# INTERMEDIATE – FUNCTION SYNTHESIS

by
W. Martin Snelgrove

A Thesis submitted in conformity with the requirements for the degree of Doctor of Philosophy in the University of Toronto.

Department of Electrical Engineering
University of Toronto.

Dec. 1981

It has been a privilege, both professional and personal, to work with Adel Sedra for the last few years. His teaching of, and feeling for, circuits has been a major influence in making me respect the area.

My friends have all joined energetically in helping and hindering my thesis and the rest of my life. I'll do what I can in return.

# Intermediate-Function Synthesis

## Abstract

A vector-space approach to the design of active circuits implementing filters is presented and developed. Principles of synthesis and analysis are given together with formulae and algorithms suitable for incorporation into computer-aided design systems. The technique is shown by example to provide tools for both practical and theoretical work.

Table of Contents

# 1. Introduction

This thesis proposes a novel technique for synthesizing circuits realizing given transfer functions. This method provides new insight into the problem of how best to interconnect a number of identical "operators", e.g. integrators or delays, with a feedback/feedforward network so as to produce the given transfer function (e.g. filter response). It does so by concentrating the designer's attention on a set of transfer functions related to that to be realized rather than on a network: thus the problem of realizing a transfer function is stated in terms of transfer functions rather than in terms of networks. We call the technique ***intemediate- function synthesis*** because this set of transfer functions describes behaviour at internal states of the filter.

## 1.1 Technological Motivation

Electronic technology is changing very rapidly, continually producing new devices and circuits with which to synthesize electronic systems. A collection of theoretical techniques, changing somewhat less rapidly, gives design engineers the power and insight to solve the new problems and variants on old problems that result.

At present the primary driving force is the continuing development of integrated circuit technology: process advances continue to increase the number of functions that can be integrated, while computer-aided design advances are simultaneously reducing the design cost overhead associated with ICs. Both of these types of change, generally grouped under the banner of "VLSI" (Very Large Scale Integration), have the effect of "fragmenting" the problem of integrated circuit design: as density increases it becomes steadily harder to choose general-purpose "building blocks" to implement as chips, and so more specialized ones are needed; and as the cost of design is reduced by

CAD the need for very large markets to amortize front-end (design) cost is reduced. As the complexity available in a single chip approaches and passes the complexity required for many complete systems it becomes vital to give the problem of circuit design, in integrated form, to the "end-user" circuit designer rather than to an IC specialist.

In this fluid kind of situation any theoretical technique that promises insight into the design of some class of circuit, and most especially one naturally suited to CAD, may hope to contribute to the current (r)evolution of microelectronics.

The filter design problem is a fundamental one because it is basically the problem of designing a "difficult" linear system to perform well in the real world: linear systems in turn appear often enough in the world to make the circuits that best deal with them inherently important. Filters are, of course, a specialization of the class of linear systems that pays special attention to "bands" of frequencies: but many techniques developed for this particularly difficult case may readily be applied to others.

Much of the network theory that was originally developed for filter design was tuned to deal with constraints that are now less important (at least at audio frequencies), Stability tests designed to reduce the computation needed to determine whether a function is stable, at the cost of making filter theory more complicated, may be ignored when computers are available to do the tedious work of searching. Theory designed to use passive elements as much as possible in order to save on vacuum tubes is out of place in a world in which transistors are cheaper than resistors and capacitors, and inductors often impractical. An odd relic of LC technology) the simulation of doubly-terminated ladders to obtain their excellent passband behaviour, [I] is working hard: the number of design techniques seeking to somehow simulate a ladder is enormous. Something created for the new problems could help to shore up design in the places where those techniques falter.

Classical circuit theory gives all the tools required for the task of circuit ***analysis,*** and one may use analysis together with the brute power of numerical search and optimization to provide some power for synthesis (which is inverse to analysis). If circuit applications continue to head in the direction of increasing complexity, however, this kind of approach seems bound to founder on the rocks of computational complexity: the problem of optimizing a circuit or system in the total absence of insight into its inherent structure appears to be exponential in the complexity of the system.

The state-variable approach to system description still applies, however, and should be expected to continue to apply as long as we are interested in interconnecting numbers of identical linear operators (like integrators or delays) to perform functions more complicated than a single one could.

This thesis attempts to contribute a theoretical tool with which to do design of linear systems, and particularly of filters, to an industry in which design cost is coming to be the principal concern.

## 1.2 Introduction tothe thesis

The central idea of this thesis, our new intermediate-function (IF) synthesis technique, is described in chapter 2 of this document, and the remaining material is intended to demonstrate that it is a good tool for filter design.

The technique uses the state-space formulation as an intermediary between abstract structure and circuits: chapter 3 shows some types of circuit that may reasonably be derived this way.

Chapter 4 shows how the practical problems of circuits, such as noise generation and sensitivity to component errors, can be measured in terms of the new method.

Chapter 5 discusses the utility of “redundancy” in linear systems in terms of the synthesis technique, and shows how “good” circuits use this effect. The principle of redundancy is shown to be an important one for high-performance circuits, and appears in several apparently different forms. Chapter 5 shows up the relationship among these forms, thus yielding a better understanding of high-performance filters.

An important property of IF synthesis is that it unifies many apparently quite different synthesis techniques, so that a computer program may be written as a design aid with enough generality to cover many different types of problem. Chapter 6 shows how the synthesis method may be formulated in terms of matrix computations for such a program. A side-effect of this work is that the matrix formulae clearly show the relationships among quantities of interest: some interesting results are derived from this.

The methods developed in the preceding chapters are used in chapter 7 to investigate filters whose sensitivity approaches a lower bound.

Chapter 8 is a design example for an eighth-order problem, and shows how the synthesis method may be applied to a non-trivial filter problem. Some well-known types of design are described in terms of IF synthesis so as to firmly connect the technique with prior art, and some quite new designs are presented to show how easy it is to use the synthesis technique to invent and modify structures to solve particular problems.

As another application of the technique, and also for its own sake, some theory regarding “complex filters” is developed in Chapter 9. These have application in single-sideband modulation and other signal-processing applications. Our synthesis method is used as a tool in investigating inherent properties of these structures as well as to obtain realizations.

One final idea is sketched in Chapter 10: a variation on the conventional approximation problem is suggested that takes sensitivity and dynamic range problems into account at the initial approximation stage of filter design. This technique is incidentally shown to give a new solution to the difficult problem of simultaneously meeting specifications on attenuation and group delay. The relationship between transfer function and realization required for this method is best understood by reference to the work of chapter 7.

Chapter 11 summarizes the earlier results and draws some conclusions on the synthesis technique itself and on some of the problems we have attacked with it. It also outlines areas that warrant further investigation.

## 1.3 References

[1] H.J. Orchard, *Inductorless Filters* Electron. Lett. vol. 2, pp.224-225, June 1966

# 2. The Synthesis Method

This chapter states the intermediate-function synthesis technique and demonstrates it with an example. Later chapters will show examples that motivate the technique, while this one is simply concerned with showing what it is.

## 2.1 Notation used in this thesis

The notation used here is fairly conventional except that we use a subscript “s” or “t” to distinguish between functions and their Laplace transforms, rather than the conventional technique of using lower-case to denote the time function and upper-case to denote the transform. This is made necessary by the fact that we also have matrices that represent some of the same functions, and wish to reserve upper-case to denote them. It is a convenient effect of this notation that we may simply elide subscripts in any formulae that apply equally to functions and to their various transforms.

As an example, we will use $u_t$ to denote the input signal as a function of time and $u_s$ to denote its transform. A formula like $y=\mathbf{c}^T\mathbf{x}+d\cdot u$ means as special cases that $y_t=\mathbf{c}^T\mathbf{x}_t+d\cdot u_t$ and that $y_s=\mathbf{c}^T\mathbf{x}_s+d\cdot u_s$.

We use lower-case italic letters to denote scalar quantities (e.g. "$p$" etcetera); lower-case bold-face quantities for vectors (e.g. “x”); and upper-case bold-face for matrices (e.g. “**A**”). The $i^{th}$ element of a vector x will be written $\mathbf{x}_i$ instead of the more conventional $x_i$ in order that quantities appear everywhere in the same typeface (again in order to avoid conflicts of names). The fact that it is subscripted serves to show that the result is scalar Similarly, we use $\mathbf{A}_{ij}$ to denote an element of **A**.

## 2.2 State Equations

Our abstraction of filters as composed of a number of integrators connected together by some structure of feedback and feedforward paths is the one described by the conventional state-space system equations:

$$\mathbf{x}_t' = \mathbf{A}\mathbf{x}_t + \mathbf{b}u_t \tag{2-1}$$

$$y = \mathbf{c}^T\mathbf{x} + du$$

where "u " is the input signal; "x" is a vector of n "states", which are just the outputs of integrators; "y" is the output signal; and "A", "b", "c" and "$d$" are coefficient.s relating these variables.

The transfer function of this kind of system is simply

$$t_s = \frac{y_s}{u_s} = \mathbf{c}^T(s\mathrm{I}-\mathrm{A})^{-1}\mathbf{b}+d \tag{2-2}$$

We can find the values of the states "x" from the vector of transfer functions

$$\mathbf{f}_s \triangleq (s\mathrm{I}-\mathbf{A})^{-1}\mathbf{b} = \frac{\mathbf{x}_s}{u_s} \tag{2-3}$$

Calculation of (2-3) is never actually needed in our syntheses: it just serves as a handy definition.

"f" gives us the effect at intermediate states in a filter of the system input. Later we will also be interested in the effect of each state on the system output. The vector of functions

$$\mathbf{g}_s^T \triangleq \mathbf{c}^T(s\mathrm{I}-\mathrm{A})^{-1} \tag{2-4}$$

measures transfer functions to the system output from the inputs of the integrators.

Again, while (2-4) is a useful definition, it is not actually used for synthesis: a formula giving g directly from f is presented in chapter 6.

A signal-flow graph analogue of the state equations (2-1) appears in figure 2.1[*];

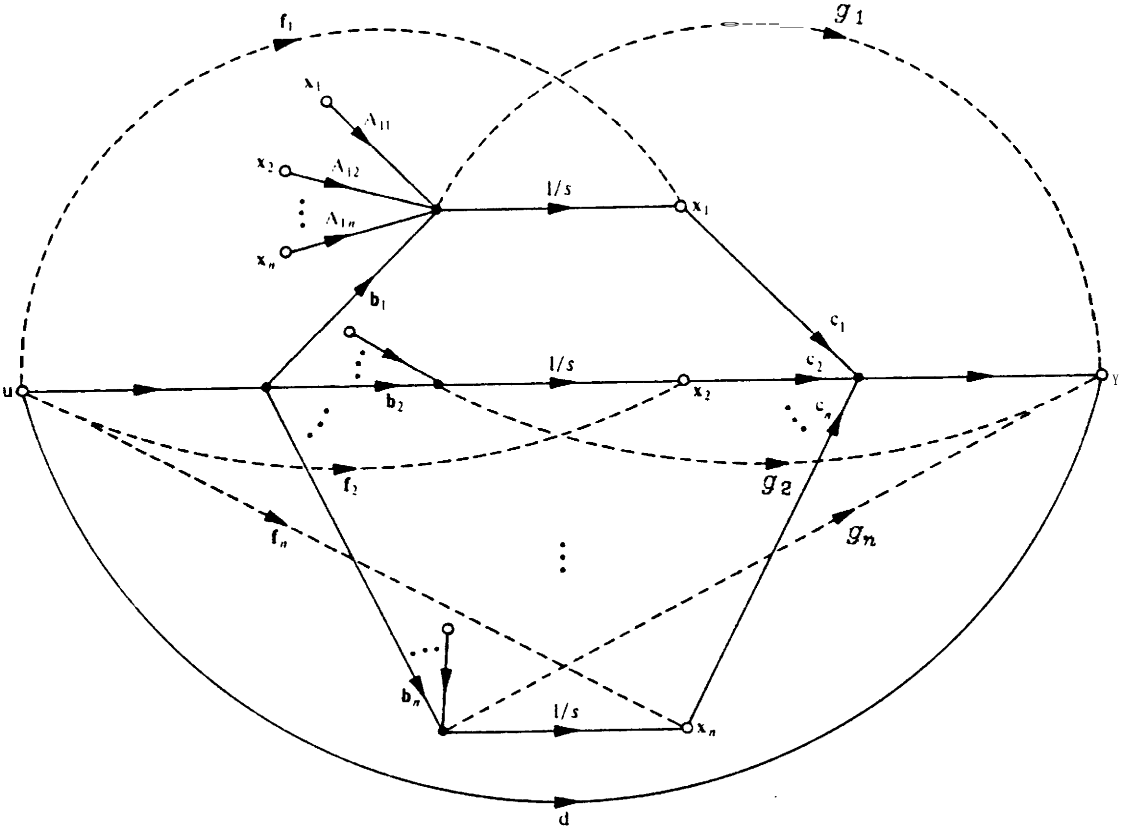


**Figure 2.1: SFG of general system of state equations**

it shows a network of feedbacks interconnecting n integrators, and indicates the significance of $\{f_i\}$ and $\{g_i\}$. This is a fully "dense" topology: "sparse" practical topologies are special cases of it that eliminate many SFG edges.

[*] -We use broken lines in signal-flow graphs to represent transferfunctions between nodes rather than actual links.

## 2.3 Intermediate-Function Synthesis

The fundamental synthesis method of this thesis is an inversion of the conventional analysis described in section 2.2 above. Rather than choosing a structure and hence **{A,b,c,d }**, and then evaluating $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ to see how the design performs, we choose a "good" vector **f** (or, dually, g) and derive **{A,b,c,d}** from that. Our design effort may now be concentrated on choosing desirable $\{\mathbf{f}_i\}$, which measure performance directly, rather than on analyzing arbitrary structures to find their $\{\mathbf{f}_i\}$.

The observation which originally motivated this approach was that every canonic system realizing a transfer function of degree n (which must contain n integrators) must necessarily have intermediate transfer functions (from the system input to the outputs of each of its n integrators) that are linearly independent. This follows from the fact that n integrators are known to be necessary by a simple argument: if an integrator's output were linearly dependent on the outputs of the other integrators in a system, that integrator could be replaced by a weighted sum of other integrator voltages without disturbing the behaviour of the rest of the circuit; but then we would have a realization of an nth order function with only n-1 integrators, which is impossible. The system input, u, must also be independent of the states x or one could similarly eliminate states and reduce order.

We may also observe that the **{A,b,c,d}** coefficients are uniquely determined by the $\{\mathbf{f}_i\}$ (for a canonic system) exactly because the $\{\mathbf{f}_i\}$ must be independent: there can only be one way to form the required input signal to each integrator and to the output summer in figure 2.1 as a linear combination of an independent set of signals.

The $n\{\mathbf{f}_i\}$ have to have a common denominator – $e_s=\det(sI-\mathbf{A})^{-1}$ – and must all have numerators of order less than n. Within these constraints any n independent $\{\mathbf{f}_i\}$ will produce a system realising $t_s$.

Because any canonic $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ system must have some $\{\mathbf{f}_i\}$ **set** of the form we have described, and because $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ are uniquely defined by $\{\mathbf{f}_i\}$, our method can produce any canonic realization of $t_s$.

One might be led by the need for independence to think that a system in which linear dependencies “almost” existed would be in some way impractical, and later chapters will show that this is the case: in particular, sensitivity and dynamic range behaviour become worse as the system approaches linear dependencies. On the other hand it turns out that a limited amount of correlation among integrators is useful.

## 2.4 A Simple Example of Synthesis

The exampIe we will use here is a second-order Butterworth filter synthesis, which we will take from an arbitrary choice of a set of intermediate functions to a circuit realization. Many other, more familiar, techniques would produce exactly the same circuit. The point here is to give an overview of the technique on **a** very simple problem.

The function to be realized is:

$$t_s = \frac{1}{s^2+\sqrt{2}s+1} \tag{2-5}$$

$$\overset{\Delta}{=} \frac{p_s}{e_s}$$

we may, for instance, arbitrarily choose intermediate functions

$$\mathbf{f}_{1,s} = \frac{1}{e_s} \tag{2-6}$$

$$\mathbf{f}_{2,s} = \frac{s}{e_s}$$

to be the n (i.e. 2) required linearly independent transfer functions from the system input to the outputs of the two integrators.

Now if we convert the first row of the system equation for $\mathbf{x}'_t$ (2-1) to Laplace form and substitute $\mathbf{f}_{1,s} \cdot u_s$ and $\mathbf{f}_{2,s} \cdot u_s$ for states $\mathbf{x}_1$ and $\cdot\mathbf{x}_2$ respectively, we get:

$$s\,\mathbf{f}_{1,s}\cdot u_s = \mathbf{A}_{11}\mathbf{f}_{1,s}\cdot u_s + \mathbf{A}_{12}\mathbf{f}_{2,s}\cdot u_s + \mathbf{b}_1\cdot u_s$$

Substituting for $\mathbf{f}_{1,s}$ and $\mathbf{f}_{2,s}$, and multiplying both sides of the equation by $e_s / u_s$,

$$s = \mathbf{A}_{11} + \mathbf{A}_{12}s + \mathbf{b}_1 e_s$$

$$= \mathbf{A}_{11} + \mathbf{A}_{12}s + \mathbf{b}_1(s^2 + \sqrt{2}s + 1)$$

And now, comparing coefficients, we get

$$\mathbf{A}_{11} = \mathbf{b}_1 = 0$$

$$\mathbf{A}_{12} = 1$$

Similarly, the second row yields

$$s^2 = \mathbf{A}_{21} + \mathbf{A}_{22}s + \mathbf{b}_2(s^2 + \sqrt{2}s + 1)$$

$$\Rightarrow \mathbf{b}_2 = 1$$

$$\mathbf{A}_{21} = -1$$

$$\mathbf{A}_{22} = -\sqrt{2}$$

Now we solve for "$\mathbf{c}$" and "$d$" by a similar scheme:

$$y_s = [p_s / e_s]\cdot u_s$$

$$= \mathbf{c}_1\mathbf{f}_{1,s}\cdot u + \mathbf{c}_2\mathbf{f}_{2,s}\cdot u_s + d\cdot u_s$$

$$\Rightarrow p_s = \mathbf{c}_1 + \mathbf{c}_2\cdot s + d\cdot e_s$$

$$=> \mathbf{c}_2 = d = 0$$

$$\mathbf{c}_1 = 1$$

So that the system we need is

$$\begin{bmatrix} \mathbf{x}'_{1,t} \\ \mathbf{x}'_{2,t} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & -\sqrt{2} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{1,t} \\ \mathbf{x}_{2,t} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \cdot u \qquad (2\text{-}7)$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}$$

which may be recognized as a companion-form realization.

We could work out $\{\mathbf{g}_i\}$ for this system in the conventional way from its definition above (although better ways are discussed in chapter 6):

$$\mathbf{g}^T \triangleq \mathbf{c}^T (s\mathbf{I} - \mathbf{A})^{-1}$$

$$= [1 \;\; 0] \begin{bmatrix} s & -1 \\ 1 & s+\sqrt{2} \end{bmatrix}^{-1}$$

$$= [1 \;\; 0] \frac{1}{s^2+\sqrt{2}s+1} \begin{bmatrix} s+\sqrt{2} & 1 \\ -1 & s \end{bmatrix}$$

$$= [\frac{s+\sqrt{2}}{e_s} \;\; \frac{1}{e_s}]$$

We could also have derived the system from these $\{\mathbf{g}_i\}$ in the same kind of way as from $\{\mathbf{f}_i\}$.

Many circuits may be regarded as implementations of (2-7) but some have structures that correspond to the system of equations more directly than do others. When there is a close correspondence between the structure of the circuit and the system equations we will be able to model performance of the circuit directly by investigating the system $\{\mathbf{f}_i\}$. The most general approach which maintains a close correspondence is that conventional in analog computing: a network of operational-amplifier integrators and summers, having interconnecting conductances proportional to matrix entries, Figure (2.2) is a schematic for such an implementation of (2-7).

Chapter 3 will show some of the other kinds of circuit possible, but note that this one is simply the well known Tow-Thomas three-amplifier biquad [1] One may conclude that the method proposed is capable of generating a well-known (and good) solution to a simple filter-design problem.

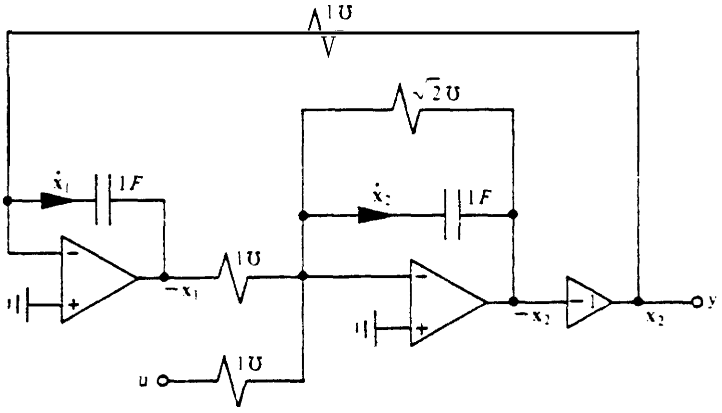


**Figure** 2.2: **Circuit simulating system (2-7)**

We show in chapter 3 how to implement "leaky" integrators with RC networks like those of figure 2.3: state $\mathbf{x}_2$ in (2-7) is a candidate for such implementation, but (using the formulae of chapter 3) can be found to require negative resistors. By performing a simple gain-adjustment on $\mathbf{f}_{2,s}$ we can get a new system (with $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ different only by scale factors from the Tow-Thomas design) that may be implemented with only positive resistors. This results in the well-known single-amplifier biquad (SAB) design of Figure 2.4. More details will be given in chapter 3.

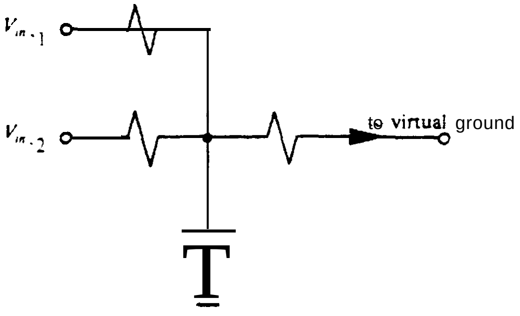


**Figure 2.3: Simple Leaky Integrator**

In this way we can use the state-space formulation not only to derive another popular circuit, but also to show that the essential difference between the (high-performance) Tow-Thomas design and the low-performance SAB is in signal scaling.

## 2.5 Choosing Intermediate Functions

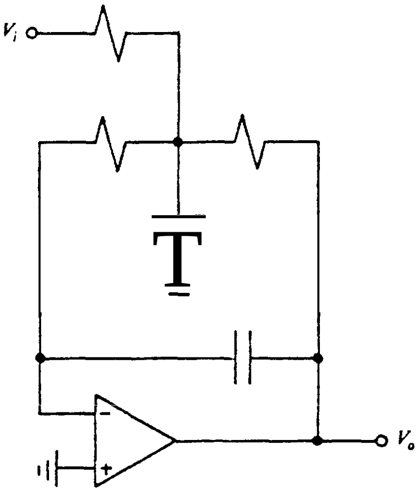


**Figure 2.4:** SAB **Derived by IF Synthesis**

The intermediate-function synthesis example above chose $\{\mathbf{f}_i\}$ arbitrarily: the idea was just to show that one can produce a system and then a circuit from a set of intermediate functions. The more dscult task is to show that this synthesis procedure provides a good way to do filter design and filter theory. An intuitive reason to believe that it does is that it states the problem of realizing a transfer function as one of designing a set of transfer functions - realizing an object with a collection of objects of the same type.

This document does not present a single way to choose $\{\mathbf{f}_i\}$ because the problem of filter design is too broad to permit a single scheme to be optimum for every case: choosing $\{\mathbf{f}_i\}$ is still a design problem. We have, however, developed some new design techniques to go with IF synthesis. Several complete designs are done in chapter 8 (on a representative filter problem) using these technique s, and chapters 3 to 7 provide background for statements and decisions made in these designs. Of these, chapter 3 is needed only to show that a state-variable system can readily be converted to a practical circuit, and chapter 6 is needed only to show that IF synthesis is easy to mechanize. Chapter 4 is important to design because it shows how we measure various important types of performance to compare and study designs, chapter 5 presents an important design trade-off, and chapter 7 investigates designs with low sensitivity to their integrators.

The types of design developed in chapter 8 are by no means the only ones possible with IF synthesis, just the only ones developed in this thesis.

Interesting new ideas have "spun off" this work on design. Some of these ideas are developed in passing in chapters 3 to 8, and others are developed separately in chapters 9 and 10. The latter chapters can almost stand alone, but assume that design of the style developed in chapter 8 is possible.

## 2.6 Conclusions

A set of n independent "intermediate transfer functions" $\{\mathbf{f}_i\}$ uniquely defines any canonic system realizing a given transfer function. Any canonic system may be derived in this way. Thus one may do design of linear systems by choosing $\{\mathbf{f}_i\}$ rather than topologies.

If $\{\mathbf{f}_i\}$ turn out to be related to measures of "goodness" for $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ systems more closely than are the $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ **coefficients** themselves then IF synthesis will offer a good way of designing systems of state equations. If systems of state equations may be used in a systematic way to derive circuits whose practical properties are closely related to "goodness" measures for the $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ systems then design of systems of state equations will offer a good way of designing a certain class of circuits.

## 2.7 References

[1] J. Tow, *"Active RC Filters – a State- Space Realization"*, Proc. IEEE, vol. 56, pp. 11374139, 1968

# 3. State-Variable Circuits

By “state-variable circuits” we mean those which may be designed in a straightforward way from an arbitrary state-space description. These are the types of circuit which can be designed by intermediate-function synthesis.

We exclude such things as passive ladders from the class of state-variable circuits because, although they are easily analyzed to give a state-variable description? they are not so readily synthesized from a given description. This chapter shows that several important types of circuit may be generated systematically from an $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ description, although some efficient ones require simple manipulations of the system description before they become practical.

