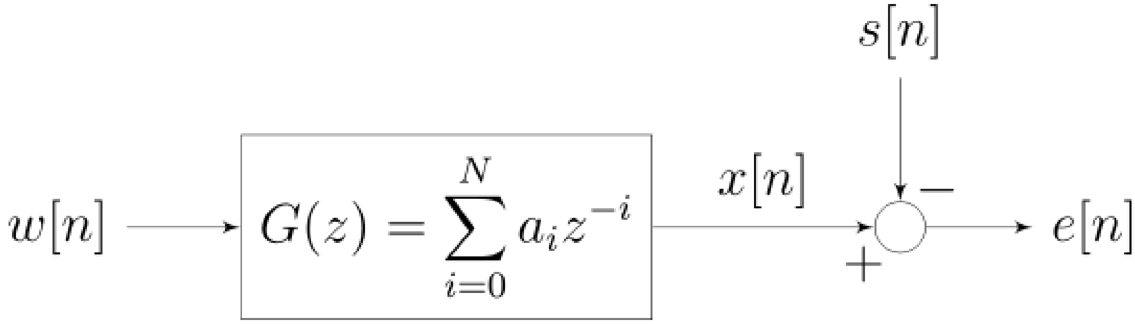
1. Start with the spectrum $S_x(s)$ in rational form and factor it into causal and anti-causal components:

   $$S_x(s) = S_x^+(s) S_x^-(s)$$

   where S^+ contains all the zeros and poles in the left hand plane (LHP) and S^- contains the zeroes and poles in the RHP. This is called the Wiener–Hopf factorization.

2. Divide $S_{x,s}(s) e^{\alpha s}$ by $S_x^-(s)$ and write out the result as a partial fraction expansion.
3. Select only those terms in this expansion having poles in the LHP. Call these terms $H(s)$.
4. Divide $H(s)$ by $S_x^+(s)$. The result is the desired filter transfer function $G(s)$.

Finite Impulse Response Wiener filter for discrete series



Block diagram view of the FIR Wiener filter for discrete series. An input signal $w[n]$ is convolved with the Wiener filter $g[n]$ and the result is compared to a reference signal $s[n]$ to obtain the filtering error $e[n]$.

The causal finite impulse response (FIR) Wiener filter, instead of using some given data matrix X and output vector Y, finds optimal tap weights by using the statistics of the input and output signals. It populates the input matrix X with estimates of the auto-correlation of the input signal (T) and populates the output vector Y with estimates of the cross-correlation between the output and input signals (V).

In order to derive the coefficients of the Wiener filter, we consider a signal $w[n]$ being fed to a Wiener filter of order N and with coefficients $\{a_i\}$, $i = 0, \ldots, N$. The output of the filter is denoted $x[n]$ which is given by the expression

$$x[n] = \sum_{i=0}^{N} a_i w[n-i].$$

The residual error is denoted $e[n]$ and is defined as $e[n] = x[n] - s[n]$. The Wiener filter is designed so as to minimize the mean square error (MMSE criteria) which can be stated concisely as follows:

$$a_i = \arg\min \ E\{e^2[n]\},$$

where $E\{\cdot\}$ denote the expectation operator. In the general case, the coefficients a_i may be complex and may be derived for the case where $w[n]$ and $s[n]$ are complex as well. With a complex signal, the matrix to be solved is a Hermitian Toeplitz matrix, rather than Symmetric Toeplitz matrix. For simplicity, we will only consider the case where all these quantities are real. The mean square error may be rewritten as:

$$\begin{aligned} E\{e^2[n]\} &= E\{(x[n] - s[n])^2\} \\ &= E\{x^2[n]\} + E\{s^2[n]\} - 2E\{x[n]s[n]\} \\ &= E\{\big(\textstyle\sum_{i=0}^{N} a_i w[n-i]\big)^2\} + E\{s^2[n]\} - 2E\{\textstyle\sum_{i=0}^{N} a_i w[n-i]s[n]\}. \end{aligned}$$

To find the vector $[a_0, \ldots, a_N]$ which minimizes the expression above, let us now calculate its derivative with respect to a_i

$$\begin{aligned}\frac{\partial}{\partial a_i} E\{e^2[n]\} &= 2E\{\left(\sum_{j=0}^{N} a_j w[n-j]\right) w[n-i]\} - 2E\{s[n]w[n-i]\} \quad i = 0, \ldots, N \\ &= 2\sum_{j=0}^{N} E\{w[n-j]w[n-i]\}a_j - 2E\{w[n-i]s[n]\}.\end{aligned}$$

If we suppose that $w[n]$ and $s[n]$ are each stationary and jointly stationary, we can introduce the following sequences $R_w[m]$ and $R_{ws}[m]$ known respectively as the autocorrelation of $w[n]$ and the cross-correlation between $w[n]$ and $s[n]$ defined as follows

$$\begin{aligned}R_w[m] &= E\{w[n]w[n+m]\} \\ R_{ws}[m] &= E\{w[n]s[n+m]\}.\end{aligned}$$

The derivative of the MSE may therefore be rewritten as (notice that $R_{ws}[-i] = R_{sw}[i]$)

$$\frac{\partial}{\partial a_i} E\{e^2[n]\} = 2\sum_{j=0}^{N} R_w[j-i]a_j - 2R_{sw}[i] \quad i = 0, \ldots, N.$$

Letting the derivative be equal to zero, we obtain

$$\sum_{j=0}^{N} R_w[j-i]a_j = R_{sw}[i] \quad i = 0, \ldots, N,$$

which can be rewritten in matrix form

$$\begin{aligned}&\mathbf{Ta} = \mathbf{v} \\ \Rightarrow &\begin{bmatrix} R_w[0] & R_w[1] & \cdots & R_w[N] \\ R_w[1] & R_w[0] & \cdots & R_w[N-1] \\ \vdots & \vdots & \ddots & \vdots \\ R_w[N] & R_w[N-1] & \cdots & R_w[0] \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} R_{sw}[0] \\ R_{sw}[1] \\ \vdots \\ R_{sw}[N] \end{bmatrix}\end{aligned}$$

These equations are known as the Wiener-Hopf equations. The matrix T appearing in the equation is a symmetric Toeplitz matrix. These matrices are known to be positive definite and therefore non-singular yielding a unique solution to the determination of the Wiener filter coefficient vector, $\mathbf{a} = \mathbf{T}^{-1}\mathbf{v}$. Furthermore, there exists an efficient algorithm to solve such Wiener-Hopf equations known as the Levinson-Durbin algorithm so an explicit inversion of $\mathbf{T}$ is not required.

The realization of the causal Wiener filter looks a lot like the solution to the least squares estimate, except in the signal processing domain. The least squares solution, for input matrix $\mathbf{X}$ and output vector $\boldsymbol{y}$ is

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}^{\mathbf{T}}\mathbf{X})^{-1}\mathbf{X}^{\mathbf{T}}\boldsymbol{y}.$$

The FIR Wiener filter is related to the least mean squares filter, but minimizing its error criterion does not rely on cross-correlations or auto-correlations. Its solution converges to the Wiener filter solution.