## 3.1 Op-Amp-RC Circuits

The problem of simulating an arbitrary system of states can be solved with the summing integrators (Miller integrators) and op-amp inverters widely used in analog computation [1,2]. Tow-Thomas biquads are of this type.

The basic building block for this kind of circuit is shown in figure 3.1* and produces both positive and negative integrals of the input signal.

The input current at the $i^{th}$ integrator’s virtual ground represents $\mathbf{x}'_i$ and the two op-amps produce $\mathbf{x}_i$ and $-\mathbf{x}_i$ from this. Coefficient $\mathbf{A}_{ij}$, for instance, is implemented simply by connecting a conductance $G_{ij}=\frac{1}{C_i}|\mathbf{A}_{ij}|$ from output $\mathbf{x}_j$ or $-\mathbf{x}_j$ (according to whether $\mathbf{A}_{ij}$ is positive or negative) to the input of integrator $i$. "$\mathbf{b}_i$" are simulated by connections from the system input "$u$" to the input of

* The combination shown is nut exactly that used in analog ***computers,*** but contains an improvement suggested by Brackett and Sedra [3] that cancels phase errors in the inverter.

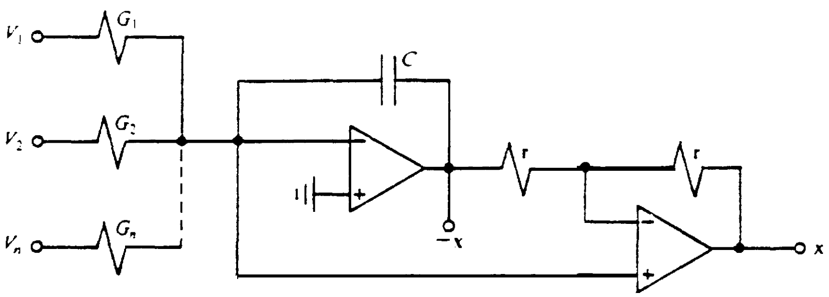


**Figure 3.1: Integrator-inverter Combination**

integrator "$i$" (for a completely general system, with some negative $\mathbf{b}_i$, straightforward simulation would require that an inverter be used to generate $-u_t(t)$).

The output can be formed with an inverting summer, having input conductances proportional to $|\mathbf{c}_i|$ driven from $\pm\mathbf{x}_i$ (according to the sign of $(\mathbf{c}_i)$) and a feedforward from $\pm u_t(t)$ proportional to $|d|$.

The impedance levels of conductances implementing all of these coefficients may be set arbitrarily by choosing the values of the feedback resistors or capacitors in the summers and summing integrators. Thus if $C_i$ is chosen to be the capacitor for integrator "$i$", then $G_{ij}=\frac{\mathbf{A}_{ij}}{R_{0,i}}$ where $R_{o,i}C_i=1$. One may choose to frequency-denormalize a filter design by a factor $\omega_n$ at this point by choosing instead $R_{o,i}C_i=\frac{1}{\omega_n}$.

While this kind of circuit is totally general it is also rather inefficient. Both $\mathbf{x}_i$ and $-\mathbf{x}_i$ are needed only if A contains elements of both signs in its $i^{th}$ column. One can usually reduce the number of inverters needed in a practical circuit by about 50% by simply choosing signs of $\{\mathbf{f}_i\}$ so as to make as many columns of A as possible have only negative signs: this may be done fairly easily by appropriate manipulations of the A matrix, as will be discussed below. Since the inverter in Figure 3.1 is needed only to produce $+\mathbf{x}_i$ it may then be eliminated.

**Phase-Lead Integrators**

The integrator of figure 3.2 [4] may be used instead of that of figure 3.1 when non-inverting integrators are needed.

An analysis [3] of the effects of finite amplifier gain-bandwidth product (usually a dominant problem) shows that this integrator's phase error is equal and opposite to that of a Miller integrator. Good systems containing equal numbers of positive and negative integrators may, to a first approximation, cancel the deleterious effects of finite amplifier bandwidth.

These circuits may be unstable with high-performance op-amps, which have gains with significantly second-order character near $\omega_t$. A compensation scheme is suggested in [5].

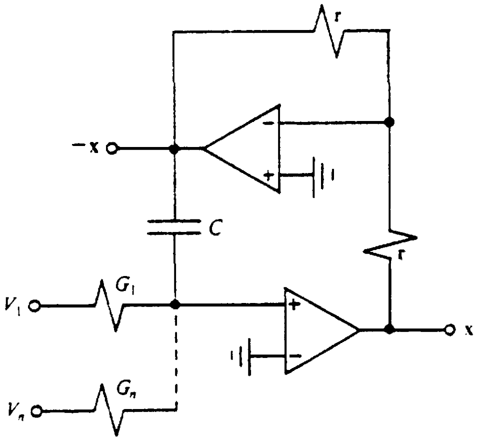


**Figure** 3.2: **Akerberg-Mossberg Integrator**

**DeBoo Integrators**

The circuit of figure 3.3 [6] also implements a positive integrator, with only one op-amp where both earlier circuits needed two.

Unlike the previous circuits, however, this one can have a phase error induced by mismatching between passive elements. Since filter transfer functions are generally much more sensitive to integrator phase than to gain errors this circuit is not often suitable.

This circuit only provides one output, $\mathbf{x}_i$, and so cannot be used where any $\mathbf{A}_{ji}$ or $\mathbf{c}_i$ are negative.

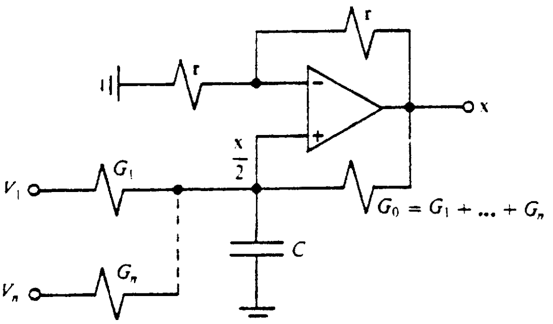


**Figure 3.3: DeBoo Integrator**

### RC Leaky Integrators

One often needs "leaky" integrators, i.e. those implementing $(-\frac{1}{s+\mathbf{A}_{ii}}\}$ rather than $\frac{1}{s}$, because a diagonal term in A is non-zero (and usually negative for syntheses of stable transfer functions with low sensitivity to integrators at low frequencies).

The circuit of figure 3.4 drives a virtual ground with the current

$$I_{o,s} = G_o \left| \frac{\sum_{i=0}^{N} G_i V_{i,s}}{sC + \sum_{i=0}^{N} G_i} \right|$$

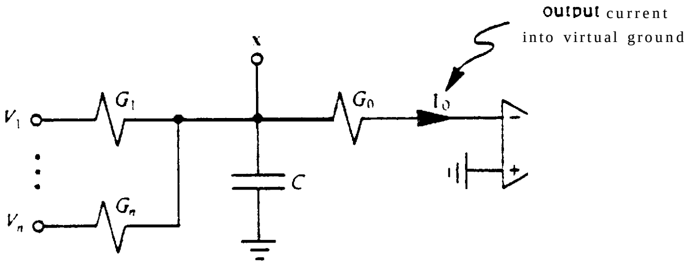


**Figure 3.4: RC 'Integrator'**

The conductance $G_0$ may if necessary be "split" to drive several virtual grounds; thus several non-zero entries (which must be positive, since $-\mathbf{x}_i$ is not produced) are possible in the corresponding column of A.

This type of "integrator" cannot (with positive $G_i$) ever have $\sum_{i=0}^{N} G_i = 0$, and so is always "leaky", It also cannot have voltage gain, so scaling will generally be necessary before it can be used.*

ln practice the conductances required to use this kind of "integrator" in a system may be much larger than those for the Miller integrators in the system, leading to problems with "element spread".

We show a detailed design example below which illustrates the issues of element spread, op-amp count, and sensitivity for a third-order synthesis.

### Using the Non-Inverting input of a Miller Integrator

The circuit of Figure 3.5 is a conventional Miller integrator as seen from the input at $V_a$,

$$t_{a,s}(s)\triangleq\frac{V_{o,s}}{V_{a,s}}=\frac{-1}{CR_a}\cdot\frac{1}{s}$$

and has a transfer function from input $V_b$ of

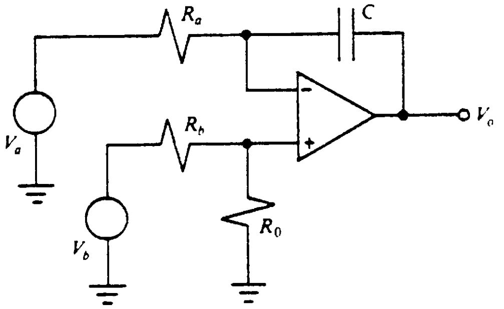


**Figure 3.5: Using the Non-inverting Input**

$$t_{b,s}(s)\triangleq\frac{V_{o,s}}{V_{b,s}}=\frac{R_o}{(R_o+R_b)CR_a}\cdot\frac{1}{s}+\frac{R_o}{R_o+R_b}$$

This is a positive integrator with a "parasitic" gain term $\frac{R_o}{R\mathrm{I}+R_b}$. Some modification of synthesis would be needed to include the effects of this type of term, but the result might be valuable: a one-op-amp integrator and summer with both positive and negative coefficients. The main difficulty in doing this is that it gives us two different kinds of "building blocks": one suitable for positive

* Note that the DeBoo integrator may be derived as a case of this kind of integrator in which a negative immittance converter [7] is used to create a negative resistor.

and one for negative coefficients. A similar situation will be encountered for switched-C inputs in section 3.3.

### 3.1.1 Design Example

This section illustrates the points made above by doing several fairly detailed active-RC designs of a simple $3^{rd}$ order Butterworth transfer function from a given state description.

The state description we choose is:

$$\mathbf{A}=\begin{bmatrix} -1 & 1 & 0 \\ -0.5 & 0 & 0.5 \\ 0 & -1 & -1 \end{bmatrix} \quad \mathbf{b}=\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad (3\text{-}1)$$

$$\mathbf{c}^T=\begin{bmatrix} 0 & 0 \end{bmatrix}. \quad d=0$$

As a matter of interest this is a description of the operation of the passive ladder of figure 3.6 when $V_{C_1}$,$I_{L_2}$, and $V_{C_3}$ are chosen to be states. The reader may have deduced this from the tridiagonal structure of A.

The system was chosen this way because the passive circuit is known to work well, and we wish to simulate it so as to avoid using the inductor.

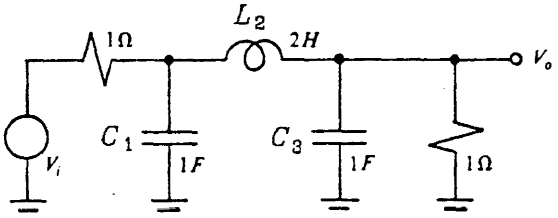
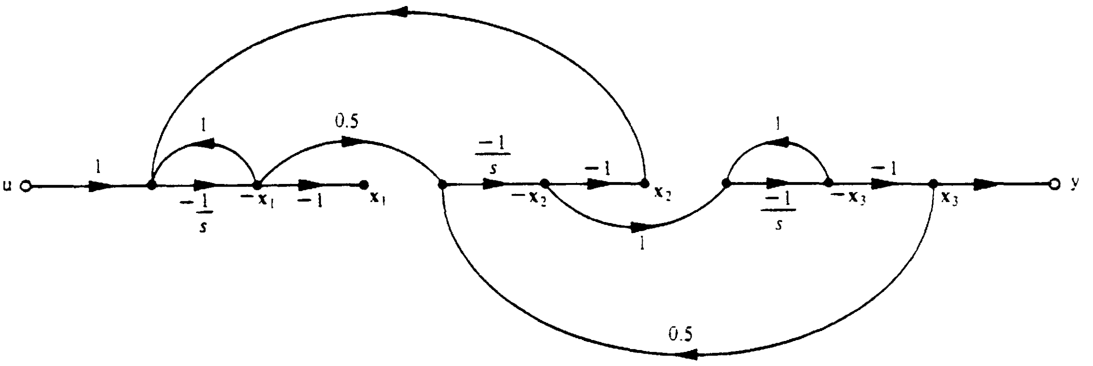


**Figure** 3.6: **LC circuit to be simulated**

#### Conversion to SFG

A signal-flow graph described by the **{A,b,c,d}** of equation (3-1) appears in figure **3.7, It** shows a straightforward decomposition into inverting integrators and



**l?igure 3.7: An SFG for Simulation**

inverting summers. Each row "$i$" of {A,b} corresponds to an integrator-inverter pair whose input $\mathbf{x}_i'$ is formed by summing $u_t$ (t) and the various $\mathbf{x}_j$. All input edge coefficients are positive (corresponding directly to feedback resistors with positive values ). Negative $\mathbf{A}_{ij}$ are handled by multiplying the inverted output "$-\mathbf{x}_j$" by $|\mathbf{A}_{ij}|$ .

The system output, which would in general be formed by a weighted sum, may be taken directly as $\mathbf{x}_3$ because of the simple structure of $\{\mathbf{c}^T, d\}$.

The corresponding circuit, which would use 2n=6 op amps, would be inefficient for several reasons, of which the most obvious is that the inverter in the first integrator does nothing.

### Minimizing Inverters

**It** doesn't really matter whether we simulate, say, $\mathbf{x}_2$ ***or*** $-\mathbf{x}_2$. ***By*** manipulating signs of states (a special case of scaling) one may reduce the number of inverters needed to one, thus obtaining a 4-op-amp circuit. One can usually reduce the number of op-amps needed to about $\frac{3}{2}n$ this **way.***

------

* $\frac{3}{2}n$ is usual for cascades of three-amplifler biquads and for SFG ladder simulations, where inductors require 2 op-amps and capacitors 1. It is not generally possible to reduce dense A

### Scaling

Scaling manipulations may be done directly on a circuit, on the SFG from which that circuit is derived, or on the {A,b,c,d} system that generates the SFG. The scaling rule on an SFG is that transmittances driving a node may be multiplied by a (thereby increasing the signal level on the node} without changing overall behaviour as long as all transmittances leading from that node are divided by $\alpha$.† Similarly, state "$i$" of an $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ description is scaled up by a factor a if row "$i$" of A and $\mathbf{b}_i$ are multiplied by a while column "$i$" of A and $\mathbf{c}_i$ are divided by $\alpha$. State $\mathbf{x}_3$ of system (3-1) may be scaled by -1 to yield

$$A = \begin{bmatrix} -0.5-1 & 01 & -0.50 \\ 0 & 1 & -1 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} 01 \\ 0 \end{bmatrix}$$

$$\mathbf{c}^T = [0 \quad 0 \quad -1] \quad d = 0$$

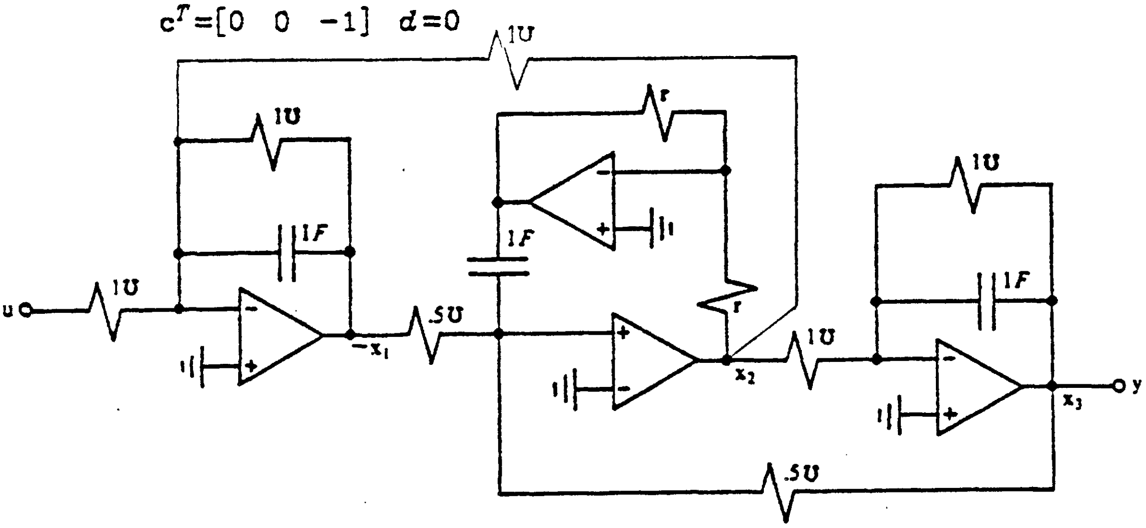


**Figure** 3.8: **Four-Op-amp Simulation** of a Third-order **Ladder**

**Now** columns 1 and **3** of A contain only negative coefficients and column **2**

matrices containing coefficients of arbitrary signs to $\frac{3}{2}n$.

† Another equivalent rule is that one may change gains arbitrarily as long as the gains around loops are left unchanged.

contains only positive ones, so that this system may be implemented with two inverting integrators and one non-inverting integrator. If Miller and Akerberg-Mossberg integrators are chosen the result will have 4 op-amps (cf. figure 3.8).

This is the kind of circuit obtained in [7] as an SFG simulation of a passive ladder.

**Using Passive Integrators**

Passive RC “integrators” may be used to simulate states when the corresponding columns of A have positive coefficients off the diagonal and negative coefficients on the diagonal. Scaling states 2 and 3 of system (3-1) by -1 yields

$$A=\begin{bmatrix} -1 & -1 & 0 \\ \\ 05 & -01 & -.1 \end{bmatrix} \quad \mathbf{b}=\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\mathbf{c}^T=[0 \quad 0 \quad -1] \quad d=0$$

Now states 1 and 3 may be implemented (except for scaling terms) by circuits like those in figure 3.4. State 2 must be realized by an active inverting integrator which provides at its output a voltage $V_2=-\mathbf{x}_2$.

Calculation of component values for the Miller integrator providing state 2 is done just as before, but component values for the passive “integrators” are a little more difficult. Since, for example, state 1 has to drive the virtual ground of state 2 with a current $(\mathbf{x}'_{2,t})$

$$\mathbf{A}_{21}\frac{\mathbf{A}_{12}\mathbf{x}_{2,s}+\mathbf{b}_1 u_s}{s-\mathbf{A}_{11}}=0.5\frac{V_{2,s}+V_{i,s}}{s+1}$$

$$=G_0\frac{G_2V_{2,s}+G_iV_{i,s}}{sC+G_0+G_2+G_i}$$ (cf. Fig, 3.9)

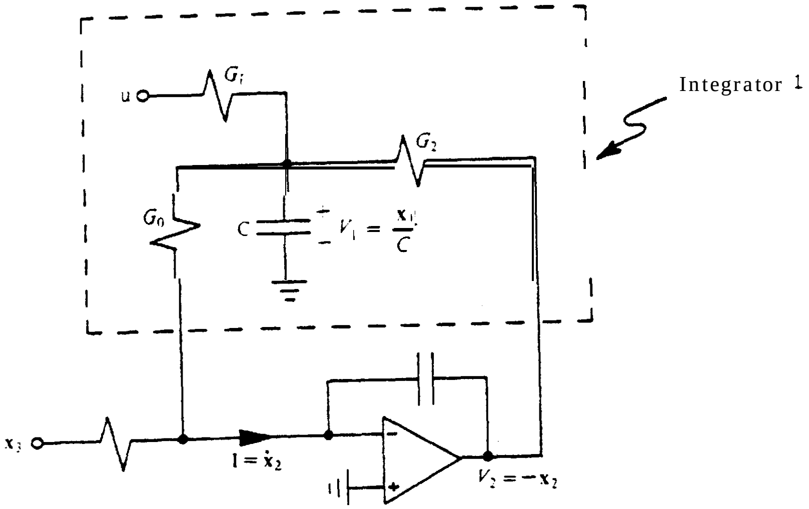


**Figure 3.9: Designing Integrator** 1

This may be solved by, for instance, arbitrarily choosing $G_i=1$. Comparing coefficients then yields

$$G_2=G_i=1 \tag{3-2}$$

$$\frac{G_0}{C}=0.5 \tag{3-3}$$

$$G_0+G_2+G_i=G_0+2=C \tag{3-4}$$

$$=>G_0=2,\ C=4$$

Now in fact, although this provides the correct signal to integrator 2, the capacitor voltage in this “integrator” will not be $\mathbf{x}_1$ but $\frac{\mathbf{x}_1}{C}$. This scaling is generally necessary because of the constraint that all conductances be positive, and will be seen to make "$\mathbf{x}_2$" more sensitive to errors in its op-amp than it would otherwise have been.

A similar technique was used to find a passive simulation of state 3, and the resulting overall circuit appears in figure 3.10. This is an interesting and novel single-amplifier implementation of a third-order transfer function, and has been derived systematically from a state-variable system description.

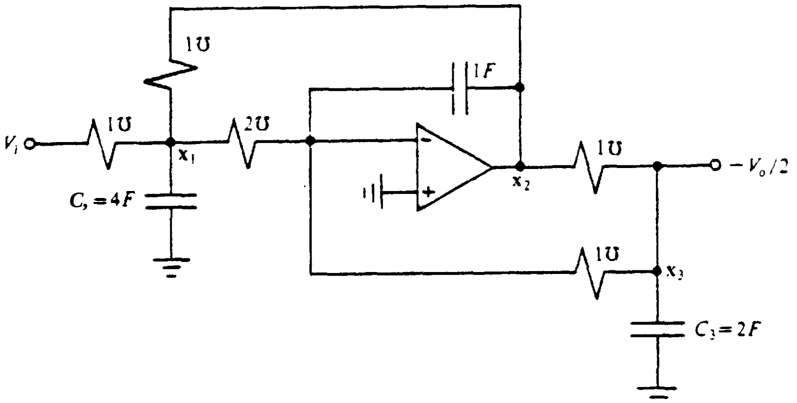


**Figure** 3.10: **One-Op-Amp Simulation of Third Order Ladder**

A system description of this circuit, showing the effects of scaling $\mathbf{x}_1$ and $\mathbf{x}_3$ is:

$$\mathbf{A} = \begin{vmatrix} -1 & \frac{1}{4} & 0 \\ -2 & 0 & -1 \\ 0 & \frac{1}{2} & -1 \end{vmatrix} \qquad \mathbf{b} = \begin{vmatrix} \frac{1}{4} \\ 0 \\ 0 \end{vmatrix}$$

$$\mathbf{c}^T = [0 \;\; \mathbf{0} \;\; -2] \quad d = 0$$

We will see in the next section that Miller integrators become more sensitive as the sum $\sum_j |\mathbf{A}_{ij}| + |\mathbf{b}_i|$ increases: note that this filter has a row-sum three times higher than that for the same integrator in the circuit of figure 3.8. Note also that the “element spread” is larger for the single-amplifier circuit than for that of figure 3.8, Which had all capacitors equal and a resistor spread of 2: 1.

Three- and two-op-amp designs in which only one of these two states (e.g. state 1, because it is not an output) is scaled for a passive integrator and the other is handled by an **AM** or a DeBoo integrator are also possible.

### 3.1.2 Capacitor Inputs

Some circuits (e.g. some notch biquads) are basically state-variable in form, but include capacitor (as well as resistor) inputs to integrators. This kind of circuit may be handled in several ways by intermediate-function synthesis, of which the easiest is an “augmentation”.

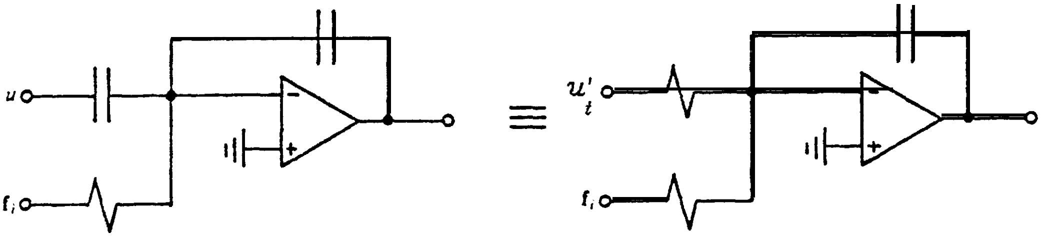


**Figure** 3.11: **Modelling Capacitor Inputs**

The effect of using a capacitor from a signal $u_t(t)$ as input to an integrator rather than a resistor is equivalent* (cf. Fig. 3.11) to that of using a resistor driven from a signal $u_t'(t)$ We can therefore model capacitor inputs by augmenting the set of signals available suitably with derivatives.

* for ideal q-amps.

## 3.2 Sensitivities

This section shows how component errors in the elements of the various building block" circuits above affect the coefficients of the systems they simulate. We will show in chapter 4 how coefficient errors in turn affect the system transfer function. These two components of sensitivity between them suffice to investigate circuit sensitivities in a way that clearly distinguishes problems caused by circuit choices from the effects of the choice of feedback structure.

### 3.2.1 Effect of Op-Amp Gain on the System

A reader familiar with filter circuits might find it surprising that the single-amplifier circuit above was so closely related to the 4 op-amp design. The same kind of operations applied to a second-order circuit, as in the closing example in chapter 2 yield a relation between a single-amplifier biquad (SAB) and a Tow-Thomas biquad, of which the SAB is a much lower-performance circuit.

We show here that the probIem with single-amplifier circuits comes from the scaling required to keep elements positive. All of the circuits in section 3.1 were equally sensitive to their "integrators", but the high-gain (i.e. short time-constant) integrator of figure 3.10 is more sensitive to its op-amp than are those of figure 3.8. This is why poorly scaled circuits, of which SABs are examples, are sensitive to their op-amps.

Note that this means that even high-performance topologies with $\frac{3}{2}n$ op-amps can be quite poor unless care is taken with their scaling.

The dominant problem caused by an op-amp in a filter comes from its finite gain at the frequency of interest, which in turn is often dominated by the amplifier's gain-bandwidth product. We will measure the performance of an op-amp integrator by grouping all effects on its transfer function into a factor $\gamma$ such that

$$\frac{V_o}{V_i}=\frac{1}{s\tau}\gamma$$

where $V_o$ and $V_i$ are the integrator's input and output voltages and $\tau$ is its nominal time-constant. $\gamma$ may often be a function of frequency and will usually be complex. It may be used to model both gain and phase errors in integrators, which result from non-idealities either in passive or active components.

We derive a formula giving

$$\frac{d\gamma}{d(1/A)}$$

for the Integrators of interest. This measure may be used together with formulae for $\frac{dt_s(s \, \mathrm{I}}{d\gamma}$ presented in chapter 4 to give the effect on a transfer function of op-amp gain errors. The quantity "l/A" is used because it is nominally zero while the more obvious ***A is nominally*** $\infty$, ***so*** that $\Delta(1/A)$ is generally small while AA is infinite. Thus a first-order sensitivity estimate of the effect of having a finite op-amp gain $A$ *in* the $i^{th}$ integrator on $t_s(s)$ may be written

$$\frac{\Delta t_s(s)}{t_s(s)}\cong\frac{1}{t_s(s)}\frac{dt_s(s)}{d\gamma_i}\frac{d\gamma_i}{d(1/A)}\Delta(1/A)$$

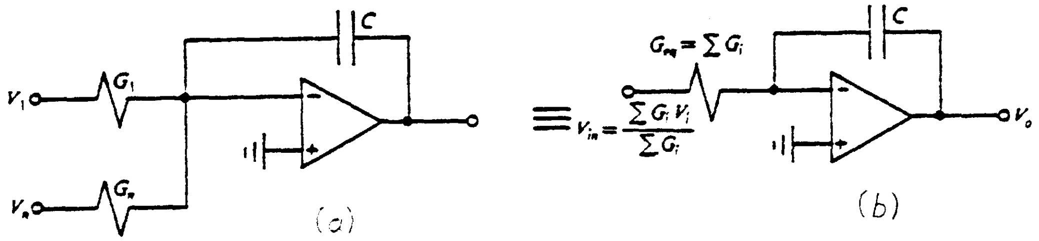


**Figure 3.12: Non-ideal Miller Integrator/Summer**

Analysis of the circuit of figure 3.12(a) reveals its equivalence to that o figure 3.12(b), for which

$$\frac{V_o}{V_i}=\frac{-1}{sCR_{eq}(1+(1/A))(1+1/sCR_{eq})} \tag{3-5}$$

$$\overset{\Delta}{=}\frac{-1}{sCR_{eq}}\cdot\gamma$$

The quantity of interest,$\gamma$, which is nominally 1 is affected by $\frac{1}{A}$ as follows:

$$\frac{d\gamma}{d(1/A)}=-\left[1+\frac{1}{sCR_{eq}}\right]\cdot\gamma^2$$

$$=-\left[1+\frac{1}{sCR_{eq}}\right] \quad \text{at } \gamma=1 \tag{3-6}$$

This-shows-clearly the effect of low $CR_{eq}$ (i.e. high integrator gain) on sensitivity.

**Example**

If the op-amp of figure 3.10 has $f_t$=3MHz, and the filter is frequency-denormalized to $f_{3dB}$=3kHz, then at the upper passband edge $\Delta(1/A)=0-\frac{s}{\omega_t}=-.001j$. At this same frequency

$$\frac{d\gamma}{d(1/A)}=-\left[1+\frac{1}{j\cdot 1/3}\right]$$

$$=-(1-3j)$$

Using techniques developed in chapter 4, we find that

$$\frac{dt_s(j\,1)}{d\gamma_2} = j\,1\mathbf{f}_2(j\,1)\mathbf{g}_2(j\,1) = -1$$

so that

$$\frac{\Delta t_s(j\,1)}{t_s(j\,1)} \cong (1-j)\cdot(-1+j3)\cdot(-j.\,001) = 0.004 - j.\,002$$

which will induce a magnitude error at the upper passband edge of

$$\Delta Att(\omega) = -8.68\Delta\ln|\,t_s(j\omega)| = -8.68\mathrm{Re}(\Delta\ln(t_s(j\,1))) \cong -.035dB$$

A well-scaled design, necessarily with more op-amps, would have a somewhat lower $S_{1/A}^{\gamma}$ and so a smaller deviation $\frac{\Delta t_s(s)}{t_s(s)}$. These effects would be more pronounced in a higher-Q example.

**Scaling to Minimize $\omega_t$ Sensitivity**

The critical term in (3-6) is $\frac{\hat{G}_{eq}}{sC} = \frac{1}{s}\sum_s \frac{\hat{G}_i}{C} = \frac{1}{s}\sum_j |\mathbf{A}_{ij}| + |\mathbf{b}_i|$. Thus one could minimize the worst-case op-amp sensitivity by minimizing $\max_i(\sum_j |\mathbf{A}_{ij}| + |\mathbf{b}_i|)$. Designs will be “good“ in the sense of best approximating their ideal “states” if row-sums $\sum_j |\mathbf{A}_{ij}| + |\mathbf{b}_i|$ are made approximately equal by scaling

## 3.2.2 Capacitor Sensitivities

All of the types of circuit discussed above are canonic in their capacitors, and have state values represented directly as capacitor voltages. For this reason all of them implement integrators such that $S_{C_i}^{\gamma} = -1$. Chapter 4 shows how to compute $S_{\gamma}^{t_s}(s)$.

### 3.2.3 Feed-in Sensitivities for Miller and AM Integrators

The conductances of the input resistors of Miller and Akerberg-Mossberg integrators directly implement $\mathbf{A}_{ij}$ entries, so $S_R^{\mathbf{A}_{ij}}=-1$. Chapter 4 shows how to compute $S_{\mathbf{A}_{ij}}^{t_s}(s)$.

### 3.2.4 Resistor Sensitivities for RC 'Integrators'

The capacitor voltage $\mathbf{x}_j$ for a leaky integrator like that of figure 3.4 is

$$V_{s,C_j}=\frac{\sum\limits_{i\neq j} G_i V_{i,s}}{sC+\sum\limits_{i=0}^{N} G_i} \qquad (3\text{-}7)$$

where we want

$$V_{s,C_j}=\mathbf{x}_j=\frac{\sum\limits_{i\neq j} \mathbf{A}_{ij} V_{i,s}}{s-\mathbf{A}_{jj}} \qquad (3\text{-}8)$$

Clearly, $S_{G_i}^{\mathbf{A}_{ij}}=1$ $(j\neq i)$, and because $\mathbf{A}_{jj}=\sum G_i$, $S_{G_i}^{\mathbf{A}_{jj}}=\frac{G_i}{\sum G}$. As long as all $G_i$ are positive, $S_{G_i}^{\mathbf{A}_{jj}}< 1$.

### 3.2.5 Resistor Sensitivities for theDeBoo Integrator

The op-amp in a DeBoo integrator is used to implement a negative-impedance converter (NIC) [6,7'] that simulates a negative resistance as shown in figure 3.13.

The result is a generalization of the RC integrator that allows $\mathbf{A}_{jj}=0$. On the other hand, when $\mathbf{A}_{ii}=0$, $\sum G_i=0$ (cf. equations (3-7) and (3-8)) and

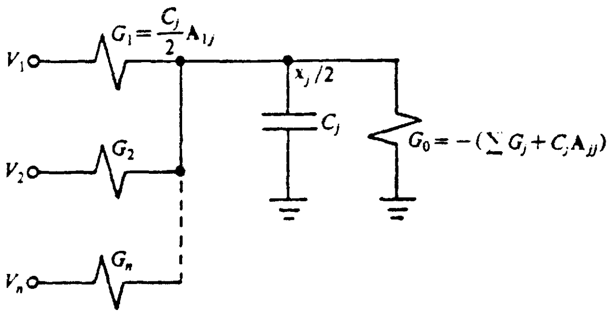

$V_1$
$G_1 = \frac{C_j}{2} A_{1j}$
$x_j/2$
$C_j$
$G_0 = -(\sum G_j + C_j A_{jj})$
$V_2$
$G_2$
$V_n$
$G_n$

**Figure 3.13: Equivalent ofDeBoo Integrator**

$$S^{A_{jj}}_{G_i} = \frac{G_i}{\sum G} = \infty \quad !$$

Classical sensitivity is a poor way to discuss effects on coefficients that are nominally zero, More interesting is the derivative

$$G_i \frac{\partial A_{jj}}{\partial G_i} = \frac{-G_i}{C} = -A_{ij}$$

This just expresses the problem mentioned earlier of phase sensitivity of the DeBoo integrator in state-space terms.

## 3.3 Discrete-Time Filters

There are many different ways to implement discrete-time filters, with correspondingly many different levels of performance. The synthesis method of this thesis may be used for discrete-time filters by discussing “delays” $(z^{-1})$ or “discrete integrators” $((z-1)^{-1})$ rather than “integrators” $(s^{-1})$. The same procedures that calculate $A_{ij}$ etcetera for the s-plane case calculate them for the $z$ and $z-1$ planes.

Analysis tools are also similar for the two cases, except that the various integrals involved in computing dynamic range and sensitivity are taken around the unit circle rather than along the imaginary axis.

### 3.3.1 Digital Filters

Intermediate-function synthesis works for digital filters, but the resulting emphasis on delays seems less natural than the emphasis on integrators for continuous-time systems, because in digital filters delay is cheap and coefficients (multiplies) expensive while in active-RC circuits coefficients are generally cheaper than integrators.

In many implementations of digital filters, however, rounding and clipping occur at the inputs to delays because data are stored with fewer bits of precision than are available in the multiplier/accumulator that forms weighted sums. When the data are fixed-point in this kind of scheme it does not even matter if overflow occurs during formation of a sum [8] as long as the final result is in range. In these practical and important cases intermediate-function synthesis is as natural for digital as for analog filters.

### 3.3.2 Filters Using Analog Delays

When filters are to be constructed using analog charge storage to implement $z^{-1}$, intermediate-function synthesis is directly applicable. Here the emphasis on delay elements is very natural because errors in delays dominate performance.

Two examples of this type of system are bucket-brigade devices [14] (BBDs) and charge-coupled devices (CCDs). In both of these technologies (for which dynamic range is the dominant problem) dynamic range is set by the maximum charge availabIe on storage capacitors and by noise added to the stored charge between cycles, while problems like charge-transfer inefficiency and charge leakage limit both transfer function accuracy and operating frequency range. Leakage may be modelled as a simple gain error in a delay: leakage causing, for example, 1% “droop” on stored signals just replaces $z^{-1}$ with $.99z^{-1}$. The result is that of setting $\gamma_i$=0.99 for that delay, where 7 is used for delays just as it was for integrators in section 3.2 above. Charge-transfer inefficiency produces a frequency-dependent gain error, or a shift in $z$: transfer inefficiency of $\varepsilon$ causes a delay to have gain $(z+\varepsilon)^{-1}$. These problems are exactly analogous to the gain and phase errors that cause Miller integrators to have transfer functions of the forms $\gamma s^{-1}$ and $(s+\varepsilon)^{-1}$.

BBDs and CCDs have usually had transversal structures: intermediate-function synthesis might be a useful tool in trying to find other structures for which their performance would be better, and so increase their range of application.

### 3.3.3 Switched-Capacitcr Filters

There are several types of switched-capacitor filters [9], all of which are basically state-variable circuits and so potentially amenable to intermediate-function synthesis. Of these, the class that appears to be most practical using present technology has the peculiar property that "integrator" transfer functions with positive and negative signs are quite different. This makes the generality of IF synthesis hard to obtain: in fact we will show that it is no longer true for this class of circuit that any arbitrary set $\{\mathbf{f}_i\}$ may be obtained.

Stray-insensitive building blocks [9] for switched-C filters are shown in figure 3.14.

These blocks have the practical advantage of being insensitive to the nonlinear stray capacitances to ground associated in integrated circuits with MOS switches and capacitor plates [10]. Three types of input network to the op-amp are shown, together with the transfer functions from each to the output. For frequencies much smaller than the sampling frequency, where $z \cong 1$, the two switched inputs are approximately equal and opposite. This is the case in which approximation of a switched input as a resistor is fairly accurate, and the two types simulate positive and negative resistors. At these frequencies it is reasonable to expect that arbitrary $\{\mathbf{f}_i\}$ are possible, or at least that for any given choice of $\{\mathbf{f}_i\}$ some "nearby" $\{\tilde{\mathbf{f}}_i\}$ will be attainable. At frequencies near the

**Second-order Problem**

A second-order inverting bandpass Butterworth transfer function, $t_{B,s}\triangleq\frac{-s}{s^2+\sqrt{2}s+1}$, after transformation to the discrete-time domain by the bilinear function $s=\frac{z-1}{z+1}$, is

$$t_z=\frac{-1}{2+\sqrt{2}}\frac{z^2-1}{z^2+\alpha}$$

where $\alpha=\frac{2-\sqrt{2}}{\text{- G E *}}$ We would like a synthesis of this transfer function with a two-op-amp switched-C circuit using the inputs shown in figure 3.14. We wish to use a structure like that of the Tow-Thomas biquad, which is known to be good. Thus the second amplifier is to be a non-inverting switched-C integrator, while the first is to have the desired bandpass $t(z)$ at its output. The inputs to amplifier $A_1$ are to be determined to make this possible.

In principle three kinds of "integrator" inputs are possible (cf. figure **3.14):** "co-p hase" switched-C $(\frac{V_{o,z}}{V_{b,z}}\triangleq t_{b,z}=\frac{-z}{z-1}\frac{C_b}{C})$; "anti-phase" switched-C $(\frac{V_{o,z}}{V_{a,z}}\triangleq t_{a,z}=\frac{I}{z-1}\frac{C_a}{C})$; and simple capacitor inputs $(\frac{V_{o,s}}{V_{c,s}}\triangleq t_{c,s}=\frac{-C_c}{C})$. The three possibilities are not, however, independent because [12] (putting $C_a=C_b=C_c$ for convenience):

$$t_{a,z}+t_{b,z}=\frac{1}{z-1}-\frac{z}{z-1}$$

$$=\frac{1-z}{z-1}$$

$$=t_c.$$

Thus a parallel combination of a "co-phase" input with capacitor $C_b$ and an "anti-phase" one with gain $C_a$ where, say, $C_b>C_a$ may be replaced by a combination of an unswitched input $C_c=C_a$ and a co-phase one with capacitor $C_b-C_a$.

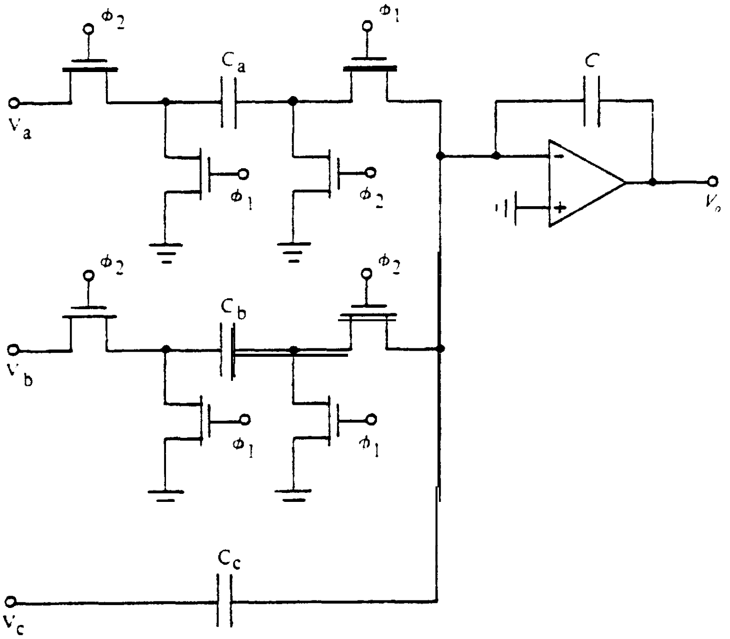


**Figure 3.14: A General Switched-C Building Block**

sampling rate, however, the situation need not be so good.

Biquadratic sections with arbitrary transfer functions may be obtained [IO] and specialized types of ladder simulation done [11,121 with these mismatched integrators, but it is not certain that high-performance structures may be produced for arbitrary transfer functions. Since dynamic range is already a problem in this technology these limitations may be serious.

We will illustrate these problems by attempting to design a second-order filter with a reasonable choice for $\{f_i\}$, which will turn out to be impossible. We will show how to modify $\{f_i\}$ so that the design is possible, and discuss the implications for system performance of these changes. This example will also illustrate the manipulations necessary to design for given $\{f_i\}$ even when it is possible.

When $C_a > C_b$ an unswitched capacitor and an anti-phase input are similarly produced.

Because of this equivalence we concern ourselves only with two types of input during synthesis, ignoring the unswitched C. The circuit equivalence may be applied to any resulting design to save switches and minimize total capacitance.

In order to obtain the desired Tow-Thomas-like structure we must have

$$\mathbf{f}_{1,z} = t_z = \frac{-1}{2+\sqrt{2}} \frac{z^2-1}{z^2+\alpha}$$

$$\mathbf{f}_{2,z} = \mathbf{f}_{1,z} t_{a,z} = \frac{-1}{2+\sqrt{2}} \frac{z+1}{z^2+\alpha}$$

where $t_{a,z}$ is the transfer function of the “non-inverting integrator” building block, $\frac{1}{z-1}$.

**Signals at integrator inputs**

With the single type of integrator heretofore assumed available, synthesis proceeded by calculating, from the given $\{\mathbf{f}_i\}$, the set of signals required at the inputs to integrators $(\{s\,\mathbf{f}_i\})$ and solving for the combination of available signals $(\{\mathbf{f}_i\}$ and $u_s)$ needed to produce them. When two types of “integrator” transfer function, $t_{a,z}$ and $t_{b,z}$ are involved a slight modification is required.

For this case we deem the op-amp input signals to be $(z-1)\mathbf{f}_{i,z}$ and need to find ways to produce them from the signals of the form $-z\,\mathbf{f}_{i,z}$ or $-zu_z$ (co-phase inputs) and $\mathbf{f}_{i,z}$ or $u_z$ (anti-phase inputs), Twice as many types of signals are available to choose from, but all coefficients are constrained to be positive. While in the ordinary case there must be a unique set of coefficients that solve the problem, in this case there may be none or many. The larger set of signals opens up the possibility that infinitely many solutions will exist while the constraint that they must all be positive makes it possible to have none.

For this particular example it turns out that no solution exists. Let us demonstrate this by searching for a solution for the input of amplifier 1,

$$(z-1)\mathbf{f}_{1,z}=-z^3+z^2+z-1/e_z$$

(where $e=(2+\sqrt{2})(z^2+\alpha)$) which must be composed of a weighted sum $\sum \mathbf{a}_i\sigma_i$ of the six signals (which we denote $\sigma_{i,z}$) available:

$$\sigma_{1,z}=u_z=z^2+\alpha/e_z$$

$$\sigma_{2,z}=-zu_z=-z^3-\alpha z/e_z$$

$$\sigma_{3,z}=\mathbf{f}_{1,z}=-z^2+1/e_z$$

$$\sigma_{4,z}=-z\,\mathbf{f}_{1,z}=z^3-z/e_z$$

$$\sigma_{5,z}=\mathbf{f}_{2,z}=-z-1/e_z$$

$$\sigma_{6,z}=-z\,\mathbf{f}_{6,z}=z^2+z/e_z$$

Comparing coefficients on the various powers of z in the numerator results in the four simultaneous equations

$$-1=-\mathbf{a}_2+\mathbf{a}_4$$

$$1=\mathbf{a}_1-\mathbf{a}_3+\mathbf{a}_6$$

$$1=-\alpha\mathbf{a}_2-\mathbf{a}_4-\mathbf{a}_5+\mathbf{a}_6$$

$$-1=\alpha a_1+a_3-a_5$$

These are four equations in 6 unknowns, and so have infinitely many solutions. None of these solutions, however, has all coefficients positive. We may see this by systematically simplifying. Combining the second and third equations to eliminate $a_6$,

$$a_5=-a_4+a_3-\alpha a_2-a_1 \qquad (3\text{-}9)$$

Combining this with the fourth equation to eliminate $a_5$ yields

$$a_4=-\alpha a_2-a_1-1 \qquad (3\text{-}10)$$

But now this shows that no solution in positive $a_i$ is possible, because the right-hand side must be strictly negative and the left positive.

This has shown what was promised: that it is not always possible to use the building blocks of figure 3.14 to produce completely arbitrary $\{f_i\}$. In addition, the example chosen is quite a reasonable one, so that it appears that the weakness of this selection of blocks is significant.

It is possible to produce arbitrary transfer functions with these building blocks, since a general biquad exists [10]: they just fail to offer the kind of flexibility needed to produce arbitrary structures. It is an open question whether or not they can nonetheless always produce structures that are “good enough”.

One of the original switched-C approaches used [13], instead of the co-phase input, an input like that in figure 3.15.

This has a transfer function exactly the negative of the “anti-phase” type, and so is capable of generating completely arbitrary $\{f_i\}$. With inexact design methods (like resistor equivalences) this exactness is no real advantage, and even a cause of some inaccuracy [15] because the “phase errors” of co- and anti-phase inputs approximately cancel and so lessen the errors induced by the design method. When using 1F synthesis, of course, design is exact even with $\frac{1}{z-1}$ building blocks.

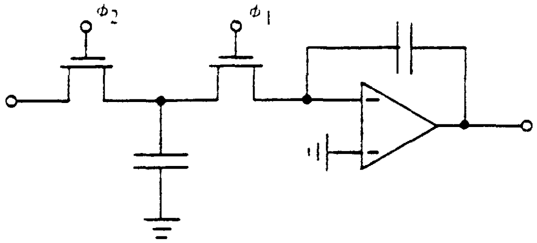


**Figure 3.15: Alternate Inverting Switched-C Integrator**

The switched-C input of figure 3.15 is not, however, fully stray-insensitive since it is affected by switch capacitances. It is insensitive to “bottom-plate” parasitics [9] which are the dominant kind.

## 3.4 Summary and Conclusions

We have shown that it is fairly easy to proceed from an abstract $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ system description to circuits whose structure closely corresponds with that of the system. Under various restrictions on $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ it is possible to use very simple “building blocks”, like passive RC integrators. A new type of single-amplifier third-order section was systematically developed in this way.

The forms that restrictions on the use of certain blocks take can suggest the root cause of performance difficulties with some structures: thus we showed a relationship between a Tow-Thomas biquad and a relatively low-performance single-amplifier biquad that turn out to differ only in the scaling of their internal variables.

We were able to use the method to produce an unusual type of result: a proof that no simulation of a particular ladder filter was possible using a popular switched-C technology. This type of power is useful when one wishes to explore the fundamental limits to a new realization technology.

The IF synthesis technique may be used to do practical design, to demonstrate the relationships among apparently different circuits, and to investigate theoretical properties of circuit approaches to filter implementation.

## 3.5 References

[1] G.4. Korn and T.M. Korn, "Electronic ***Analog Computers (dc Anolog Computers)"*** McGraw-Hill 1956

[2] 0. Wing, "Ladder Network Analysis by Signal ***Flow Graph – Application to Analog Computer Programming",*** IRE Trans. CT-3, pp. 289-294, Dec. ***1956***

[3] P.0. Brackett and A,S. Sedra, ***"Active Compensation*** *GOT* ***High- Frequency Effects in op- amp circuits with Applications to Active- RC*** Filters" IEEE Trans. Circuits and Systems, CAS-23, pp.68-73, Feb. 1976

[4] D. Akerberg and K. Mossberg ***"A Versatite Active- RC BuildingBlock with Inherent Compensation*** *for* ***the Finite Bandwidth of the Amplifier",*** IEEE Trans. Circuits and Systems, CAS-22, pp. ***407-415,*** May 1975

[5] K. Martin and A. Sedra, "On ***the Stability of the Phase- Lead Integrator",*** IEEE Trans. Circuits and Systems, CAS-24, pp.321-324 June 19'7'7

[6] R.W. Newcomb, ***Active Integrated Circuit Synthesis*** Prentice-Hall, 1968

[7] A.S. Sedra and P.0. Brackett, ***Fitter Theory and Design: Active and Passive*** Matrix Publishers, Champagne, Illinois, 1978

[8I L.B. Jackson, "On ***the Interaction of Roundoft Noise and Dynamic Range in Digital Filters***" Bell Syst. Tech. J., v. 49, No. 2, pp. 159-184,, **1970**

[9] G-C. Temes, MOS ***Switched- Capacitor Filters – History and St&e of the Art",*** European Conf. Circuit Theory and Design, The Hague, The Netherlands, August 1981, pp.l76-185

[10] K. Martin and A.S. Sedra, ***"Exact Design of Switched- Capacitor Bandpass Filters Using Coupled- Biquad Structures",*** IEEE Trans. Circuits and Systems, CAS-27, pp. 469-475, June 1980

[11] P.E. FIeischer and K.R. Laker ***"A Family of Switched- Capacitor Biquad Building Blocks"*** BelI Syst. Tech. J., vol 58, pp. 2235-2269, Dec.1979

[12] M.S. Ghausi and K.R. Laker, ***Modem Filter Design, Active RC and Switched Capacitor,*** Prentice-HalI, 198 1

***[13]*** D.L. Fried, 'Analog Sampled- data Filters", IEEE JSSC, SC-7, pp.302-304, Aug. 1972

[14] C.H. Sequin and M.F. Tompsett, **Charge** ***Transfer Devices,*** Academic Press, NY, 1975

[15] R.W. Brodersen, P.R. Gray and D.A. Hodges, "MOS ***Switched- Capacitor Filters", Proc.*** IEEE, vol. 67, pp. Jan. 1979

# 4. Filter Performance

This chapter shows how some important measures of the “practicality” of a filter design may be related to intermediate functions. We do this so that a designer can compare different approaches easily, without having to investigate them on the component level.

We will concentrate on simple formulae that measure dominant problems. Thus, for instance, sensitivity to time-constant variation will be discussed more fully than sensitivity to $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ coefficient mismatch. Two general performance areas are addressed in some detail: sensitivity and dynamic range. Several other areas of interest (e.g. tuning, propagation of non-linear distortion and minimization of the range of component values required) will be mentioned in relation to these dominant concerns.

## 4.1 Intermediate Functions

The two interesting vectors of functions $\mathbf{f}_s$ and $\mathbf{g}_s$ were defined in terms of $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ in chapter 2. They could be interpreted as

$$\mathbf{f}_{i,s} = \frac{\mathbf{x}_{i,s}}{u_s} \tag{4-1}$$

and

$$\mathbf{g}_{i,s} = \frac{y_s}{\varepsilon_{i,s}} \tag{4-2}$$

where $\varepsilon_{i,s}$ is the Laplace transform of a disturbance signal injected at the input, of integrator "i", according to a version of the standard state equations including disturbance terms as follows:

$$\mathbf{x}'_t=\mathbf{A}\mathbf{x}_t + \mathbf{b}u_t + \begin{bmatrix}\varepsilon_{1,t}\\ \varepsilon_{2,t}\\ \cdot\\ \cdot\end{bmatrix} \tag{4-3}$$

$$y=\mathbf{c}^T x+d\cdot u \tag{4-4}$$

These functions, when combined in various ways, may be used to predict many of the important indicators of filter performance and practicality.

## 4.2 Derivative Sensitivity

The vectors $\mathbf{f}_s$ and $\mathbf{g}_s$ combine in various ways to measure sensitivities.

The formula we use to relate **f** and g to sensitivity is (cf. Fig 4.1)

$$\frac{dt_{ab}}{dt_{mn}}=t_{am}\cdot t_{nb} \tag{4-5}$$

where $t_{am}$, *e.g.,* is thc transfer function from point “a” to point "m".

This formula appeared originally in [1,2] and was derived in terms of Tellegen’s theorem, but we give a simpler derivation of it here that will help to show the close relationship between the sensitivity and dynamic range problems.

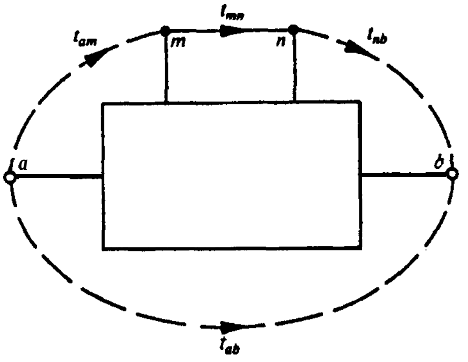


**Figure 4.1: Investigating one arm of an SFG**

Figure 4.2 shows a system like that of Fig. 4.1 with a gain perturbation *At,,,,,* added to the transmittance of interest. The signals appearing at various nodes when the system input is $u_s$ are marked.

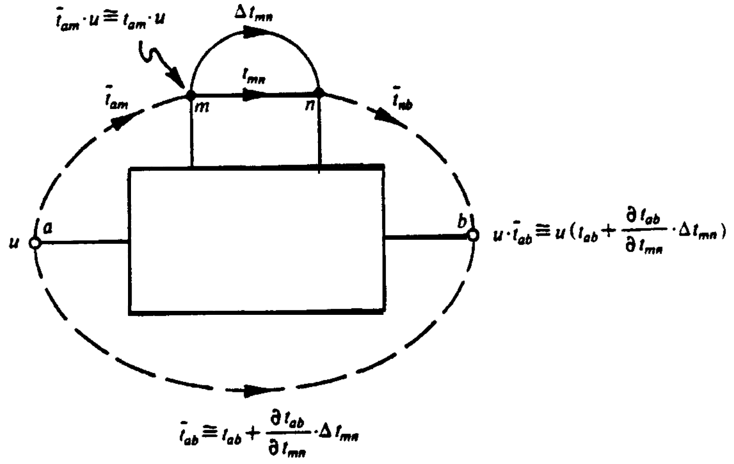

$\bar{t}_{am}\cdot u \cong t_{am}\cdot u$
$\Delta t_{mn}$
$t_{mn}$
$\bar{t}_{am}$
$m$
$n$
$\bar{t}_{nb}$
$u$
$a$
$b$
$u\cdot\bar{t}_{ab} \cong u(t_{ab}+\frac{\partial t_{ab}}{\partial t_{mn}}\cdot\Delta t_{mn})$
$\bar{t}_{ab}\cong t_{ab}+\frac{\partial t_{ab}}{\partial t_{mn}}\cdot\Delta t_{mn}$

**Figure 4.2: Perturbed SFG**

Now one may compare this with the SFG of figure 4.3, which has (instead of a gain perturbation in $t_{mn}$) an extra signal injected at node N identical to that injected by the arm modelling gain error in Fig. 4.2.

Now signals reaching the rest of the system are unchanged between the two SFG's, and all other signals and gains are unchanged, so the two systems must (to a first-order approximation) have the same output $y_s$. Comparing the two expressions for $y_s$ gives equation 4-5.

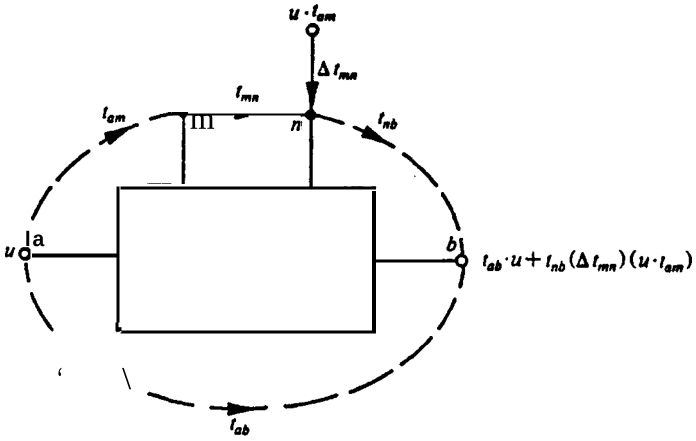


**Figure 4.3: SFG with two sources**

‘**The** relationship between figures 4.2 and 4.3 illustrates the close relationship between sensitivity problems (in which gain errors like that of Fig. 4.2 are considered) and dynamic range and distortion problems (in which unwanted signals are injected at various nodes as in Fig. 4.3).

## 4.3 Sensitivity Measures

Simple derivative measure like $\frac{\partial t}{\partial\mu}$ are not often the most natural ones to choose to measure the significance of component errors. Component tolerance is more often expressed as fractional change $\frac{\Delta\mu}{\mu}$, so that the classical sensitivity measure

$$S_\mu^{t_s}=\frac{\partial t_s}{\partial\mu}\cdot\frac{\mu}{t_s}=\frac{\partial\ln t_s}{\partial\ln\mu} \tag{4-6}$$

is more useful. In filter stopbands, where $t_s \to 0$, a hybrid of the two common forms is more easily interpreted:

$$S_\mu^{t_s}\triangleq\frac{\partial t_s}{\partial\ln\mu} \tag{4-7}$$

If, for instance, $S_\mu^{t_s}=1$ and $\mu$ is changed by 1%, signal feedthrough of $\Delta t_s \cong .01$ is induced, which corresponds to only 40dB of attenuation.

Note that (4-7) is almost equal in magnitude to (4-6) in passbands (where $|t_s(j\omega)|\cong 1$), so that if $|S|$ is the quantity of interest (4-7) might as well be used both for pass- and stop-bands.

$S_\mu^{t_s}$ is a complex function of frequency, and so contains information about sensitivity of magnitude *and* phase of $t_s$ to magnitude *and* phase of $\mu$. Because

$$\ln(\mu)=\ln|\mu|+j\varphi(\mu)$$

we can interpret the real and imaginary parts of $S_\mu^{t_s}=\frac{\partial(\ln t_s)}{\partial(\ln\mu)}$ as follows:

$$\mathrm{Re}(S_{\mu}^{t_s})=\frac{\partial \ln|t_s|}{\partial \ln|\mu|}=S_{|\mu|}^{|t_s|} \tag{4-8}$$

$$=\frac{\partial \varphi(t_s)}{\partial \varphi(\mu)}$$

$$\mathrm{Im}(S_{\mu}^{t_s})=-\frac{\partial \ln|t_s|}{\partial \varphi(\mu)} \tag{4-9}$$

$$=\frac{\partial \varphi(t_s)}{\partial \ln|\mu|}$$

Although phase errors in $t_s$ are often unimportant, note that both Re($S$) and Im($S$) measure sensitivity of $|t_s|$ to some type of error in the value $\mu$: magnitude and phase respectively. For some components (e.g. resistors at audio frequency) $\Delta\varphi(\mu)$ is negligible in comparison to $\Delta \ln|\mu|$; but for others (like Miller integrators), the two terms are of comparable importance. In the latter case $|S_{\mu}^{t_s}|$ takes both types of effect into account in **a** balanced way.

## 4.4 Sensitivity Formulae

This section uses the derivative sensitivity formula of section 4.2 to find $S^{t_s}$ for various system components in terms of the design vector $\mathbf{f}_s$ and the related $\mathbf{g}_s$. These sensitivities of $t_s$ to system coefficients may be used together with the sensitivities of system coefficients to circuit components developed in Chapter 3 to analyze the performance of circuits.

Sensitivity formulae are also useful in tuning filters: by correlating the measured transfer function error with first-order sensitivities one may obtain strategies for adjusting a selected set of coefficients to minimize error.

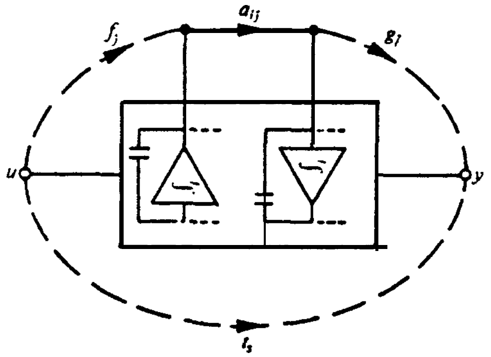


**Figure 4.4: A-matrix sensitivity**

Let us start with $S^{t_s}_{\mathbf{A}_{ij}}$, the system sensitivity to the $\mathbf{A}_{ij}$ coefficient. In the system this coefficient interconnects the output of integrator "j" and input of integrator "i", as shown in Figure (4.4). According to equation (4-5):

$$\frac{\partial t_s}{\partial \mathbf{A}_{ij}} = t_{am} \cdot t_{nb}$$

But $t_{am,s} = \mathbf{f}_{j,s}$ and $t_{nb,s} = \mathbf{g}_{i,s}$. It follows that

$$S^{t_s}_{\mathbf{A}_{ij}} = \mathbf{g}_i \mathbf{f}_j \frac{\mathbf{A}_{ij}}{t_s} \tag{4-10}$$

Table *4.1* summarizes $t_{am}, t_{nb}$ and $S^{t_s}$ for the other system coefficients.

| Coefficient | $t_{am}$ | $t_{nb}$ | $S^{t_s}$ |
|---|---|---|---|
| $b_i$ | 1 | $\mathbf{g}_{i,s}$ | $\mathbf{g}_i \cdot \frac{b_i}{t_s}$ |
| $c_i$ | $\mathbf{f}_{i,s}$ | 1 | $\mathbf{f}_i \cdot \frac{c_i}{t_s}$ |
| $d$ | 1 | 1 | $\frac{d}{t_s}$ |

Table 4.1: Coefficient Sensitivities

One other interesting sensitivity measure is $S_{\gamma_i}^{t_s}$, the sensitivity of $t_s$ to a gain (or time-constant) error in an integrator (cf. section 3.2.1).

This may be computed by finding the sensitivity of $t_s$ to errors in an SFG edge of gain $\gamma_i$ (nominally equal to 1) inserted in cascade with integrator i as shown in Figure 4.5. This edge corresponds to the extra factor 7 introduced in chapter 3 into the expression for integrator gain: $\frac{1}{s\tau}\cdot\gamma$. We use it as a convenient tool for modelling integrator errors.

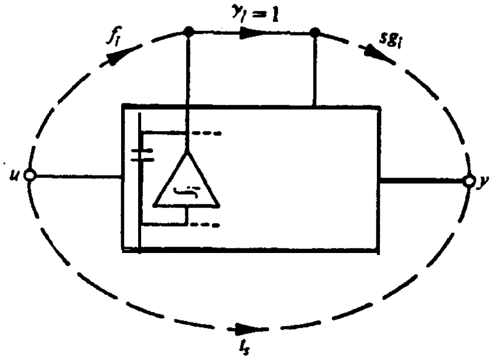


**Figure 4.5: Integrator Sensitivity**

For this case $t_{am,s}=f_{i,s}$ and $t_{nb,s}=s\,g_{i,s}$ (where the factor s appears because $g_{i,s}$ is the gain from the *input,* not output, of integrator i). Thus

$$S_{\gamma_i}^{t_s}=f_i g_i\cdot\frac{s}{t_s} \qquad (4\text{-}11)$$

(no term in $\gamma_i$ appears because it was defined to have nominal value 1).

______________________________ - - - ________________ - - -

*Example (4.1)*

*The* third-order Butterworth filter of section *3.1* was designed by choosing $\{f_i\}$ to be the transfer functions from the input to each state (capacitor voltage or inductor current) of the circuit of figure *3.6.* These functions are

$$f_{1,s}=\frac{V_{C1,s}}{V_{in,s}}=\frac{(s+0.5+j0.5)\cdot(s+0.5-j0.5)}{e_s}$$

$$f_{2,s}=\frac{I_{L2,s}}{V_{in,s}}=0.5\cdot\frac{s+1}{e_s}$$

$$\mathbf{f}_{3,s}=\frac{V_{C3,s}}{V_{in,s}}=\frac{V_{o,s}}{V_{in,s}}=0.5\cdot\frac{1}{e_s}$$

where

$$t_s\triangleq\frac{p_s}{e_s}=\frac{0.5}{(s+1)(s+0.5+j0.866)(s+0.5-j0.866)}$$

From the formulae for **f** and g ( 2-3 and 2-4 ) or by the methods of chapter 6 one may find that

$$\mathbf{g}_{1,z}=0.5\cdot\frac{1}{e_s}$$

$$\mathbf{g}_{2,z}=-1\cdot\frac{s+1}{e_s}$$

$$\mathbf{g}_{3,z}=\frac{(s+0.5+j0.5)(s+0.5-j0.5)}{e_s}$$

We can therefore use (4-10) to find that, e.g.,

$$S_{\mathbf{A}_{11}}^{t_s}=\mathbf{g}_1\cdot\mathbf{f}_1\cdot\frac{\mathbf{A}_{11}}{t_s}$$

$$=\frac{(0.5)(s+0.5+j0.5)(s+0.5-j0.5)(-1)}{p_s\cdot e_s}$$

A plot showing $\left|S_{\mathbf{A}_{11}}^{t_s(j\omega)}\right|$ and $\mathrm{Re}(S_{\mathbf{A}_{11}}^{t_s})=S_{\mathbf{A}_{11}}^{|t_s(j\omega)|}$ appears in figure 4.6. Inspection of the curve for $S_{\mathbf{A}_{11}}^{|t_s|}$ reveals a maximum value of about 0.75 near $s=j\,1$, the passband edge. This may be interpreted according to the formula $\Delta Att(\omega)\cong 8.68\left|\frac{\Delta t}{t}\right|\cong 8.68 S_{\mathbf{A}_{11}}^{|t_s|}\cdot\frac{\Delta\mathbf{A}_{11}}{\mathbf{A}_{11}}$ to find that a 1% error in $\mathbf{A}_{11}$ causes a 0.065dB error in attenuation at the passband edge.

It is interesting that, because of the special nature of this system, $\mathbf{f}_1=\mathbf{g}_3$ and $\mathbf{f}_3=\mathbf{g}_1$. It foIIows that $\mathbf{f}_{1,s}\mathbf{g}_{1,s}=\mathbf{f}_{3,s}\mathbf{g}_{3,s}$ and so that $S_{\mathbf{A}_{11}}^{t_s}=S_{\mathbf{A}_{33}}^{t_s}$; i.e. that errors in the terminations have identical (to first order) effects.

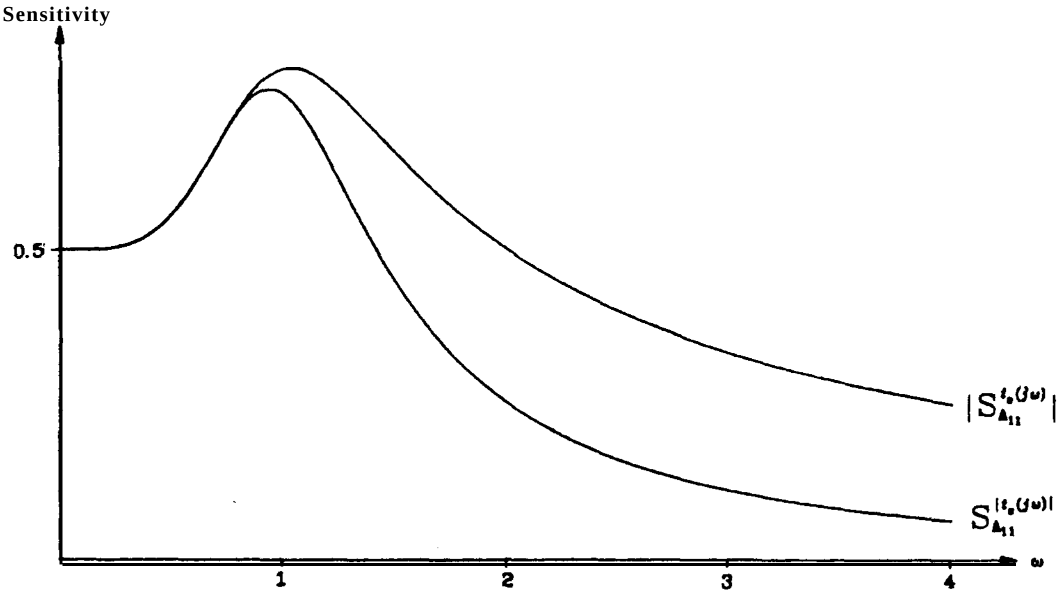


**Figure 4.6: Sensitivity to $A_{11}$**

Figure 4.7 shows plots of the real and imaginary parts of integrator

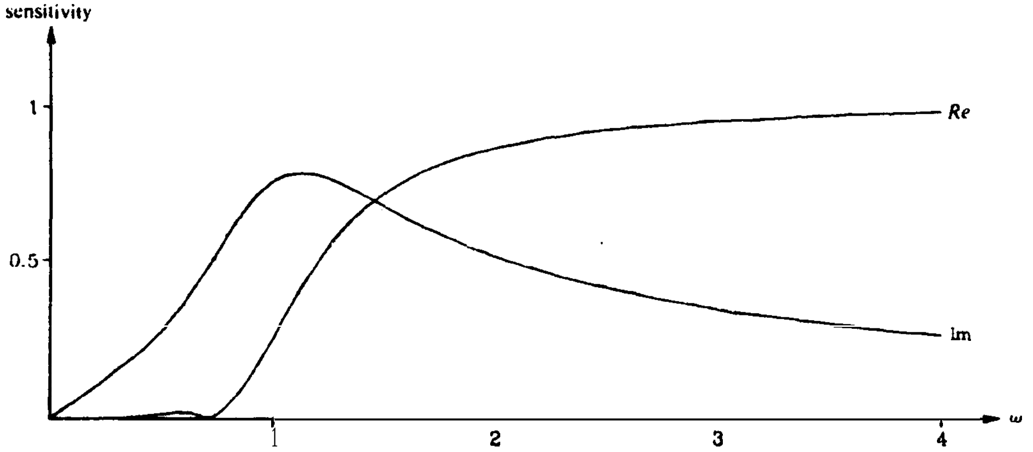


**Figure 4.7: Magnitude Sensitivity to Integrator Gain and Phase**

sensitivities $S_{\gamma_i}^{t_s}=f_i g_i \times \frac{s}{t_s}$, for integrators 1 and 2. Again, because of the symmetry that makes $f_{1,s}\, g_{1,s} = f_{3,s}\, g_{3,s}$, $S_{\gamma_3}^{t_s}=S_{\gamma_1}^{t_s}$. Notice (equations 4-6 and 4-9) that the real and imaginary parts measure, respectively, the sensitivity of $|t_s|$ to gain and phase errors in integrators.

---

**Summary**

**We** have shown that extremely simple formulae connect $\mathbf{f}_s$ and $\mathbf{g}_s$ with sensitivities to coefficients and integrator gains in the system description; the circuits in chapter 3 had simple relationships between component sensitivities (e.g. sensitivities to op-amps) and coefficient sensitivity. Thus $\mathbf{f}_s$ and $\mathbf{g}_s$ are the essential functions for describing sensitivity of state-variable circuits.

## 4.5 Aggregate Measures

One is usually interested in sensitivity functions in order to know the overall (or "aggregate") effect of random component errors on system performance There are several ways of measuring the aggregate effect.

The two most common forms of aggregate are worst-case and statistical types. A worst-case measure might be, e.g.,

$$S^{t_s}_{A,\infty}(\omega) \triangleq \max_{i,j} | S^{t(j\omega)}_{A_{ij}} |$$

Note that we use the subscript $\infty$ here to indicate that we have defined a type of "infinity norm", which takes a maximum over the items of interest. This type of notation will appear in more detail when we discuss dynamic range.

Measures like this frequently appear in the literature, but lack rigorous justification. The best that can be said for them is that they are easy to evaluate and that the resulting figures are often larger for bad filters than for good ones.

A statistically based aggregate measure might look like

$$S_{\mathbf{A},2}^{|t_s|}(\omega) \triangleq \sqrt{\sum_{i,j} (S_{\mathbf{A}_{ij}}^{|t(j\omega)|})^2}$$

The "2" subscript here denotes a "squared" norm just as the "$\infty$" earlier denoted a "maximum" norm.

This type of measure lets one relate the statistics of a filter's overall performance to the statistics of its components: if, for instance, all $\mathbf{A}_{ij}$ have independent Gaussian statistics and standard deviations 1% of nominal values, we could deduce from a curve showing that $S_{\mathbf{A},2}^{|t_s|}$ was equal to 3 at some frequency $\omega$ that the standard deviation of $|t_s|$ resulting from variations in $\mathbf{A}_{ij}$ would be 3% of $|t_s|$.

An extensive literature is available on "multi-parameter sensitivity measures" [3], which are variations on these $S_2$ measures that suitably weight contributions of several effects. If one knows the statistical properties of the various components of a filter (e.g. resistors and integrators) one may construct a weighted measure of the general sum-of-squares type that predicts the variability of $t_s$ due to variations in all its components.

An important point raised in chapter 7 is that systems that are nearly optimum in performance under any of these measures are good for any other measure of this general type. For this reason the concept of a "good" filter is reasonable? because "goodness" is not a strong function of the details of a figure of merit.

---

*Example* (4.2)

For the third-order running example we can find, say,

$$S_{\gamma,2}^{t_s}(\omega) \triangleq \sqrt{\sum_i \left| S_{\gamma_i}^{t(j\omega)} \right|^2}$$

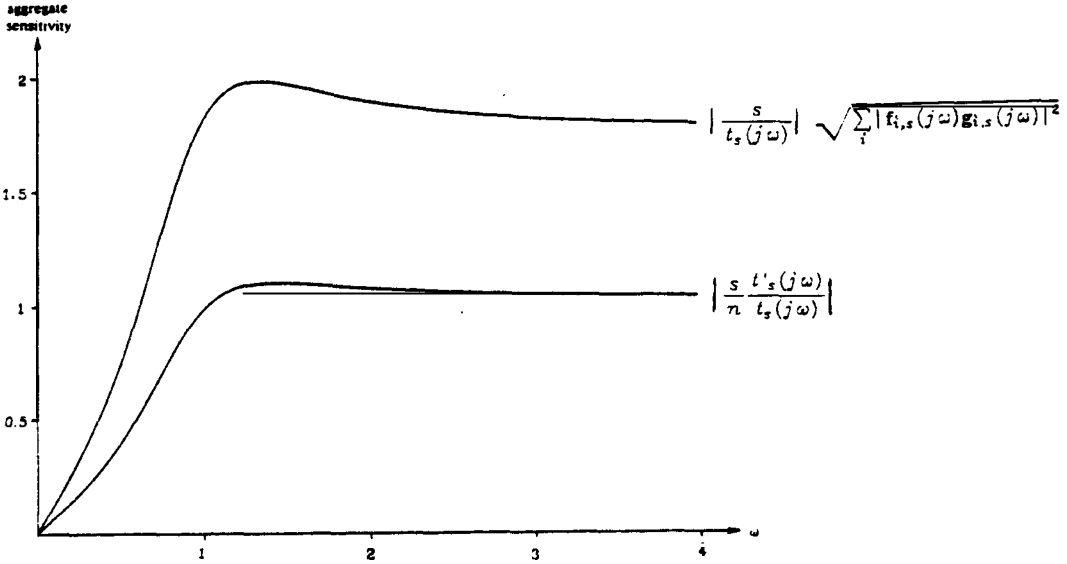


**Figure 4.8: An Aggregate Sensitivity Measure**

$$= \left| \frac{s}{t_s} \sqrt{\sum_i \mathbf{f}_{i,s}^2(j\omega)\mathbf{g}_{i,s}^2(j\omega)} \right|$$

which is plotted in F'igure 4.8 together with a lower bound (derived in chapter 7) on this quantity. A scalar value could be derived (as a figure of merit) as some suitably weighted integral of this curve.

From the curve we may deduce, for instance, that if all three integrators have complex gain factors $\gamma_i$ (cf. section 4.4) with standard deviations of 1% then the overall transfer function will have an error with a standard deviation of about 2% near the upper end of the passband.

## 4.6 Dynamic Range: Analysis

By "dynamic range" we mean the ratio between the largest and smallest signals that a system can accommodate. The "largest" signal for which the system works properly is usually determined by amplifier saturation or arithmetic overflow, while the "smallest" useful signal is determined by electrical noise or arithmetic quantization effects.

Note that the term “dynamic range” is occasionally used in the literature to refer to what we call “signal swing”, the largest possible signal value. Our usage is consistent with [4].

The reason that the dynamic range problem is particularly difficult in filter design (as compared to amplifier design) is that internal components can limit signal swing and generate noise? and all of these intermediate signal values and noise gains are related to the input signal and system gain by different $n^{th}$ order transfer functions.

Dynamic range has been fairly thoroughly studied for digital filters [5,6,7]. This section shows the results as they relate to our synthesis method and for the less thoroughly studied continuous-time case.

There are two kinds of things that we must be able to do about dynamic range in filter design: analyze a design to find its dynamic range, and synthesize designs that have good dynamic range. This section discusses the analysis problem, and sections 4.7 and 4.8 discuss two synthesis approaches.

### 4.6.1 Mathematical Measures of Signal and Noise

There are infinitely many ways of measuring the magnitude of a signal in order to evaluate dynamic range, and the most appropriate choice will generally depend on details of how the filter is to be used – what kinds of signal and noise it will be subject to and how its performance will be measured.

The situation is analogous to that for measuring aggregate sensitivity (cf. section 4.5): several mathematical measures are possible for the collection of functions representing integrator output signals and noise gains. Some measures will be appropriate in some situations and some in others. Even the types of measures reasonable in different situations are strongly analogous to those needed for analogous sensitivity problems, because the basic mathematical problem is to measure the lengths of vectors in both cases.

Most of the important types of signal and noise discussed below fit reasonably well into the category of signals with known spectrum and Gaussian statistics. For this class of signals it is natural to use “root-mean-square” (or "$L_2$", see below) magnitude measures, which are the ones that we prefer. The naturalness comes from the fact that root-mean-square measures for intermediate functions allow us to compute output signal statistics from input statistics. We will, however, briefly discuss the alternatives.

A whole class of “norms” or measures for intermediate-function magnitudes exists. In the digital-filter case the $L_p$ measures of functional analysis [8]

$$\|f\|_p \triangleq \left[\frac{1}{\omega_s}\int_0^{\omega_s} |f(\omega)|^p \, d\omega\right]^{1/p}$$

have been studied. This class of norms includes $L_2$, which measures “root-mean-square” values; $L_\infty$, which measures maximum value; and $L_1$, which measures average absolute value. The one we emphasize is $L_2$, which is the best model to use for noise-like signals because it is concerned with noise power.

The issue of choosing "$p$" in the definition of an $L_p$ measure was touched on in section 4,5 where we were concerned with aggregate measures of sensitivity performance, which are also types of norm. We will often meet with situations in which this choice comes up, and will therefore often use the concepts of “infinity norms” (those involving maxima) “2-norms” (involving rms values) and “l-norms” (involving sums of magnitudes).

As an application to digital filters, Jackson [6] showed for the digital-filter case that estimates of the magnitude of a product $\mathbf{x}_{i,s}=\mathbf{f}_{i,s}u_s$ could be made according to

$$|\mathbf{x}_{i,t}| \leqq \|\mathbf{f}_{i,s}\|^p \|u_s\|_q, \text{ where } \left|\frac{1}{p}+\frac{1}{q}\right| = 1$$

This means that if we know an $L_q$ norm on a signal $u$, and wish to avoid clipping by forcing $|\mathbf{x}_{i,t}|<M$ (where $M$ is the clipping level), we should use scaling to force $\|\mathbf{f}_i\|_p \leqq \frac{M}{\|u_s\|_q}$.

### 4.6.2 Limits to Signal Swing

We will pay most attention to signal Ievels at integrator outputs $\mathbf{x}_i$, because in the type of circuit in which we are most interested [cf. section 3.11 all amplifier outputs are either $\mathbf{x}_i$ or $-\mathbf{x}_i$, and "clipping" of signals is the dominant limiting mechanism.

Another important type of amplifier overload is "slew-rate limiting", [9]. This mechanism may be regarded as limiting derivatives $\mathbf{x}'_{i.t}$.

We are therefore interested in the magnitudes of $\mathbf{x}_s = \mathbf{f}_s \cdot u_s$ (to investigate "clipping") and perhaps in the magnitudes of $s\mathbf{x}_s = \mathbf{f}_s \cdot s \cdot u_s$ (for slew-rate limiting).

### 4.6.3 Properties of the Input Signal

The signal appearing at the output of an integrator depends on two things: the input signal $u$ and the appropriate "intermediate transfer function" $\mathbf{f}_i$. In order to "scale" a filter to avoid clipping we must regulate the magnitudes of integrator output signals by suitable choice of $\{\mathbf{f}_i\}$: in order to know the best way to choose magnitudes of $\{\mathbf{f}_i\}$ **we** must use whatever information we can find about the input signal u.

What we know about u at the time a filter is designed depends strongly on the application. Some examples are:

1. swept-frequency systems: $u_t$ is known to consist primarily of a sinusoid of known amplitude occuring somewhere in a band of frequencies. The natural measure is $\max_{\omega \epsilon (a,b)} \mathbf{f}_{i.s}(j\omega)$, which is just the highest gain for a sinusoid in the appropriate range. This is an $L_\infty$ measure on the *frequency* response. This type of measure, which is easy to evaluate, is suggested for analog filters in [10].

ii. pulse-shaping: $u_t$ is known in advance. The most appropriate measure of $\mathbf{x}_i$ is $\|\mathbf{x}_i\|_\infty \triangleq \max_t \left| \int_0^\infty \mathbf{f}_{i,t}(\tau) u_t(t-\tau) d\tau \right|$ an $L_\infty$ measure on the time-domain response which lets us stop the intermediate signals from clipping. This may also be regarded as a norm on $\{\mathbf{f}_i\}$.

iii. bounded input: a magnitude bound (and nothing more useful) is known for $u_t$, e.g. because it is known to be the output of an op-amp or a digital-to-analog converter. Thus, if we know a bound, $|u_t| \leq M$, we can say that $|x_{i,t}| \leq M \int_0^\infty |\mathbf{f}_{i,t}(t)|\, dt$. This maximum magnitude for $|x_{i,t}|$ is attained for the special case when the input signal is a clipped, time-reversed, version of the intermediate-function impulse response, i.e. $u_t = M sgn(\mathbf{f}_{i,t}(-t))$. This means that if we scale so that $\int |\mathbf{f}_{i,t}(t)|\, dt \leq 1$ we can guarantee, e.g., that no input from an op-amp (i.e. bounded by the power supplies of a circuit) can "clip" the outputs of op-amp integrators, i.e. demand $\mathbf{x}_{i,t}$ outside the supplies. This is probably an overly conservative type of scaling, but is an example of the use of an $L_1$ measure.

iv. known spectrum: an average power spectrum is known for u, which is "wide-sense stationary" [ 111. Speech and music signals may reasonably be treated this way. This is the case for which some type of $L_2$ measure is most appropriate. If we denote the power spectrum of a signal u as $S_u$, then we can write

$$S_{x_i} = |\mathbf{f}_{i,s}|^2 S_u$$

to deduce the spectral properties of $\mathbf{x}_i$ from those for u.

### 4.6.4 Noise Sources

Analog signals are corrupted by various sources of noise within a filter. These sources may arise from several types of effect and are accordingly characterized in different ways, and may be injected at various points within a filter.

Digital signals are not usually considered to be affected by noise in this same way, but the rounding operations usually necessary in implementing digital filters are often modelled as noise sources.

### Where Noise is Injected

In a continuous-time analog system thermal, shot and flicker noise are contributed by all active devices and resistors and interfering signals are coupled into the circuit in various ways [12]. We can usually treat the overall effect as equivalent to that of injecting noise only at the inputs of integrators. It is also usually, but not always, reasonable to assume that these noise sources are independent: thermal noise sources are quite independent of each other, but interference effects can be highly correlated.

The effects of noise signals injected at integrator inputs may be modelled by equation (4-3), which added a noise vector $\boldsymbol{\varepsilon}$ to the system equations. The $\{\mathbf{g}_i\}$ functions give the gain to the system output for each $\varepsilon_i$, so that we may write the system's output signal as a sum of a signal and noise:

$$y_s = t_s u_s + \varepsilon_{o,s}$$

$$\overset{\Delta}{=} t_s u_s + \sum_i \varepsilon_s \mathbf{g}_{i,s}$$

### How Noise Affects the Output

The total effect of all of the intermediate noise sources in a filter will be to add **a** certain noise signal to the system output. One might be interested in the variance of the noise signal (as it represents a measurement error), its subjective effects on **a** signal to be interpreted by people, or even the maximum possible value of error. Subjective effects are often modelled by the signal/noise ratio at the output of a frequency-weighting network that models human sensitivity, and so can be regarded as a mean-square type of measure.

If, for instance, rms noise level at the output (noise variance) is of interest then an appropriate way to measure $\varepsilon_o$ is to take $\|\varepsilon_o\|_2 = \left[\int_{-\infty}^{\infty} |\varepsilon_{o,s}(j\omega)|^2 d\omega\right]^{1/2}$. If, on the other hand, we wish to measure the subjective effect of the output noise we should include a suitable frequency-weighting function $W(\omega)$ into the norm, producing $\|\varepsilon_o\|_2=\left[\int_{-\infty}^{\infty}|\varepsilon_{o,s}(j\omega)|^2W(\omega)d\omega\right]^{1/2}$. "A"-weighting and "c-message"-weighting are practical examples of this kind of measure.

**Types of Noise**

We can distinguish two important types of noise as mathematically distinct:

i. noise of known power spectrum (e.g. Johnson and shot noise). $L_2$ measures are appropriate for noise gains $\{g_i\}$. If, for instance, $\varepsilon_i$ are all white with power spectral density $S_{\varepsilon_i}=1$, then the output noise power will have spectral density $S_{\varepsilon_o}=\sum_i|g_{i,s}(j\omega)|^2$. This in turn will have rms value $\sum_i\int_{-\infty}^{\infty}|g_{i,s}(j\omega)|^2d\omega$†, so that the natural choice of measure for $\{g_i\}$ is $\|g_i\|_2=\left[\int_{-\infty}^{\infty}|g_{i,s}(j\omega)|^2d\omega\right]^{1/2}$. We can then simply say that white integrator noise of unit power spectral density results in an output noise level of $\sum_i\|g_i\|_2^2$.

ii. noise of bounded amplitude (e.g. "roundoff noise" and some types of "flicker noise"). An $L_1$ measure $\left(\int|g_t(t)|\,dt\right)$ could be used here to predict the largest possible error in the output. Thus, e.g., if we know (only) that $|\varepsilon_{i,t}|\leq 1$, we can say that the output noise signal $\varepsilon_{o,t}(t)=\sum_i\int_0^{\infty}\varepsilon_{i,t}(t-\tau)g_{i,t}(\tau)d\tau$ cannot exceed $\sum_i\|g_i\|_1$ where $\|g_i\|_1\triangleq\int_0^{\infty}|g_{i,t}(\tau)|d\tau$. We could in principle use this kind of scaling to design so that we could guarantee that the output signal was never in error by more than a stated amount, so long as the input noise remained bounded by a given figure.

Even with input noise known to be bounded in magnitude, $L_2$ measures may be used to predict output noise variance from a noise power spectrum.

† We use a two-sided definition for consistency with the material of chapters 8 and 9.

This information may well be more useful than the conservative type of bound implied by the use of an $L_1$ measure.

These types of noise are analogous to two of the types of signal discussed earlier. It is also possible to use "infinity norms" like $||\mathbf{g}_i||_\infty \triangleq \max_\omega (\mathbf{g}_{i,s}(j\omega))$, but there is no obvious justification except that the figure is easy to compute and visualize.

## 4.7 Dynamic Range: Scaling

One is generally free to set the signal level at the output of an amplifier arbitrarily by scaling operations (cf. Chapter 3). If too high a level is chosen the amplifier will occasionally clip or slew-rate limit, while if too low a level is chosen the ratio of signal to amplifier noise will be low. Thus a trade-off between clipping and noise exists at every amplifier.

By "scaling" a set of $\{\mathbf{f}_i\}$ we mean producing from them a new set $\{\tilde{\mathbf{f}}_i\}$ that differ only by constant factors, i.e.

$$\tilde{\mathbf{f}}_i = \alpha_i \mathbf{f}_i$$

so that the magnitudes of $\{\tilde{\mathbf{f}}_i\}$ are what we want. Our usual reason for doing this is to avoid "clipping" the integrator outputs $\{\tilde{\mathbf{x}}_i\}$. In the terms of section 4.6, this means forcing $||\tilde{\mathbf{x}}_i|| \leq M$ for some suitable norm on $\{\mathbf{x}_i\}$ and a clipping limit M. We will show one way to choose a good M later in this section.

Section 4.6 discussed how to get $||\tilde{\mathbf{x}}_i||$ from information about the input signal u and the $\{\mathbf{f}_i\}$ in different situations. Using this information, one may do scaling by putting

$$\tilde{\mathbf{f}}_i = \frac{M}{\|\tilde{\mathbf{x}}_i\|}\mathbf{f}_i$$

Further, because we are generally free to define norms on $\{\mathbf{f}_i\}$ etcetera to suit ourselves, we can usually simply say

$$\tilde{\mathbf{f}}_i = \frac{\mathbf{f}_i}{\|\mathbf{f}_i\|}$$

so that for a properly scaled filter $\|\tilde{\mathbf{f}}_i\|=1\,\forall i$. Note that this simple expression for the magnitudes of $\{\mathbf{f}_i\}$ in a properly scaled filter assumes that a careful job has been done of choosing a suitable norm for intermediate functions: one that includes all information known about the input signal and about the “clipping” behaviour of integrators.

The widely-used [ 13,101 technique of scaling filters to have equal maxima of $\mathbf{f}_{i,s}(j\omega)$, for instance, is the special case of this in which a type of “infinity norm” is chosen on the s-domain representation representation of $\mathbf{f}_i$. We suggested above that this was properly justifiable only for swept-frequency type input signals: it tends to be used in other situations just because it is easy to compute.

We would next like to know what effect scaling has on filter performance figures other than probability of clipping: how it affects output noise, sensitivity, system structure and $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ coefficients. It is easy to show (as we do in section 6.8) that the effect of scaling $\{\mathbf{f}_i\}$ by an arbitrary set of factors $\alpha_i$ is to scale the $\{\mathbf{g}_i\}$ by the reciprocal factors $1/\alpha_i$. It also turns out that scaling $\{\mathbf{f}_i\}$ has no effect on the general structure of the system **{A,b,c,d}** or on sensitivities to $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ coefficients or to sensitivities to $\gamma_i$. Sensitivities to $\gamma_i$, for instance, are unchanged because they are proportional to $\mathbf{f}_{i,s}\mathbf{g}_{i,s}$, and one term is increased by scaling while the other is decreased an equal amount. Scaling does, however, affect the sensitivities of integrator gains $\gamma_i$ to operational amplifier non-idealities; it therefore has an important effect on the sensitivity of the overall filter to the op-amps used in implementing its integrators. This effect was mentioned in section 3.2.1 to explain why a realization using passive “integrators” should be expected to be sensitive to its op-amps.

Because it leaves the system structure unchanged, scaling is a particularly harmless operation (except as regards op-amp sensitivities), and in fact should be performed on any filter design before it is realized. It minimizes the output noise level obtained from any given structure with given characteristics of the input signal u because it makes $\{\mathbf{f}_i\}$ as large as possible before clipping sets in, which in turn makes $\{\mathbf{g}_i\}$ as small as possible because of the reciprocal relationship between scaling effects on $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$. This means that it maximizes the dynamic range of the structure. Well-scaled structures tend to have low integrator gains, and so improve op-amp sensitivity performance as well as dynamic range. We show results in chapter I that imply that systems with good dynamic range tend to have good sensitivity performance.

---

***Example*** **(4.3)**

Let us say, for the third-order Butterworth running example, that we expect the input signal to have a constant (white) spectral density of $1V/\sqrt{rad/s}$. If we choose an $L_2$ norm for $\{\mathbf{f}_i\}$ functions

$$\|\mathbf{f}_i\|_2=\left[\int_{-\infty}^{\infty}|\mathbf{f}_i(j\omega)|^2 d\omega\right]^{1/2},$$

we will be able to say that the rms signal levei at the output of each integrator is simply $\|\mathbf{f}_i\|_2$. For the three $\{\mathbf{f}_i\}$ taken directly as simulations of capacitor voltages and inductor currents, we can compute (by the methods of chapter *6)*

$$\|\mathbf{f}_1\|_2=1.618$$

$$\|\mathbf{f}_2\|_2=0.8862$$

$$\|\mathbf{f}_3\|_2=0.7236$$

If we decide, for instance, that a 1.5V rms level is acceptable at the output of each integrator,* we may scale $\{\mathbf{f}_i\}$ accordingly, getting e.g.

- we show how to choose this number in section 4.7.1

$$\tilde{f}_3 = \frac{1.5}{0.7236} f_3$$

$$\cong \frac{1.036}{e_s(s)}$$

This scaling reduces output noise gains $\{g_i\}$ by the same factors by which it increases signal Level, so that for this exampIe we could expect scaling to reduce the output noise contribution of, e.g., integrator 3 by a factor of almost 2.

This scaling makes the $\{g_i\}$ as small as alIowable for this structure and type of input signal, and so minimizes the noise output Because our scaling took the magnitude of the signal into account, this in turn maximizes the dynamic range of the structure.

Note that this approach to scaling is more general (since it is applicable to arbitrary structures), more exact (because it uses a realistic statistical model for the input signal rather than a swept-frequency type of model), and easier to understand than that presented in [10].

### 4.7.1 Choosing Signal Levels

When scaling is to be done using $L_1$ or $L_\infty$ measures to guarantee that clipping never occurs, the question of how large a signal to permit at each integrator output can be precisely answered. Signals are more often, however, only known statistically so that it may not be possible to set the probability of clipping to ***zero;*** further, even when it is possible to do so the resulting scaling may be so conservative with respect to clipping that its noise is unacceptable. When the probability of clipping is to be permitted to be non-zero the question of what value to choose for it lacks a precise answer.

This section shows a way to choose acceptable rms signal levels. Because noise and clipping are quite different phenomena, it is by no means obvious how to manage the trade-off between the two. One approach is to model the effects of clipping as a type of noise, and choose the signal level that maximizes the resulting signaI-to-noise ratio. This kind of approach seems natural enough in digital signal processing, where “noise” is already simply a way of modelling a non-linear effect – quantization – but is considerably less obvious in analog systems where the physics of noise are very different from those of clipping.

The “noise” source that models clipping (cf. figure 4.Q) is one that injects a noise voltage $\varepsilon_{c,t}(t)$ equal to the difference between an ideal system’s signal level $\mathbf{x}_{i,t}$* and the clipping level $V_c$ whenever $|\mathbf{x}_{i,t}(t)|>V_c$.

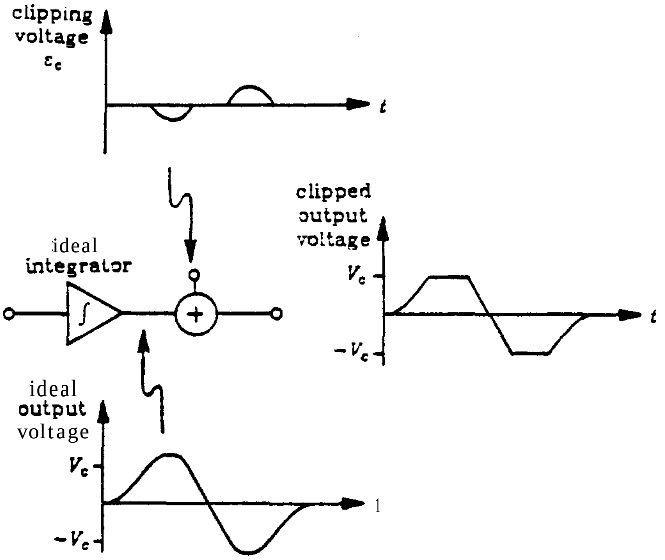


Figure 4.9: Modelling Clipping as Noise

Thus the probability distribution for $\varepsilon_{c,t}(t)$ can be computed from the “tail” of that for $\mathbf{x}_{i,t}(t)$ (cf. figure 4.10).

If the signal $\mathbf{x}_{i,t}(t)$ desired at an amplifier output has gaussian statistics and an rms level $\sigma$ (which we can set by scaling), the probability that $|\mathbf{x}_{i,t}|$ exceeds $V_c$ is

$$P(|\mathbf{x}_{i,t}(t)|>V_c)=21-\left(1/\sqrt{2\pi}\int_{\frac{V_c}{a}}^{\infty} e^{\frac{-v^2}{2}}dv\right)$$

$$\triangleq 2erfc\left[\frac{V_c}{\sigma}\right]$$

The $L_2$ norm of $\varepsilon_{c,t}(t)$ is the rms clipping noise voltage at the amplifier output.

* that is, the signal level which would have been attained if clipping were not to occur.

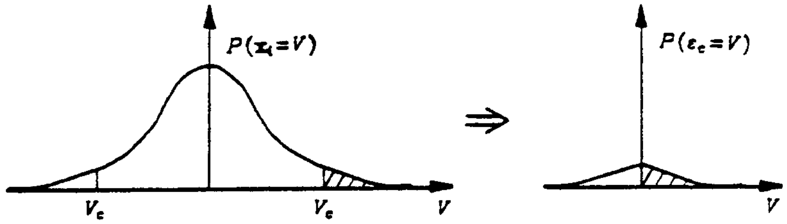


**Figure 4.10: The Probability Distribution of Clipping Noise**

It may be computed from

$$\|\varepsilon_c\|_2^2=2\int_{V_c}^{\infty}(V-V_c)^2P(V)dV$$

$$=\frac{2\sigma^2}{\sqrt{2\pi}}\int_{V_c/\sigma}^{\infty}(v-V_c/\sigma)^2e^{\frac{-\nu^2}{2}}d\nu \quad (\nu=V/\sigma)$$

$$=2\sigma^2\left[(1+(V_c/\sigma)^2)erfc(V_c/\sigma)-\left(V_c/\sigma\sqrt{2\pi}\right)e^{\frac{-V_c^2}{2\sigma^2}}\right] \qquad (4\text{-}12)$$

We **can** use this formula to predict the variation of “clipping noise” with scaling (i.e. changing $\sigma$). By simultaneously looking at the effect of scaling on $\{\mathbf{g}_i\}$, we can get two curves for “output noise” as a function of scaling. Figure 4.11 shows a typical pair of curves for the output effect of clipping noise $\varepsilon_i$ and integrator noise $\varepsilon_i$ as a function of scaling of the output signal level, which is measured as $\frac{V_c}{\sigma}$ – the ratio of maximum allowable signal to the rms level. Note that the total resulting noise is at a minimum when the two contributions to it are approximately equal. A curve like this was used in [5], where it suggested that the rms level be a factor of 3 below overload.

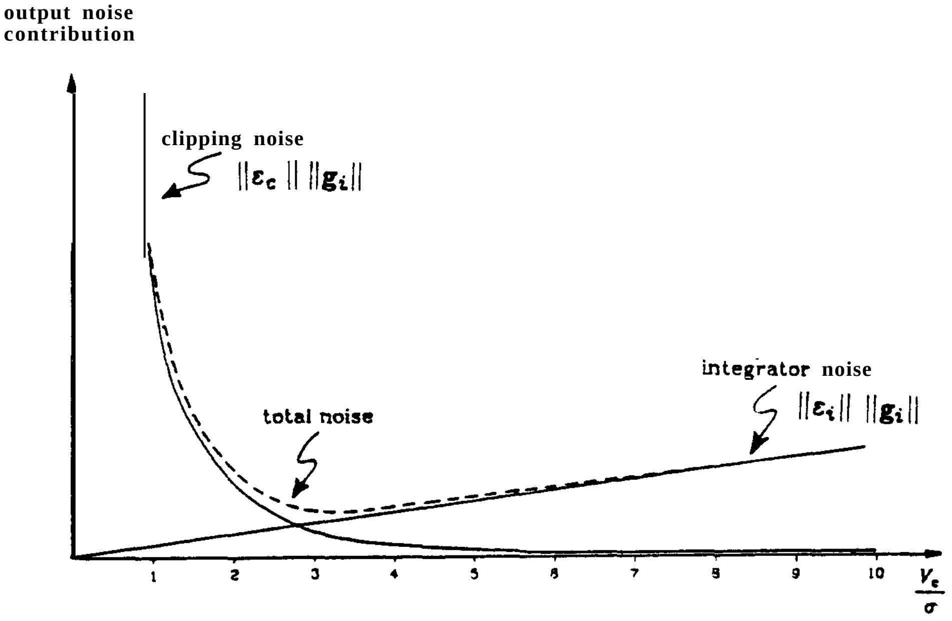


**Figure 4.11: Noise versus Scaling Level**

In general the right choice of $\frac{V_c}{\sigma}$ will depend on the noise characteristics of the integrator to be used: if integrators generate $1mV$ of noise in the passband we should choose a lower (less conservative) $\frac{V_c}{\sigma}$ (i.e. higher signal levels) than if they generate only $1\mu V$ of noise. As long as the two "noise" curves have the general shape shown in figure 4.11, a good policy will be one that equalizes the effects of $\varepsilon_c$ and $\varepsilon_i$.†

Table 4.2 may be used to choose nominal signal levels in this way. It shows for a number of choices of $\frac{V_c}{\sigma}$ both the probability of clipping and the equivalent "clipping noise" level, assuming $V_c$=lOV. This equivalent noise voltage is compared in some cases to typical quantization and thermal noise contributions $\varepsilon_i$ for various integrator noise mechanisms. Thus, for instance, setting the rms level of $x_i$ to 1.25 volts when the clipping level is 10 volts results in a

† Note that, properly speaking, one should not use integrator noise referred to the input of each integrator, which is how we have described $\varepsilon_i$ up to now, but to the output, because that is where $\varepsilon_c$ is applied in figure 4.9.

| rms voltage | $\frac{V_c}{\sigma}$ | *P (clipping* ) | clipping noise | equivalents |
|---|---|---|---|---|
| at output | | (eq. 4-12) | voltage (rms) | see text |
| 10 | 1 | .37731 | 3.88V | |
| 5 | 2 | $45.5 \cdot 10^{-3}$ | 537mV | |
| 3.i | 3 | $2.7 \cdot 10^{-3}$ | 67mV | ~5bits |
| 2.5 | 4 | $63.3 \cdot 10^{-6}$ | 6.2mV | ≅9*bits* |
| 2 | 5 | $573 \cdot 10^{-9}$ | 393$\mu$V | ≅ 13bits |
| 1.6 | 6 | $1.9\, 10^{-9}$ | 16.4$\mu$V | |
| 1.43 | 7 | $2.56 \cdot 10^{-12}$ | 440nV | ≅1$k\Omega$,10$kHz$ |
| ***1.25*** | ***8*** | ***1.2442.*** $10^{-15}$ | 7.5nV | ≅1$k\Omega$,4$Hz$ |

**Table 4.2: Choice of Scaling Level**

clipping noise level approximately equivalent to that produced by Johnson noise in a 1$k\Omega$ resistor over a bandwidth of ***4Hz.*** The table suggests that choosing $\frac{V_c}{\sigma}$ in the range 3-5 makes the contributions of clipping and quantization noise similar for a digital filter with inputs to delays (i.e. outputs of multiplier/accumulators) quantized to 5-13 bits, and that $\frac{V_c}{\sigma}$≅7 looks like **a** sensible choice for an analogue filter, because it makes "clipping noise" similar in level to the kinds of thermal noise likely to be found in an active filter (depending on its bandwidth and the components used). These figures correspond to rules requiring that the rms level in digital filters be about 2 bits below overload and that analogue filters have 15-20dB of "headroom" (a term used in audio engineering for the ratio between clipping level and rms signal level).

## 4.8 Optimum Dynamic Range

Scaling optimizes the dynamic range of a given structure, but some structures are inherently better for dynamic range than others. This section adapts to analog systems the results of [5] which show how to construct two types of state-space digital filter with "optimum" dynamic range in two different senses. One type produces a structure with the lowest possible $\sum_i \|\mathbf{g}_i\|_2^2$ over all $L_2$ scaled filters, and so (if the optional frequency-weighting function in the norm $\|\mathbf{g}_i\|_2$ is chosen properly for the problem and if $L_2$ measures make sense) produces structures with the highest dynamic range possible. The other type optimizes the rather less useful aggregate measure $\prod_i \|\mathbf{g}_i\|_2^2$.

The results follow from analysis of the behaviour of two matrices, K and $\mathbf{W}$, which give correlation among $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ respectively. They are defined by

$$\mathbf{K}_{ij} \triangleq \mathbf{f}_i \cdot \mathbf{f}_j \qquad (4\text{-}13)$$

$$\mathbf{W}_{ij} \triangleq \mathbf{g}_i \cdot \mathbf{g}_j \qquad (4\text{-}14)$$

Where the symbol . denotes an "inner product". The work in [5] referred specifically to a particular inner product, defined for discrete-time filters in terms of impulse response sequences

$$\mathbf{f}_i \cdot \mathbf{f}_j \triangleq \sum_{k=0}^{\infty} \mathbf{f}_{i,t}(kT)\mathbf{f}_{j,t}(kT) \qquad (4\text{-}15)$$

where $T$ is the sampling interval. In fact any inner product may be used in definitions (4-13) and (4-14) without affecting their results or methods, so we can apply their work directly to analog filters as soon as we decide on suitable inner products.

The reason that inner products are relevant is that they are related directly to norms: from an inner product one may produce a norm by

$$\|\mathbf{f}_i\| \triangleq \sqrt{\mathbf{f}_i \bullet \mathbf{f}_i} \qquad (4\text{-}16)$$

so that the diagonal elements of K and **W** are squared norms for $\mathbf{f}_i$ and $\mathbf{g}_i$. We have already seen that under some reasonable assumptions these norms may be related directly to the limiting and noise generation problems of dynamic range. We therefore wish to choose the definition of $\bullet$ in such a way as to automatically produce the kind of norm we need on $\{\mathbf{f}_i\}$ or $\{\mathbf{g}_i\}$: in fact the work of [5] applies even if we choose different definitions of inner product for $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$.

Whenever we have a norm of the general $L_2$ type we may obtain an inner product that degenerates to that norm when both vectors in the product are the same. In particular, an inner product may be defined as a weighted integral

$$\mathbf{f}_i \bullet \mathbf{f}_j \triangleq \int_{-\infty}^{\infty} \bar{\mathbf{f}}_{i,s}(j\omega) W(\omega) \mathbf{f}_{j,s}(j\omega)\, d\omega \qquad (4\text{-}17)$$

where $W(\omega)$ is any (positive, real) weighting function. This degenerates directly to an $L_2$ norm when $\mathbf{f}_i = \mathbf{f}_j$ and satisfies the requirements for a functional to be an inner product [14]. This means that we can use the results of [5] to minimize output noise power for a scaled filter whenever $L_2$ norms are appropriate for both $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$.

### 4.8.1 Linear Transformations

We have explained the relevance to our work of the diagonal entries $\mathbf{K}_{ii}$ and $\mathbf{W}_{ii}$: the reason that the off-diagonal elements matter is that they enable one to predict what will happen to the diagonals if different $\{\mathbf{f}_i\}$ are chosen.

Because a canonic system is determined by its $f$ vector, and because the $\{\mathbf{f}_i\}$ must be independent, the relationship between two systems may be studied in terms of the relationships between their $\{\mathbf{f}_i\}$ vectors. A linear transformation

$$\tilde{f}=Tf$$

may be used to transform one system into any other.

Mullis and Roberts [5] showed how to derive a transformation T that would optimize dynamic range as measured by K and W. Under transformation by T, K and W change as follows:

$$\hat{K}=TKT^{T} \tag{4-18}$$

$$\hat{W}=T^{-T}WT^{-1} \tag{4-19}$$

It follows from the above that $(\tilde{K}\tilde{W})=T(KW)T^{-1}$, which is a similarity transform. One may deduce from this that the eigenvalues of KW are invariant under transformation, i.e. are the same for every canonic filter structure realizing a given transfer function. We will use the notation of [5] and denote these eigenvalues $\{\mu_i^2\}$.* Mullis and Roberts referred to the $\{\mu_i\}$ (which we will take to be the positive square roots for convenience) as "second-order modes", and they are called "singular values" elsewhere (e.g. [15].

The operation of "scaling" a filter is, in these terms, just a very simple choice of T: a diagonal matrix. This makes it easy to discount the effects of scaling and search for structures with inherently good dynamic range.

**Conditions for Optimality**

Mullis and Roberts [5] investigated the dynamic range of structures by looking at products $K_{ii}W_{ii}=||f_i||^2||g_i||^2$. This product is not affected by scaling, because $f_i$ and $g_i$ change by opposite amounts when scaling is done, but it does change from structure to structure. If one scales a structure with a given $\sum_i K_{ii}W_{ii}$ in order to get $\tilde{K}_{ii}\triangleq||\tilde{f}_i||^2=1\ \forall i$, then the output noise of the $L_2$-scaled system is $\sum_i \tilde{W}_{ii}=\sum_i \tilde{K}_{ii}\tilde{W}_{ii}=\sum_i K_{ii}W_{ii}$. The figure $\sum_i K_{ii}W_{ii}$ therefore shows the ***inherent*** dynamic range of a structure.

• These eigenvalues are always positive. K and W are positive definite, but it is not automatic that $(KW)$ also is. In this case, though it it possible to construct a T such that K=I, in which case $(\tilde{K}\tilde{W})=\tilde{W}$: but any $\tilde{W}$ is positive definite, so $\tilde{K}\tilde{W}$ is too. If one $(\tilde{K}\tilde{W})$ has positive eigenvalues then they all do because the eigenvalues are invariant.

Mullis and Roberts [5] gave a procedure (which we show in more detai1 in section 5.2.2) to minimize the rather odd aggregate $\prod \mathbf{K}_{ii}\mathbf{W}_{ii}$, and another to modify the resulting system to get a minimum $\sum_i \mathbf{K}_{ii}\mathbf{W}_{ii}$. These procedures based their operation on the invariance of "second-order modes" $\mu_i^2$.

Both techniques may be replicated for analog filters just by choosing inner products for $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ that suit the analog problem.

The structure minimizing $\prod_i \mathbf{K}_{ii}\mathbf{W}_{ii}$ has a particularly interesting property: K and **W** are simultaneously diagonal. In fact, by scaling suitably, this structure may be made to satisfy the equations

$$\mathbf{K}=\mathbf{W}=diag\{\mu_1,\mu_2,\cdots,\mu_n\} \tag{4-20}$$

This structure was called a "principa1 axis realization" (or, in[I5], a "balanced" realization) because one could interpret the simultaneous diagonalization to mean that the filter's state variables are aligned with the principal axes (in state space) of an ellipsoid describing the most probable system states, while its $\{\mathbf{g}_i\}$ are aligned with the axes of another ellipsoid that may be used to describe the effects of each state on total output noise.

The structure minimizing $\sum_i \mathbf{K}_{ii}\mathbf{W}_{ii}$ also has interesting properties. By suitable scaling it may be made to satisfy two conditions,

$$\mathbf{K}=\mathbf{W} \tag{4-21a}$$

and

$$\mathbf{K}_{ii}=\mathbf{K}_{jj} \quad \forall\, i,j \tag{4-21b}$$

The condition **K=W** which applies to both of these types of optimum implies a sort of self-duality, because it says that the interrelationships among $\{\mathbf{f}_i\}$ are the same as those among $\{\mathbf{g}_i\}$. This kind of symmetry appears again in some of the minimum-sensitivity structures of chapter 7, and bears a thought-provoking resemblance to a reciprocity condition.

When an $L_2$ measure of dynamic range is applicable, the work of [5] provides the best possible structure for filter dynamic range. It is, however, in general fully dense (i.e. has nonzero values for every {A,b,c,d} coefficient) and one might therefore wish .to find a relatively sparse sub-optimum system as long as the dynamic range performance sacrificed by so doing is not excessive.

## 4.9 Orthogonality and Angles

The concept of an inner product introduced in section 4.8 may be used to define "angles" among $\{\mathbf{f}_i\}$ vectors. As a particularly interesting case, two (nonzero) vectors $\mathbf{f}_i$ and $\mathbf{f}_j$ are said to be orthogonal when $\mathbf{f}_i \bullet \mathbf{f}_j = 0$. In general, the angle between two vectors is given by $\vartheta$ where

$$\vartheta=\cos^{-1}\left[\frac{\mathbf{f}_i \bullet \mathbf{f}_j}{\|\mathbf{f}_i\|\|\mathbf{f}_j\|}\right]$$

This definition of angle has the natural geometric properties for angles: in particular if the angle between two vectors is 0, then they are aligned (i.e. one is just a scalar multiple of the other).

The concept of angle permits us to quantify the concept of "near-dependence" introduced in chapter 2: we can say that a system "almost" has a linear dependency among its $\{\mathbf{f}_i\}$ if the angle between two $\{\mathbf{f}_i\}$ is very small (or, better, if the angle between some $\mathbf{f}_i$ and the subspace spanned by the others is very small).

As an example if one investigates companion form filters, for which $\{\mathbf{f}_i\}=\{\frac{s^{i-1}}{e_s}\}$, one finds that the angles between vectors $\mathbf{f}_i$ and $\mathbf{f}_{i+2}$ are small (often less than 1°). This pinpoints the problem with companion form, as well as suggesting why it is not too bad at orders 1 and 2 (where all $\{\mathbf{f}_i\}$ are orthogonal).

## 4.10 Distortion Propagation

When the internal devices of a filter are known to generate distortion products of signals, one is interested in knowing how these propagate to the system output. The $\{\mathbf{g}_i\}$ functions provide the answer: signals appearing at integrator inputs because of distortion "see" gains $\{\mathbf{g}_i\}$ to the system output. Thus filters that tend to keep $\{\mathbf{g}_i\}$ as small as possible propagate distortion-induced signals as little as possible. The situation here is completely analogous to that for noise, so it suffices to design filters for low noise to obtain a minimum of distortion at the output.

## 4.11 Element Spread

"Element spread", the ratio between the Iargest and smallest vaIues of a given type of element in a filter circuit, is often mentioned as a figure of merit. In some technologies it is difficult or expensive to obtain large ratios, while in other cases the element spread is held to reflect sensitivity performance. This issue is confused by the fact that element spreads are affected both by scaling and by the structure chosen.

When there is a direct correspondence between **A** matrix entries and element values it suffices to investigate coefficient spread in $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ to investigate element spread.

Some experimental investigation of the behaviour of coefficient spread in filter designs by the author has suggested the following:

1 That good structures of frequency-normalized filters tend not to have *A* coefficients greater than 1 in magnitude after $L_2$ scaling, but may have some quite small elements. This observation may be interpreted as implying that

good filters do not obtain internal inputs $\{s\mathbf{f}_i\}$ by the ill-conditioned technique of subtracting nearly equal $\{\mathbf{f}_i\}$. When a lowpass or bandpass filter is normalized to its upper passband edge, the bulk of the power in $\{\mathbf{f}_i\}$ will be at frequencies less than 1 rad/sec (i.e. in the passband): it follows that the norms of the inputs $\{s\mathbf{f}_i\}$ will generally be less than 1 because $\|\mathbf{f}_i\|$ is 1 (we said the filter was scaled) and $s<1$ for the band in which $|s\mathbf{f}_i|$ is large. The A matrix is therefore usually trying to add n vectors $\{\mathbf{f}_i\}$ of length 1 in order to obtain each $\{s\mathbf{f}_i\}$, and will only need to use coefficients larger than 1 to do so if it is required to subtract correlated $\{\mathbf{f}_i\}$.

2. Because, in good systems, products of the form $\mathbf{g}_i\mathbf{f}_j$ are all fairly similar in magnitude, the simple derivative sensitivity $\frac{at}{\partial \mathbf{A}_{ij}}$ is approximately the same both for large and small $\mathbf{A}_{ij}$ (cf. equation (4-10)). Thus classical (fractional) sensitivity is proportional to $|\mathbf{A}_{ij}|$ and small entries need not be simulated as accurately as large ones.

3. That inherently poor structures, like companion form, can have large A entries even when scaled (but need not). They can also have wildly varying $\mathbf{g}_i\mathbf{f}_j$ products.

## 4.12 Tuning and Debugging

High-performance filter designs tend to have strong coupling among their components, and are therefore fundamentally hard to tune [16]. First-order sensitivity information, together with measured data for transfer-function error, may be used to alleviate this difficulty [17]. It is also hard to locate a faulty component in a good filter circuit [18], and a coarse version of a tuning procedure would make it easier.

This section simply relates our existing formulae for first-order sensitivity of filters to the well-known [14] problem of least-squares optimization.

We can use first-order sensitivity to a selected set of circuit elements $\alpha_i$ to model a transfer function error as

$$\Delta t_s \cong \sum_i \frac{\partial t_s}{\partial \alpha_i} \Delta\alpha_i \tag{4-22}$$

If we wish to correct a measured error $\Delta t_s$ towards zero, equation (4-20) tells us to make changes $(-\Delta\alpha_i)$. Since we can readily find $\frac{dt_s}{d\alpha_i}$ from the formulae in section 4.4, the first-order tuning problem just reduces to solving a set of linear equations (4-20) for $\Delta\alpha_i$.

Each of the {A,b,c,d} coefficients involved in a system (potentially $o(n^2)$ of them) can be in error, and there are only *2N+l* degrees of freedom for error in $t_s$ (one for each coefficient in p and e). Because there are usually more coefficients that may be adjusted than there are degrees of freedom for error, there are generally several circuit changes that would solve a problem. When the correspondence between circuit elements and {A,b,c,d] coefficients is not one-to-one, there will usually be an even larger surplus of ways to tune out an error. Thus, e.g., if a MilIer integrator's gain $\frac{1}{CR}$ is too low one could correct the problem by reducing either C or R. This surplus of choice means that in general there will be infinitely many solutions to (4-20). The problem of designing a tuning strategy therefore includes that of choosing a good set of $\alpha_i$, a problem that depends both on technological considerations (e.g. which types of components are most easily trimmed) and on how well any given selection allows the tuning system to remove important or common types of error,

One can think of the error function $\Delta t_s$ and first-order sensitivities $\frac{\partial t_s}{\partial \alpha_i}$ as vectors in a 2N+l-dimensional space of possible errors. The problem of locating a fault may be addressed by looking for the sensitivity vector most closely aligned with $\Delta t_s$; the problem of choosing a good set of coefficients to use in trimming a filter is the problem of choosing a set of sensitivity vectors that span the error space (or a subspace of "plausible" errors): the problem of trimming as few components as possible while improving $t_s$ a given amount is that of finding a subspace spanned by as few sensitivity vectors as possible that is close enough to $\Delta t_s$.

If a least-mean-square error criterion is used to measure $\|\Delta t\|$ tuning problems may be solved in a straightforward way with a few matrix operations. The problem is the statistical one of making a least-squares estimate of $\Delta t_s$ in terms of given sensitivity functions, and is solved by an equation in the Gram matrix [14]. calling the derivative sensitivities $\frac{\partial t_s}{\partial \alpha_i}$ of the components to be trimmed $\mathbf{s}_i$, the Gram matrix is defined as

$$G(\mathbf{s}_1, \mathbf{s}_2, \cdots \mathbf{s}_N) \triangleq$$

$$\begin{bmatrix} \mathbf{s}_1 \bullet \mathbf{s}_1 & \mathbf{s}_1 \bullet \mathbf{s}_2 & \mathbf{s}_1 \bullet \mathbf{s}_N \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \mathbf{s}_N \bullet \mathbf{s}_1 & \cdot & \mathbf{s}_N \bullet \mathbf{s}_N \end{bmatrix}$$

and the problem of best approximating an error vector **At** by a weighted sum of the $\mathbf{s}_i$ is solved by choosing a vector of weights $\hat{\beta}$,

$$\hat{\beta} = G^{-1} \begin{bmatrix} \mathbf{s}_1 \bullet \Delta t \\ \mathbf{s}_2 \bullet \Delta t \\ \cdot \\ \mathbf{s}_N \bullet \Delta t \end{bmatrix} \tag{4-23}$$

When test data are available for a set of frequencies $\omega_j$, one may use the inner product

$$\mathbf{s}_k \bullet \mathbf{s}_l \triangleq \sum_{\omega_j} \overline{\mathbf{s}_k(j\omega_j)} \mathbf{s}_l(j\omega_j) \tag{4-24}$$

It would also be possible to weight measurements at different $\omega_j$ differently in a weighted inner product

$$\mathbf{s}_k \bullet \mathbf{s}_l \triangleq \sum_{\omega_j} \alpha_j \overline{\mathbf{s}_k(j\omega_j)} \mathbf{s}_l(j\omega_j). \qquad (4\text{-}25)$$

These formulae need not involve a great deal of computing for each filter built, as most of the work can be done in advance for the design. One may precompute $\mathbf{G}^{-1}$ for a design and also write the vector of inner products in (4-23) as a multiplication of a vector At of measured errors by a fixed matrix. Thus one need only do mn multiplications to tune each individual filter towards the nominal design.

Because the first-order functional tuning problem is concerned with sensitivity functions that are readily derived from $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$, filter design in terms of these sets of functions makes it possible to look at the implications for tunability of a particular design strategy. as a concrete example, we show in chapter 7 that good filter designs tend to have nearly equal sensitivities to all of their integrators: it follows that it is generally a poor idea to include more than one integrator gain $\gamma_i$ in a set of parameters to use for tuning.

## 4.13 Summary and Conclusions

We have shown how important intermediate functions are in studying filter performance: they combine in simple ways to provide expressions allowing one to study the dominant problems of filter design.

These simple expressions make the task of designing in terms of a set of $\{\mathbf{f}_i\}$ easier than that of more conventional topology-based design.

## 4.14 References

[1] A. Fettweis, *"A General Theorem for Signal- Flow Networks, with Applicatioms,"* Arch. Elektronik Ubertragungstechnik, vol. 25, 1971, pp. 557-561

[2] R. Seviora and' M. Sablatash *A Tellegen's Theorem for Digital Filters* IEEE Trans. Circuit Theory, CT-18 Jan 1971 pp. 201-203

[3] M.S. Ghausi and K.R. Laker, Modern ***Filter Design, Active RC and Switched Capacitor,*** Prentice-Hall, 198 1

[4] Editor F. Jay ***IEEE Standard Dictionary of Electrical and Electronics Terms Second Edition*** IEEE, 1977

[5] C.T. Mullis and R.A. Roberts, ***Synthesis of Minimum Roundoff Noise Fixed- point DigitaL Filters*** IEEE Trans. Circuits and Systems, vol. CAS-23, pp. 551-562, Sept. 1976.

[6] LB. Jackson, "On ***the Interaction of Roundoff Noise and Dynamic Range*** in ***Digital Filters***" Bell Syst. Tech. J., v, 49, No. 2, pp. 159-184, 1970

[7] S.Y. Hwang, "On ***Optimization of Cascade Fixed- point Digital Filters",*** IEEE Trans. Circuits and Systems, CAS-21, pp.l63-166, Jan. 1974

[8] L. Collatz ***Functional Analysis and Numerical Mathematics*** Academic Press, 1966

[9] J.E. Solomon, ***"The Monolithic Op Amp: A Tutorial Study",*** IEEE J. Solid State Circuits, SC-9 pp. 314-332, Dec. 1974

[10] AS. Sedra and P.0. B.rackett, Filter ***Theory*** and ***Design: Active and Passive*** Matrix Publishers, Champagne, Illinois, 1978

[11] W.B. Davenport Jr., Probability and ***Random Processes,*** McGraw-Hi& 1970

[12] C.D. Motchenbacher and F.C. Fitchen ***Low- Noise Electronic Design*** Wiley 1973

[13] E. Lueder ***"Optimization of the Dynamic Range and the Noise Distance of*** RC- ***Active Filters by Dynamic Programming",*** Int. J. of Circuit Theory Appl., vol. 3, pp. 365-370, Dec. 1975

[14] D.G. Luenberger, ***Optimization by Vector Space Methods,*** Wiley, 1969

[15] B.C. Moore, Singular ***Value Analysis of Linear Systems,*** Systems Control Report No. 7801, Systems Control Group, Univ. Toronto, Toronto, Canada, July 1976

[16] P.V. Lopresti ***Optimum Design of Linear Tuning Algorithms*** IEEE Trans. Circuits and Systems, CAS-24, Mar. 1977 pp. 144-151

[17] J.F. Pinel, ***Computer- Aided Network Tuning*** IEEE Trans. Circuit Theory, CT-18, Jan. 1971, pp. 192-194

[18] R. Saeks, ***"Criteria for Analog Fault Diagnosis",*** Proc. European Conf. Circuit Theory and Design, The Hague, Aug. 1981

# 5. Redundancy and Independence

We show that the problem of choosing a good set of system states is subject to two mildly conflicting requirements: that the states must be "independent enough" to keep internal gains low, but that there should usually be some similarity – or "redundancy" – to get the best possible performance.

## 5.1 Introduction

We know that the $\{\mathbf{f}_i\}$ functions of a canonic system have to be independent, and can demonstrate that various practical problems afflict filters more seriously as their $\{\mathbf{f}_i\}$ functions come closer to having linear dependencies. Some configurations which are known to be much too sensitive to component errors (e.g. companion form) can also be shown to behave badly because they have pairs of $\mathbf{f}$ vectors almost collinear.

It might therefore seem that the way to design good filters would be to choose $\{\mathbf{f}_i\}$ orthogonal, but this chapter shows why this is not the case. A limited amount of correlation between intermediate transfer functions can serve to improve performance by redundantly representing important vectors. This chapter discusses the struggle between independence and redundancy.

We will show two extreme cases: a type of transfer function for which an orthogonal realization is best; and another for which a realization redundant to the point of degeneracy is optimum. Transfer functions encountered in practice will fall between these two extremes. We will also discuss the significance of redundancy in practical situations and show some of the ways in which it can be advantageously used. An understanding of the effects of redundancy is also helpful in analyzing the filter literature, because there are situations where "hidden" redundancies serve to make a topology look better than it really is.

Study of the use of redundancy also has bearing on investigation of the use of non-canonic realizations. One reason to expect that these might be interesting is that ladder realizations are not in general canonic.

Of the two extreme cases, the one illustrating the use of redundancy is the more fertile. Let us leave it till last, and now look at a case in which orthonormal $\{\mathbf{f}_i\}$ leads t.o a structure that is in some significant sense optimum.

## 5.2 When orthogonal realizations are good

This section shows that a second-order bandpass transfer function can be realized with orthogonal $\{\mathbf{f}_i\}$ in such a way that it has the best possible dynamic range (in the $L_2$ sense) when equal amounts of noise are injected at the inputs of the two integrators. This is only one of a class of transfer functions which seem to be realizable this way: arbitrary-order all-pass functions also appear to be of this type. This case is particularly interesting for two reasons: one is that there is a relationship with LC ladders, and the other is that in this case there turn out to be many realizations with optimum dynamic range, only one of which also has optimum integrator sensitivity. This last point suggests that optimum integrator sensitivity is related to optimum dynamic range, but is a somewhat stronger constraint.

This section also looks at the "principal axis realization" of [l], which is a special type of orthogonal filter. It is optimum when some states may be represented more accurately than others and the "cost" of a state is logarithmic in accuracy. We will later treat. redundancy as a different way of attaining the same objective: concentration of effort on important states.

The measure we use for inner product (and therefore our definition of orthogonality, cf. section 4.9) – a correlation as described in chapter 4 – has an important physical meaning. If two $\mathbf{f}_i$ functions are orthogonal and the input signal has a uniform (white) power spectrum, then there will be zero correlation between the two corresponding output signals, which implies that no information about one signal is present in the other. One might expect this to be good, since individual integrators are then recording completely different things about the history of the input signal.

“Orthonormal“ systems are those which are both orthogonal (i.e. have all $\{\mathbf{f}_i\}$ mutually orthogonal) and scaled (i.e. $\|\mathbf{f}_i\|_2=1$).

### 5.2.1 Second-OrderOrthonormalBandpass

The functions $\{\mathbf{f}_i\}=\{\omega_o/e_s(s),s/e_s(s)\}$ are orthogonal (for an integral measure like that of equation 4.17 one may show that functions with even numerators are uncorrelated with those with odd numerators). The system that they generate for the bandpass $t_s(s)\triangleq\frac{s}{s^2+\frac{\omega_o}{8}s+\omega_o^2}$ is

$$A=\begin{bmatrix}0 & \omega_o\\ -\omega_o & -\frac{\omega_o}{Q}\end{bmatrix} \quad b=\begin{bmatrix}0\\1\end{bmatrix}$$

$$c^T=[0 \quad 1] \quad d=0$$

and

$$g^T\triangleq c^T(sI-A)^{-1}=\{-\frac{\omega_o}{e_s(s)},\frac{s}{e_s(s)}\}.$$

Now we see that the $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ sets are identical module sign: this, together with the fact that $\mathbf{f}_1\cdot\mathbf{f}_2=0$, gives K=W, which is the first condition of section 4.8 for optimum dynamic range. Furthermore, computing norms (either by integration or by the method to be presented in chapter 6) reveals that $\mathbf{K}_{11}=\mathbf{K}_{22}$, which is the second part of the condition.

Thus this system has optimum dynamic range, and also has orthogonal states. This system w-ill be encountered again later, in section 7.3, when we investigate biquadratic structures. It turns out that this $\{\mathbf{f}_i\}$ set models inductor current and capacitor voltage in a bandpass LC ladder. As a peculiarity of the second-order bandpass case, the singly and doubly-terminated ladders are equivalent, as will be discussed in section 7.3.

As a matter of interest, this is just a special case of a phenomenon that applies for arbitrary order. Choosing a set $\{\mathbf{f}_i\}$ by using Gram-Schmidt orthogonalization on the $\{\mathbf{f}_i\}$ for companion form, $\{1/e_s(s), s/e_s(s), \cdots s^{N-1}/e_s(s)\}$, turns out to give an orthonormal system with a symmetry that produces $\mathbf{K}=\mathbf{W}=\mathbf{I}$ when the output is taken to be $t_s(s)=\mathbf{f}_n(s)$. In this special case, which unfortunately doesn't appear to generate anything very useful except allpass $t_s(s)$, we again have orthogonal $\{\mathbf{f}_i\}$ and optimum dynamic range (cf. section 4.8, equation 4.21}. Since the result is of only passing interest and a proof would be fairly involved, none is attempted. These syntheses appear to have the structure of singly-terminated all-pole LC ladders. A medium-order system with this feedback structure is presented in chapter 8.

### 5.2.2 The Principal-Axis Realization

Mullis and Roberts [1] proposed a type of system that would have minimum round-off noise contributed by quantization at delay inputs for a given average word-length: a biquad might have one 8-bit and one 12-bit delay [†], for an average of 10 bits. While this type of system is not particularly practical (cost is not often dictated by average word-length, and the systems are generally fully dense and thus need $(n+1)^2$ multiplies), it will help show why orthogonal systems are ***not*** generally the best ones. We will deal with analogies in continuous-time systems later.

[†] An “8-bit delay” stores 8 bits of signal data from one cycle to the next. As discussed in section 3.3.1, it is natural to worry about resolution at delays, not because memory is expensive but because a multiplier-accumulator of the same resolution precedes the delay's input.

The optimum of [1] is obtained when K and W are simultaneously diagonal. The ‘originators call this a “principal axis realization” because it has the property that its states are aligned with the principal axes of an ellipsoid in state-space that represents the volume in which the state is most likely to be found.

One can simultaneously diagonalize K and W by “rotating” appropriately any system in which K=I. A system with K=I, i.e. with orthonormal $\{\mathbf{f}_i\}$, may be obtained from any starting set by using the Gram-Schmidt procedure. The next step is to find the eigenvalue-eigenvector decomposition of W, where

$$\mathbf{W}=\mathbf{U}\mathbf{D}\mathbf{U}^T,$$

D is diagonal, and U is unitary (i.e. $\mathbf{U}^{-1}=\mathbf{U}^T$). Transforming states according to

$$\tilde{\mathbf{f}}=\mathbf{U}^{-1}\mathbf{f}$$

results in

$$\tilde{\mathbf{K}}=\mathbf{U}^{-1}\mathbf{K}\mathbf{U}^{-T}=\mathbf{U}^T\mathbf{I}\mathbf{U}=\mathbf{I}$$

$$\tilde{\mathbf{W}}=\mathbf{U}^T\mathbf{W}\mathbf{U}=\mathbf{U}^T\mathbf{U}\mathbf{D}\mathbf{U}^T\mathbf{U}=\mathbf{D}$$

where the new K and **W** are simultaneousy diagonal, as required.

Now, in order for this system to have minimum noise, one must represent each state with a number of bits dependent on the corresponding element of **W=D.** The overall effect is that some states need to be represented more accurately than others because they are fundamentally more important. We show in the next section how this difference in importance is handled in the more practical case, when an “equal amount of effort” (i.e. one integrator) must be devoted to each state.

For comparison, the second-order special case discussed in section 5.2.1 above is one in which this algorithm would anyway assign an equal number of bits to each state: the “ellipsoids” of [1] degenerate into spheres because of the choice of transfer function.

Analogies with the continuous-time case are possible: the manipulations of X and W required to simultaneously diagonalize them are identical for the discrete-time and continuous-time case, and one could use relatively expensive (low-noise) integrators for the states to which the algorithm above assigns many bits of precision. Because the principal-axis realization minimizes $\prod \mathbf{K}_{ii}\mathbf{W}_{ii}$ it minimizes $\sum_i \ln(\mathbf{W}_{ii})$ over all scaled realizations $(\mathbf{K}_{ii}=1)$. This means that it is an optimum if the “cost” of reducing noise from an integrator depends logarithmically on the performance required. Even if the relationship between noise power and cost in an integrator is not logarithmic, it is possible that the principal-axis realization would provide good performance for an analog filter on which tolerance assignment is to be performed.

This realization was also found by Moore [4], who dealt with three versions of it that differ only in scaling. The version with K=I he called “input normal”.

## 5.3 Why Highly Correlated $\{\mathbf{f}_i\}$ are Generally Bad

The “companion-form” for system realization is known to be very poor for high-order filters: it has high sensitivities and low dynamic range. Investigation reveals that, for practical high-order transfer functions, some of its $\{\mathbf{f}_i\}$ can be very highly correlated: in fact it is not unusual to find pairs of vectors less than 1° apart (cf. section 4.9). An example of a companion-form design of an $8^{th}$ order filter that appears in chapter 7 will illustrate this point.

In section 0.5 we will develop a formula that relates $\{\mathbf{g}_i\}$ to $\{\mathbf{f}_i\}$, which will show that $\{\mathbf{g}_i\}$ are related to the ***inverse*** of F, a matrix representation of the $\{\mathbf{f}_i\}$. As $\{\mathbf{f}_i\}$ become more highly correlated, the inverse of this matrix will contain larger entries, and so $\{\mathbf{g}_i\}$ will grow.

Since the sensitivity and dynamic range measures given in chapter 4 are all worsened by increasing $\{\mathbf{g}_i\}$, we should expect them to be bad. The formulae in chapter 6 also show a relationship between system matrix coefficients $(\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$

elements, or internal gains) and this $\mathbf{F}^{-1}$ matrix: thus systems with strong dependencies among their $\{\mathbf{f}_i\}$ have large internal gains, from which come their poor performance.

## 5.4 When Redundancy is Important

This section shows an extreme example of a transfer function best realized by a system whose $\{\mathbf{f}_i\}$ are highly correlated. We can either think of this as an example of the use of non-canonic structures or as a study of the effect of pole-zero cancellation. In the latter way of discussing the issue, the function used in the example is second-order but has a pole-zero cancellation so that it is essentially first-order. Thus one of the natural modes is much more "important" than the other: as long as the cancelling zero follows it, it can move anywhere at all without disturbing $t_s(s)$ at all. For this reason the best realization has both "states" concentrating on the non-cancelled pole – it completely ignores the cancelled one.

While this example is certainly degenerate, it is not unimportant. In general transfer functions will be more sensitive to some poles than to others, and we would expect a similar "concentration" on important poles to be useful. The insight gained by studying the straightforward degenerate case may be applied to more practical ones.

Non-canonic structures are often encountered in the filter literature: the concept of redundant representation of states is useful in understanding their significance.

This discussion will also show up **a** strong relationship between high-performance filter topologies and the tolerance assignment technique [2] of using high-quality components in critical places (which in its simplest form is a common circuit design "trick"). We have already discussed the principal-axis realization as a case of optimality after a particular type of tolerance assignment.

Let the transfer function to be realized be

$$t_s(s) \triangleq \frac{-(s+2)}{(s+1)(s+2)} = \frac{-1}{s+1} \tag{5-1}$$

### 54.1 First-order Function Implemented with Two Integrators

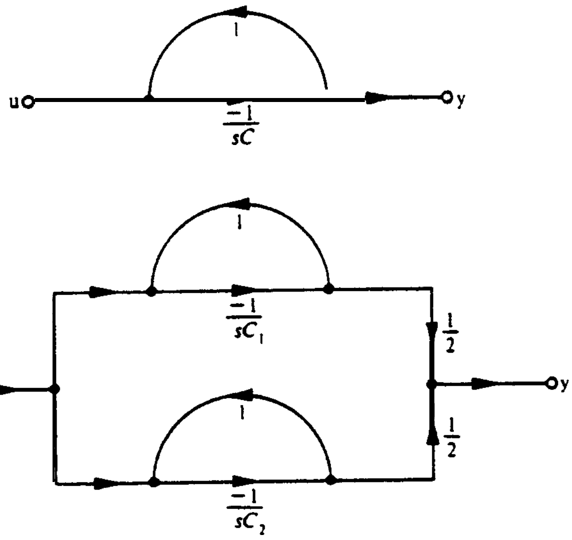


Figure 5.1: Two Realizations of $t_s(s) = \frac{-1}{s+1}$

The SFGs of figure 5. 1a and 5.1b both implement the transfer function $t_s(s) = \frac{-1}{s+1}$, but the second one has only half as much output noise as the first (if we count only integrator noise) because the state $x_1 = y$ is redundantly represented: i.e. appears at two outputs. This appears paradoxical, because the second circuit has twice as many integrators and so is subjected to twice as much noise power as the first. Each noise source, however, "sees" a lower voltage gain to the output: $\tilde{g}_1 = \tilde{g}_2 = g_1/2$: This represents a factor 4 less ***power*** gain for the redundant circuit than for the canonic, so that (if the noise sources are uncorrelated) the overall noise power at the output of the redundant circuit is lower by a factor of two than that for the canonic circuit.

This effect is familiar from elementary statistics: by **averaging several "Samples"** of the signal $u(s)/s+1$, each corrupted by noise, we can obtain a more accurate estimate than by taking only a single sample.

The same principle applies to sensitivity, although again only when component errors may be assumed to be uncorrelated. For the canonic circuit,

$$S_{C}^{t_s(s)} = \frac{-s}{t_s(s)} \mathbf{f}_1 \mathbf{g}_1$$

while for that of Figure 5.lb,

$$S_{C_1}^{t_s(s)} = S_{C_2}^{t_s(s)} = \frac{-s}{t_s(s)} \mathbf{f}_1 \frac{\mathbf{g}_1}{2}$$

and a sum-of-squares sensitivity measure for the redundant circuit is better by a factor of 2 than that for the canonic one.

It is tempting to argue that this way of reducing sensitivity and noise is somehow “cheating”. A more extreme example is offered by Figure 5.2, in which the passive elements for the canonic realization are replaced by parallel combinations. This circuit has the same capacitor (and resistor) sensitivities as the redundant circuit, and for the exact same reason. Depending on the mechanism of noise generation in the integrator, it may also have lower output noise.

As long as capacitor errors are uncorrelated, both kinds of “redundancy” could seriously be expected to improve performance.

One can advance arguments (depending heavily on the details of the implementation technoIogy) to suggest that the two capacitors in Figure 5.2b might have correlated errors, thus avoiding having to admit that it is a better circuit than that of Fig. 5.1a; but then to be consistent the same suspicions must be applied to other “low-sensitivity” structures. It may be that these structures do something that is no better than a disguised version of the “trick” of Fig. 5.2. Note, in particular, that low-sensitivity structures are often quite complicated (as will be seen for the large design example of chapter 8) and thus have many components and noise sources over which to take averages.

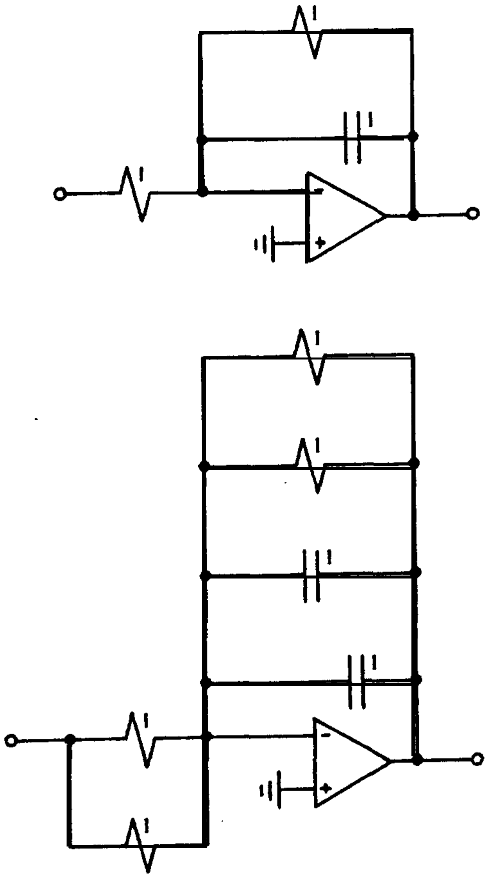


Figure 5.2: A degenerate Kind of Redundancy

**When to use more accurate components**

The “principal-axis realization” shown above had no use for redundancy because it could weight some states more heavily than others. The particular result obtained there is not particularly applicable to analog technology (or perhaps even to digital), but the nature of the optimum is interesting.

In order to use the results of [1] to produce a second-order system we will have to introduce the pole-zero cancellation mentioned at the beginning of this section. Choose a design such that:

$$\mathbf{f}_{1,s}(s)=\frac{1}{s+1}=t_s(s) \tag{5-2}$$

$$\mathbf{f}_{2,s}(s)=\frac{1}{s+2}-(\mathbf{f}_{1,s}\cdot(\frac{1}{s+2}))\mathbf{f}_{1,s} \tag{5-3}$$

which are orthogonal (because of the second term in (5-3)). This means that K is diagonal, the first requirement for a principal-axis realization. Now the corresponding system takes its output from $\mathbf{f}_1$, and $\mathbf{f}_2$ is not involved in any way in the output signal. It follows that $\mathbf{g}_2=0$, which means that only the (1,l) element of **W** is non-zero; **W** is therefore certainly diagonal. But the fact that K and **W** are simultaneously diagonal means that this is a principal-axis realization (sections 4.8 and 5.2.2).

We can now use the results of [ 1] to calculate the number of bits to assign to each state and the theoretical minimum output noise, but the result is just as degenerate as the problem. The optimum output noise figure turns out to be zero (the geometric average of the eigenvalues of KW, one of which is zero) and the optimum bit assignment is: $\infty$ bits to $\mathbf{f}_1$ and $-\infty$ to $\mathbf{f}_2$ (which means that the average number of bits assigned can be quite finite and still give zero noise).

Apart from the humour implicit in negative bit assignments, this demonstrates what we want: that the principal-axis realization has no use for non-canonic or redundant structures because it obtains good performance by concentrating its efforts on important states. There may often be other ways to do this: thus in analog circuits one can choose to minimize cost for given performance by using cheaper but noisier or less accurate components in some places than others.

## 5.5 Introducing Redundancy

We have seen extreme examples above, in one of which states were required to be completely independent and in the other of which they were required to be identical, in order to get optimum dynamic range. We can use a simple second-order lowpass example to demonstrate the value of adding controlled amounts of redundancy in implementing a transfer function: this case is intermediate between the extremes above.

If we wish to synthesize

$$t_s(s)=\frac{1}{s^2+\sqrt{2}s+1}\triangleq\frac{1}{e_s(s)}$$

we could try choosing an orthonormal pair of $\{\mathbf{f}_i\}$:

$$\mathbf{f}_1=0.67094/e_s(s)$$

$$\mathbf{f}_2=0.67094s/e_s(s)$$

where the multiplying constant 0.67094 is chosen to do scaling. Because $\{\mathbf{f}_i\}$ are orthogonal this system makes no attempt to use "redundancy". The resulting system has

$$\mathbf{A}=\begin{bmatrix}0 & 1\\ -1 & -\sqrt{2}\end{bmatrix}\mathbf{b}=\begin{bmatrix}0\\ 0.67094\end{bmatrix}$$

$$\mathbf{c}^T=[\ 1.49045\quad 0]\ d=0$$

and noise gains

$$\mathbf{g}_1=1.49045(s+\sqrt{2})/e_s(s)$$

$$\mathbf{g}_2=1.49045/e_s(s)$$

For this system

$$g_2 = -.685161(s-0.517638)/e_s(s)$$

Note the peculiar symmetry: functions $\{g_i\}$ are identical (within scaling) to corresponding $\{\mathbf{f}_i\}$. **K** and **W** for this system are

$$\hat{\mathbf{K}} = \mathrm{I} \quad \hat{\mathbf{W}} = \begin{bmatrix} 18.417 & 0 \\ 0 & 1.32227 \end{bmatrix}$$

which simply means that state $\mathbf{f}_1$ is much more important than state $\mathbf{f}_2$ in determining the system output. Thus a good realization must somehow determine the first state more carefully than the second. Redundant representation of $\mathbf{f}_1$ provides a way to do this.

Let us construct a new system with

$$\tilde{\mathbf{f}}_1 = (2\hat{\mathbf{f}}^1 + \hat{\mathbf{f}}^2)/\sqrt{5} = 0.54234(s+0.72834)$$

$$\tilde{\mathbf{f}}_2 = (2\hat{\mathbf{f}}^1 - \hat{\mathbf{f}}^2)/\sqrt{5} = 0.00937(s+71.4)$$

This choice (which is the type of rotation described in [1]) has the property of increasing the number of states associated with the important $\mathbf{f}_1$ by a factor of almost 2, since it is now the dominant signal in two outputs, which increases the power associated with it by a factor of about 2. In exchange, it reduces the signal power used for estimating $\mathbf{f}_2$ by a factor of about 2. The factors of $\sqrt{5}$ are introduced to maintain scaling.

The net effect is a system very close to the optimum (shown in [I] to be $\frac{1}{n}\left[\sum_i \mu_i\right]^2$) for the dynamic range figure $\sum_i \mathbf{K}_{ii} \mathbf{W}_{ii}$, because this *is* exactly the kind of "rotation" of states used to produce the optimum [l]. This system may also be produced by the technique described in [3] and by a technique to be described in chapter i', where it is shown to have minimum sensitivity to its integrators. The point here has been to relate these "optimum systems" to the general concept of redundancy, so that a designer may investigate (for a particular technology) whether the resulting performance improvement over a simpler design is or is not just a disguised version of the trick shown in figure 5.2.

When this kind of rotation is applied to the principal-axis realization of the degenerate transfer function investigated in section 5.4 the effect is to make $f_1=f_2=1/(s+1)$ and to ignore completely the cancelled pole at **s=-2.** Thus in the degenerate pole-zero cancellation case we see that the “optimum topology” really is just equivalent to the statistical-averaging kind of circuit.

## 5.6 Redundancy Improves Performance by At Most n

If we use $\sum_i K_{ii}W_{ii}$ as a performance measure, we can note from the extreme second-order example in section 5.4 that a factor 2 (=n) improvement was attained by redundantly representing the important state 1/(s+1). In fact it is easy to show that this is always the largest improvement possible.

Because the principal vaIues $\mu_i^2$, the eigenvalues of $KW$, are invariant we can say that $tr(KW)=\sum\mu_i^2$ is invariant. For any orthonormal filter **(K=I),** $\sum K_{ii}W_{ii}=tr(IW)=\sum W_{ii}$. Formula (32} of [1] may be manipulated to give $\sum K_{ii}W_{ii}\geq\frac{1}{n}\left[\sum\mu_i\right]^2$, and $\left[\sum\mu_i\right]^2\geq\sum\mu_i^2$, (recall that $\{\mu_i\}$ are positive, section 4.8.1) so we can write

$$\frac{\sum K_{ii}W_{ii}}{\left[\sum K_{ii}W_{ii}\right]^2}\geq\frac{1}{n}$$

Note that equality can be obtained only when $\left[\sum\mu_i\right]^2=\sum\mu_i^2$, which in turn can only happen when all but one of the $\mu_i$ are zero. This is exactly the type of degenerate case we already encountered in section 5.4.

## 5.7 Summary

We have investigated the use of two related properties of the set of $\{f_i\}$ for a filter: the extent to which states are independent of each other, and the extent to which one may average estimates of important signaIs over several states to reduce sensitivity to errors *in* those states. We have shown a relationship between the rather subtle phenomenon of coupling between states in a good filter and the relatively straightforward "trick" of tolerance assignment to match component sensitivities.

This work suggests that a good way to design filters might be to start from an orthunormal design (e.g a principal-axis design) and add redundancy wherever it is most useful while using other degrees of freedom to, for example, obtain a sparse A matrix.

## 5.8 References

[1] C.T. Mullis and R.A. Roberts, ***Synthesis of Minimum Roundoff Noise Fixed- point Digital Filters*** IEEE Trans. Circuits and Systems, vol. CAS-23, pp. 551-562, Sept. 1976.

[2] J.F. Pine1 and K.A. Roberts, ***"Tolerance Assignment in Linear Networks Using Nonlinear Programming",*** IEEE Trans. Circuit Theory, CT- 19, pp.475-479, Sept. 1972

[3] L.B. Jackson, "On ***the Interaction of Roundoff Noise and Dynamic Range in Digital Filters",*** Bell Syst. Tech. J., v. 49, No. 2, pp. 159-184, 1970

[4] B,C. Moore, Singular Value Analysis of Linear Systems, Systems Control Report No. 7801; Systems Control Group, Univ; Toronto, Toronto Canada, July 1976

# 6. Computational Methods

The synthesis example presented in chapter 2 is simple to understand, but by no means offers a good way to compute coefficients from a given set of intermediate transfer functions $\{\mathbf{f}_i\}$. This chapter presents some matrix formulae that are numerically better and that show clearly the relationships among various aspects of filter designs.

## 6.1 Introduction

The algorithms discussed here are those implemented in "dot", a FILTOR2-compatible [l] program developed to test the ideas of this thesis. They have therefore been quite thoroughly tested in designing a wide range of filters over a two-year period. No claim is made *as* to "optimality" of these computational methods, but they have certainly proven adequate.

We start by discussing vector spaces and a particular "choice of basis" we make in this chapter.

Sections 6.4 to 6.7 of this chapter show how to compute {A,b,c,d} system elements, $\{\mathbf{g}_i\}$, K and W from $\{\mathbf{f}_i\}$ and give some useful invariants.

Section 6.8 shows, by way of a summary of the purpose of these formulae, some features of the current version of "dot". It shows how these features, and thus the synthesis method of this thesis, may be used in filter design.

## 6.2 Bases in general and the residue basis

It is natural to use the concepts of vector-space theory to deal with filter synthesis because the freedom available in choosing functions for $\{\mathbf{f}_i\}$ is exactly the freedom to choose n-dimensional vectors. All of the synthesis methods discussed here either explicitly or implicitly choose a "basis" of n independent possible $f$ functions and express particular $\mathbf{f}_i$ in terms of that basis. Thus all of these methods choose a set $\{\nu_i\}$ and express any possible function as a sum $\mathbf{f}_i=\sum_1^n \alpha_i \nu_i$. **The** example given in chapter 2 dealt with numerators expressed as polynomials $\mathbf{f}_{i,s}=\sum_{i=0}^{n-1} \alpha_i s^i$, which simply takes $\nu_{i,s}=\frac{s^i}{e_s}$ as basis vectors. The method we prefer, that used in "dot", uses a residue expansion $\mathbf{f}_{i,s}=\sum_{i=1}^{n} \frac{\alpha_i}{s-e_i}$ and thus chooses $\nu_{i,s}=\frac{1}{s-e_i}$ as a basis.

We do not use the obvious basis $\{s^i/e_s\}$ because coefficient-form expansions of filter transfer functions are known to be numerically poor [2,3]: for practical filter transfer functions the basis vectors $\frac{s^i}{e_s}$ can be very strongly correlated, so that small errors in coefficient values have a large effect on transfer functions.

One technique used in the literature to solve the conditioning problem for coefficients involves a change of variable [2] which "expands" the complex plane near passband edges. This kind of technique could, if necessary, be formulated as a selection of basis vectors and used for synthesis, but it is not clear that it would offer advantage over the simpler and more general residue expansion we use.

There are, of course, many ways to represent intermediate transfer functions other than in the form $\sum_{i=1}^{n} \alpha_i \nu_i$: for instance the product-form for polynomials $c_i \prod (s-s_i)$, [3] which is used for numerical reasons elsewhere in FILTOR2. This kind of formulation does not lead to linear equations in the $s_i$, and is therefore not a candidate for use in synthesis.

As will be seen, many important matrices have simple forms when the residue basis is chosen: this gives a “naturalness” to our choice.

**Notation**

An n-vector representing a function $\mathbf{f}_{i,s}$ will be written $\mathbf{F}_i$. Its $j^{th}$ element will be written $\mathbf{F}_{i,j}$. Thus

$$\mathbf{f}_{i,s}=\sum_j \mathbf{F}_{i,j}\nu_j \tag{6.1}$$

where $\nu_j$ is a basis vector.

A matrix of $n$ rows, each an $\mathbf{F}_i$ corresponding to **a** different $\mathbf{f}_{i,s}$ will simply be written $\mathbf{F}$, and is a matrix representation of the set $\{\mathbf{f}_i\}$.

Similarly, we define a matrix G to correspond to$\{\mathbf{g}_i\}$, so that

$$\mathbf{g}_{i,s}=\sum_j \mathbf{G}_{i,j}\nu_j \tag{6.2}$$

We will call the space of allowable intermediate transfer functions F, and a vector made of the n basis functions $\nu_s$. We will also use a vector $\mathbf{f}_s$ composed of the n $\mathbf{f}_{i,s}$ functions and a vector $\mathbf{g}_s$ composed of $\mathbf{g}_{i,s}$ functions. In matrix terms these may be written:

$$\mathbf{f}_s=\mathbf{F}\nu_s$$

$$\mathbf{g}_s=\mathbf{G}\nu_s$$

The net effect of this process of expressing functions in terms of a basis is to leave us dealing with a matrix of scalars F instead of a vector of functions $\mathbf{f}_s$.

This makes the formulae we develop easy to implement as programs.

---

***Example*** (61)

**Basis**

We use $\nu_i = \frac{1}{s - e_{ii}}$ in “dot”. If, say, $e_s = s^2 + \sqrt{2}s + 1$, and

$$\mathbf{f}_{1,s} = 1/e_s$$

$$\mathbf{f}_{2,s} = s/e_s$$

then

$$\mathbf{f}_{1,s} = \frac{j/\sqrt{2}}{s - e_1} + \frac{-j/\sqrt{2}}{s - e_2}$$

$$\mathbf{f}_{2,s} = \frac{j-1}{s - e_1} + \frac{-j-1}{s - e_2}$$

where $e_1 = \overline{e_2} = \frac{-1-j}{\sqrt{2}}$ and the $e_i$ are roots of $e_s$.

Now we have chosen $\nu_j = \frac{1}{s - e_j}$, and have

$$F = \begin{vmatrix} \frac{j}{\sqrt{2}} & \frac{-j}{\sqrt{2}} \\ \frac{j-1}{2} & \frac{-j-1}{2} \end{vmatrix}$$

---

As usual, $\mathbf{F}^{-1}$ denotes the inverse of a matrix and $\mathbf{F}^T$ its transpose. We use $\mathbf{F}^*$ to denote the conjugate of the transpose of F.

## 6.3 Complex Vectors

Since natural modes $e_i$ are generally complex, our basis functions $\frac{1}{s-e_i}$ will generally have complex coefficients. This in turn means that the coefficients $\alpha_i$ will be complex.

If we were only interested in transfer functions with real coefficients this would represent a minor inefficiency, since one could easily select a basis with real coefficients and make operations like matrix inversion quicker. This restriction does not seem to be either necessary or justifiable in terms of application: one of the major areas of application we explore later in this thesis involves transfer functions with complex coefficients. We show how to use these functions and how to make "complex systems" in chapter 9. All of the. mathematical operations we. are interested in on vectors are perfectly well-defined on vectors with complex coefficients.

## 6.4 Getting system coefficients from f functions

This section sketches the development of matrix formulae for intermediate-function synthesis. It gives results only for the "residue basis", although they are easily adapted to other bases. The required equations are derived simply by manipulating state equations. Taking the frequency-domain versions of the system equations and substituting $\mathbf{f}_s u_s$ for the vector of states:

$$s\,\mathbf{f}_s u_s = \mathrm{A}\,\mathbf{f}_s u_s + \mathbf{b} u_s \tag{6-3}$$

$$y_s = \mathbf{c}^T \mathbf{f}_s u_s + d u_s \tag{6-4}$$

There are several ways to go about solving these equations for {A,b,c,d}: we choose to emphasize vector-space methods to clarify the structure of the synthesis technique.

The equation

$$s\mathbf{f}_s = \mathbf{A}\mathbf{f}_s + \mathbf{b}1_s\dagger \tag{6-5}$$

obtained by dividing (6-3) by $u_s$ could be seen as an equation in our vector space $\boldsymbol{F}$ except for two things: that the function $1_s$ is not in $\boldsymbol{F}$ and that products $s\mathbf{f}_s$ may not be in $\boldsymbol{F}$.

Augmenting $\boldsymbol{F}$ by adding the function $1_s$ produces a new space $\boldsymbol{F}'$ of dimension ***n+l*** of which all functions in **(6-5)** are members.

In terms of coefficients we can understand this augmentation as giving us a way to deal with rational functions whose numerator order ***is n,*** while $\boldsymbol{F}$ only included numerators of order 0 to n-l.

Taking the portion of (6-5) corresponding to any row $\mathbf{A}_i$ of **A** yields

$$s\mathbf{f}_{i,s} = \sum_j \mathbf{A}_{i,j}\mathbf{f}_{j,s} + b_i 1_s \tag{6-6}$$

which expresses **a** vector of F' in terms of n independent elements of F and the "extra" vector $1_s$. This may be converted readily to the form of an equation in matrices (of scaIars). In doing so we will find the operation of "multiplying by s" to be very important.

Taking the projection of (6-5) onto F:

$$S_{LP}(\mathbf{f}) = \mathbf{A}\mathbf{f} \tag{6-7}$$

where $S_{LP}$ is an operator that takes the projection onto $\boldsymbol{F}$ of the result of multiplying its operand by "s" Because $S_{LP}(\cdot)$ is a bounded linear operator mapping $\boldsymbol{F}$ onto itself, a matrix expression exists for it:

† we use "$1_s$" to denote the function of s that is equal to "1" everywhere, while "1" is just a scalar. This distinction is introduced so that (6-5) may be seen as ***an*** equation in functions of s.

$$\mathbf{S}_{LP}(\mathbf{f})=(\mathbf{F}\mathbf{S}_{LP})\nu$$

We also need to know components on the vector $\mathbf{1}_s$ of the product of s and arbitrary functions. A vector $s_{HP}$ may be used to compute these just as $S_{LP}$ gives components on F.

Writing (6-6) in terms of the augmented basis $\{\nu_i, 1_s\}$

$$s\,\mathbf{f}_{i,s}=\sum_j \mathbf{F}_{i,j}\, s\, \nu_{j,s}$$

$$=\sum_{j,k} \mathbf{F}_{i,j}\, \mathbf{S}_{LP,j,k}\, \nu_{k,s}+\sum_j \mathbf{F}_{i,j}\, \mathbf{s}_{HP,j}\, 1_s$$

$$=\mathbf{F}\mathbf{S}_{LP}\,\nu+\mathbf{F}\mathbf{s}_{HP}\cdot 1_s$$

$$=\mathbf{A}\mathbf{F}\nu+\mathbf{b}\cdot 1_s$$

so that, comparing coefficients on the $\nu_k$

$$\mathbf{F}\mathbf{S}_{LP}=\mathbf{A}\mathbf{F} \qquad (6\text{-}8)$$

and comparing coefficients for the vector $\mathbf{1}_s$

$$\mathbf{F}\mathbf{s}_{HP}=\mathrm{b} \qquad (6\text{-}9)$$

Thus if we can just construct $\mathbf{S}_{LP}$ and $\mathbf{s}_{HP}$, we will have "**b**" from (6-9) and

$$\mathbf{A}=\mathbf{F}\mathbf{S}_{LP}\mathbf{F}^{-1} \qquad (6\text{-}10)$$

from (6-8) (assuming that F is invertible, which follows from mdependence of the $\{\mathbf{f}_i\}$).

In order to compute $\mathbf{S}_{LP}$ and $\mathbf{s}_{HP}$ we will look at the effects of multiplying a basis function by "s":

$$s\,\nu_i = s\cdot\frac{1}{s-e_i} = 1+\frac{e_i}{s-e_i} = 1+e_i\cdot\nu_i$$

This expresses $s\,\nu_i$ in terms of the augmented basis $\{\nu_i, 1_s\}$. It follows that (for this particular choice of basis)

$$\mathbf{S}_{LP} = diag\,(e_1, e_2, .. e_n) \qquad (6\text{-}11)$$

and

$$\mathbf{s}_{HP} = (1, 1, .. 1) \qquad (6\text{-}12)$$

We still have to compute $\mathbf{c}^T$ and $d$, but they are even easier than **A** and **b.**

When $t_s$ is expressed as a vector in $\boldsymbol{F}'$,

$$t_s = \sum_i \mathbf{t}_i\,\nu_{i,s} + \mathbf{t}_{n+1}1_s$$

$$= \sum_{j,i} \mathbf{c}_j\,\mathbf{F}_{j,i}\,\nu_{i,s} + d\,1_s \qquad \text{(from 6-4)}$$

It follows that

$$d = \mathbf{t}_{n+1} \qquad (6\text{-}13\}$$

and

$$\mathbf{c}^T = \mathbf{t}^T\mathbf{F}^{-1} \qquad (6\text{-}14)$$

## 6.5 G directly from F

Since we have formulae in terms of **F** for $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ we could obviously compute

$$\mathbf{g}^T=\mathbf{c}^T(s\mathbf{I}-\mathbf{A})^{-1},$$

but in fact a simple formula gives G directly from F, and so lays bare the relationship between them.

This formula, which we derive below, is simply:

$$\mathbf{G}^T=\mathbf{H}\mathbf{F}^{-1} \qquad (6\text{-}15)$$

where **H** is a diagonal matrix formed from the residues of the desired $t_s$ such that

$$t_s=\sum_i \mathbf{H}_{i,i}\nu_i \qquad (6\text{-}16)$$

This equation shows (as we might expect) that G is inversely related to F and directly to $t_s$. We would expect to see this because it just shows that decreasing signal levels increases output noise, and increasing the required gain level also increases noise. It also shows that allowing rows of F to approach linear dependencies will increase noise levels and sensitivities by increasing the norm of $\mathbf{F}^{-1}$ and hence of G.

**Derivation**

This section derives (6-11).

Let us investigate:

$$\mathbf{g}_s^T=\nu_s^T\mathbf{G}^T=c^T(s\mathbf{I}-\mathbf{A})^{-1}$$

Now

$$A=FS_{LP}F^{-1}$$

so

$$g_s^T=c^T(sI-FS_{LP}F^{-1})^{-1}$$

$$=c^T(F(sI-S_{LP})F^{-1})^{-1}$$

$$=c^TF(sI-S_{LP})^{-1}F^{-1} \qquad (6\text{-}17)$$

Now, expressing these things in terms of matrices that we already know, $g^T=\nu_s G^T$ and $c^T F=\mathbf{t}$, where $t_s=\nu_s^T\cdot\mathbf{t}$. The interesting term is $(sI-S_{LP})^{-1}$, (a matrix of functions) which turns out to be very easy to express in the residue basis $\{\nu_i\}$, for which we have seen that $S_{LP}=diag(e_1,\cdots,e_n)$. Substituting this $S_{LP}$ gives

$$(sI-S_{LP})^{-1}=diag(\nu_{1,s},\cdots,\nu_{n,s})$$

so that we can re-write (6-12) as

$$\nu_s^T G_s^T=\mathbf{t}_s^T diag(\nu_{1,s},\cdots)F^{-1}$$

$$=\nu_s^T diag(\mathbf{t}_1,\cdots,\mathbf{t}_n)F^{-1}$$

from which we can eliminate references to $\{\nu_i\}$ to get (6-15).

## 6.6 Inner Product

We will often need to take inner products between $\mathbf{f}_i$ functions, e.g. to calculate correlations between functions or to do scaling. It is straightforward to define inner products for vectors, and we do so in such a way as to give physical significance to the result. Chapter 4 discussed the issue of choosing a “norm” or measure of magnitude for signals. When an “rms” type of norm is chosen it can be made a special case of the inner product: $||\mathbf{f}_i||^2=\mathbf{f}_i\bullet\mathbf{f}_i$.

In general one may compute inner products among vectors by using a bilinear form in an hermitian matrix Q:

$$\mathbf{f}_i\bullet\mathbf{f}_j=\mathbf{F}_i \bullet \mathbf{Q}\mathbf{F}_j \tag{6-18}$$

All we need to do is find an expression for the matrix Q from a physically significant definition of inner-product and our basis $\{\nu_i\}$.

The most natural definition of inner product to use is the correlation between functions

$$\mathbf{f}_{i,s}\bullet\mathbf{f}_{j,s}\triangleq\int_\omega \overline{\mathbf{f}_{1,s}(j\omega)}\mathbf{f}_{2,s}(j\omega)d\omega \tag{6-19}$$

because this inner-product then produces a “norm” or measure of magnitude that is proportional to the mean-square signal level at the output of a transfer function when its input is white noise. It may sometimes be useful, as when the spectrum of the signal exciting a filter is known and non-white, to modify this definition by a weighting function. We will first look at the simple case (6-19).

One may compute elements of Q by deriving values for the inner products among basis vectors. The $(i,j)$ element of Q is just $\overline{\nu_i}\bullet\nu_j$. We can compute general inner products among $\{\nu_i\}$ from the definition (6-13).

$$\overline{\nu_i}\bullet\nu_j=\int_\omega\overline{\nu_i}(-j\omega)\nu_j(j\omega)d\omega$$

$$=\int_\omega\frac{1}{-j\omega-\overline{e_i}}\cdot\frac{1}{j\omega-e_j}d\omega$$

$$=\int_{\omega}\left[\frac{1/(e_j+\overline{e_i})}{j\omega+\overline{e_i}}+\frac{1/-(e_j+\overline{e_i})}{j\omega-e_j}\right]d\omega \qquad \text{(6-19\}}$$

Assuming stability, both $e_i$ and $e_j$ are in the left half-plane, so that (6-19) has one pole in each half-plane. We can use Cauchy's integral formula [6] to compute this integral from the left half-plane residues $\alpha_i$ of the integrand:

$$\mathbf{Q}_{i,j}=\overline{\nu_i}\bullet\nu_j=2\pi j\sum\alpha_i$$

$$=\frac{-2\pi j}{\overline{e_i}+e_j} \qquad \text{(6-20)}$$

Note (as a useful check) that one can deduce from (6-20) that diagonal entries $\mathbf{Q}_{i,i}$ are real and positive: i.e. that the inner product of a basis vector with itself (its squared norm) is real and positive.

One may include the effects of non-white spectra in (6-18) by including a rational weighting function $W_s$ (bounded on s $=j\omega$) without much difficulty Chapter 4 discussed reasons for doing this, and use of the Cauchy integral formula as above allows one to compute a suitable Q matrix.

In general one might want different weighting functions for $\{\mathbf{f}_i\}$ **and** $\{\mathbf{g}_i\}$, so **that two inner product matrices $\mathbf{Q}_f$ and $\mathbf{Q}_g$ might be needed.**

## Computing K and W

Using **Q as** derived above, one may compute the K and **W** matrices of [Mull76] as

$$K=\mathbf{FQF}^T \qquad \text{(6-22)}$$

$$\mathbf{W}=\mathbf{GQG}^T \qquad \text{(6-23)}$$

**Inner products with $1_s$**

It might be useful to be able to compute norms in the augmented space $F'$. Under the definition (6-19) for inner product, however, the "norm" of $1_s$ is $\infty$! This simply states that the energy in a delta function is infinite.

$Q_{n+1,n+1}$ is therefore not finite for (6-19), but for a weighted measure like (6-20) where $W_s$ tends to zero as $s \to \infty$ (like A-weighting and the spectra of practical signals) all inner products are perfectly well-defined.

We do not pursue this further because we are primarily interested in $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$, which are in F.

**Other Frequency Variables**

The inner-product formulae above are good for use in the "s"-plane, but incorrect for, e.g., the discrete-time case.

A matrix Q may still be defined, however, so that (6-16) applies.

If we choose

$$\mathbf{f}_i \bullet \mathbf{f}_j \triangleq \int_{|z|=1} \overline{\mathbf{f}_{i,s}(z)}\mathbf{f}_{j,s}(z)dz$$

we will have a correlation measure for the "z"-plane.

## 6.7 An **Important Invariant**

A formula derived from (6-15) yields an important invariant quantity which we use later to study optimization of sensitivity. Muhiplying (6-15) by F yields

$$\mathbf{G}^T\mathbf{F}=\mathbf{H} \tag{6-24}$$

We can interpret this formula in terms of functions $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ rather than matrices by inserting $\nu_s$ as follows:

$$\nu^T \mathbf{G}^T \mathbf{F}\nu = \nu^T \mathbf{H}\nu$$

$$=> \quad \mathbf{g}^T \mathbf{f} = \sum_i \mathbf{H}_{i,i}\nu_i^2$$

$$=> \quad \sum_i \mathbf{f}_{i,s}\mathbf{g}_{i,s} = \sum_i \frac{\mathbf{H}_{i,i}}{(s-e_i)^2} \qquad (6\text{-}25)$$

Now note that $t_s = \sum_i \frac{\mathbf{H}_{i,i}}{(s-e_i)}$, so $d\frac{t_s}{ds} = \sum_i -\frac{\mathbf{H}_{i,i}}{(s-e_i)^2}$. thus

$$\sum_i \mathbf{f}_{i,s}\mathbf{g}_{i,s} = -t_s{}' \qquad (6\text{-}26)$$

The sum over all products $\mathbf{f}_{i,s}\mathbf{g}_{i,s}$ is independent of the realization!

Now in fact this identity (which is used heavily in Chapter 7') may be related to a cIassica1 sensitivity result. Because (cf. section 4.3)

$$S^{t_s}_{\gamma_i} = \mathbf{f}_{i,s}\mathbf{g}_{i,s} \cdot \frac{s}{t_s} \qquad (4\text{-}11)$$

we can combine (4-11) and (6-22) to get

$$\sum_i S^{t_s}_{\gamma_i} = \frac{s}{t_s} t_s{}' \qquad (6\text{-}27')$$

This last result can be obtained in general for active-RC networks: because terms in "s" can only derive from capacitor admittances $sC$,

$$\sum_i S^{t_s}_{C_i} = S^{t_s}_{s} = \frac{s}{t_s} \cdot d\frac{t_s}{ds}. \qquad (6\text{-}26)$$

The form (6-26), however, is interesting because it allows us to relate a minimum sensitivity condition to our synthesis method (cf. section 7.3)

## 6.8 Transformations

Several authors [4,5] discuss “equivalent” systems in terms of similarity transformations. Any canonic system $\{\tilde{\mathbf{A}},\tilde{\mathbf{b}},\tilde{\mathbf{c}},\tilde{d}\}$ with the same transfer function as a canonic system $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ may be derived by transforming states by an (invertible) matrix **T**:

$$\tilde{f}=\mathbf{T}f \tag{6-29}$$

This affects other quantities as follows:

$$\{\tilde{\mathbf{A}},\tilde{\mathbf{b}},\tilde{\mathbf{c}},\tilde{d}\}=\{\mathbf{TAT}^{-1},\mathbf{T}b,c\mathbf{T}^{-1},d\} \tag{6-29}$$

$$\tilde{\mathbf{F}}=\mathbf{TF}$$

$$\tilde{g}=\mathbf{T}^{-T}g$$

$$\tilde{\mathbf{G}}=\mathbf{T}^{-T}\mathbf{G}$$

$$\tilde{\mathbf{K}}=\mathbf{TKT}^{T}$$

$$\tilde{\mathbf{W}}=\mathbf{T}^{-T}\mathbf{WT}^{-1}$$

This way of looking at changes in a system design can be very useful. In particular, it makes **it easy to** study the effects of some simple special **cases.**

### Scaling

**One** may "scale" a system to control the magnitudes of states $x_i$ by choosing a diagonal $\mathbf{T}$. In this case $\mathbf{T}^{-1}$ is trivially computed and the effects on other quantities are obtained from (6-29).

Scaling does not affect the sensitivity of a system to the gains of its integrators or **{A,b,c,d}** coefficients, but does control overflow and noise and affects the sensitivity of Miller integrators to their amplifiers (cf. section 3.2.1).

### Transforming Subsystems

If one chooses to change only **a** few $\{\mathbf{f}_i\}$, and to change them so that the changed $\{\tilde{\mathbf{f}}_i\}$ span the same subspace of $\boldsymbol{F}$ as the originals, a simple kind of $\mathbf{T}$ emerges. This $\mathbf{T}$ is different from the identity matrix only in rows and columns corresponding to changed $\{\mathbf{f}_i\}$.

In particular, changing two particular $\{\mathbf{f}_i\}$, say $\mathbf{f}_{i,s}$ and $\mathbf{f}_{j,s}$, **so** that

$$\begin{bmatrix} \tilde{\mathbf{f}}_{i,s} \\ \tilde{\mathbf{f}}_{j,s} \end{bmatrix} = \begin{bmatrix} \alpha_{11} & \alpha_{21} \\ \alpha_{21} & \alpha_{22} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{f}_{i,s} \\ \mathbf{f}_{j,s} \end{bmatrix}$$

gives

$$\begin{bmatrix} \tilde{\mathbf{g}}_{i,s} \\ \tilde{\mathbf{g}}_{j,s} \end{bmatrix} = \frac{1}{det(\alpha)} \cdot \begin{bmatrix} \alpha_{22} & -\alpha_{21} \\ -\alpha_{12} & \alpha_{11} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{g}_{i,s} \\ \mathbf{g}_{j,s} \end{bmatrix}$$

This kind of transformation does affect some system sensitivities, and so could be used for introducing controlled amounts of redundancy. This kind of transformation is used in [5] to iteratively improve the dynamic range of a system.

## 6.9 Overview of 'dot'

“Dot” is a computer program, compatible with the FILTOR2 package [1] which has been developed in parallel with work on this thesis. Because it is a design tool, and because the purpose here is to show that intermediate-function synthesis is a useful design aid, a description of “dot” is a good way of demonstrating the thesis.

The major function of “dot” is to implement the algorithms of this chapter. It allows one to design and (to some extent) analyze state-variable systems by intermediate-function synthesis. Many of the features required for design have been implemented together with some others that are useful for studying a design before implementation. This section discusses implemented features together with some others that might be useful additions in the general context of design.

**Setting Functions** – “dot” allows the user to type in polynomials ($p_s$, $e_s$, numerators of $\{\mathbf{f}_i\}$) or will read their values from a file created by any other FILTORZ program. In particular, $p_s$ and $e_s$ may be produced by “remez” and “polman” which handle various phases of approximation. Ladders may be simulated by having “ladman” define $\{\mathbf{f}_i\}$ on the design file in such a way that integrator outputs simulate arbitrary voltages, currents, or even wave variables in a ladder.

Polynomials are usually defined by giving a list of their roots together with a Ieading coefficient. “Dot” also allows $\{\mathbf{f}_i\}$ and $p_s$ to be defined as lists of residues.

One may set $t_s$ and $\{\mathbf{f}_i\}$, investigate the resulting system, and then change one or more $\{\mathbf{f}_i\}$ and try again.

**Getting System Coefficients** – “dot” has a “get-system” command which (if $p_s$, $e_s$ and $\{\mathbf{f}_i\}$ have been defined) computes $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ system entries using formulae (6-10), **(6-9}, (6-14)** and **(6-13).**

**Getting g** – the $\{g_i\}$ functions may be computed from $\{f_i\}$ and $t_s$ using equation (6-15) with a "get-g" command.

**Displaying Functions, Matrices, and vectors** – "dot" has a "show" command to allow the user to inspect **{A,b,c,d}** coefficients, K and W matrices, the inner product matrix Q, and the various polynomials.

$\{f_i\}$, $\{g_i\}$ and $t_s$ may be displayed either in residue form or as lists of roots. Q, K and W are computed when needed by formulae (6-20), (6-22) and (6-23).

scaling – a "scale" command allows the user to scale one or more $\{f_i\}$ **to** have unit norm. Other quantities $(\{A,b,c,d\},K,W,\{g_i\})$ are scaled simultaneously.

**Analyzing Performance** – a reverse-polish notation "calculator" is included, which is capable of computing and printing or plotting arbitrary functions of the $\{f_i\}$, $\{g_i\}$, $p_s$ and $e_s$. This powerful feature may be used with the formulae of chapter 4 to calculate sensitivities, expected transfer function deviations from ideal, signal and noise levels, and so forth.

**Getting Explanations**– the "?" command of FILTOR2 is included, so that an on-line tutorial on filter design can be made available to the user. Little text specifically on "dot" is included as yet, but it seems that a sophisticated tool like intermediate-function synthesis will only be widely usable if help is available.

## 6.10 Summary and Conclusions

We have presented a set of matrix formulae by means of which intermediate-function synthesis may be done. Computer programs using these formulae have been used for many designs during the course of this work, and have proven to be powerful design tools.

## 6.11 References

[1] W.M. Snelgrove, ***FILTOR2 User's Manual*** University of Toronto

[2] H.J Orchard and G.C. Temes, "***Filter Design Using Transformed Variables",*** IEEE Trans. Circuit Theory, CT-15, pp. 385-408, Dec. 1968

[3] J.K. Skwirzynski "On ***Synthesis of Filters",*** IEEE Trans. Circuit Theory, CT-16 pp. 152-163, Jan. 1971

[4] R.F. Mackay, ***"Generation of Low- Sensitivity State- Space Active Filters",*** Ph.D. Thesis, University of Toronto, Toronto, Canada 1980

[5] C.T. Mullis and R.A. Roberts, ***Synthesis of Minimum Roundoff Noise Fixed- point Digital Filters*** IEEE Trans. Circuits and Systems, vol. CAS-23, pp. 551-562, Sept. 1976.

[6] W. Kaplan, ***Advanced Calculus*** Addison-Wesley, 1952

# 7. Minimum-Sensitivity Filters

This chapter presents new results in filter design that have been obtained at least partly by means of intermediate-function synthesis. It concentrates in particular on the problem of designing filters with optimum or near-optimum sensitivity to their integrators.

## 7.1 Importance of Integrator Sensitivity

Optimizing sensitivity to integrators addresses only part of the general problem of designing a good filter, but it appears to be an important part for two reasons: that integrators are often the “weak link”: and that designs insensitive to their integrators appear to have good dynamic range, low component spreads, and low sensitivities to system coefficients in general. This section discusses these points,

### 7.1.1 Relationship to Dynamic Range

It was shown in [ 1,2] that there is a close relationship in digital filters between noise performance and coefficient sensitivities. Holder’s inequality was used in [1] to derive a lower bound on roundoff noise from the sensitivity to multipliers. The same relationship holds in continuous-time systems between integrator sensitivities and dynamic range.

The best possible (after scaling) dynamic range at integrator $i$ is proportional to $\|\mathbf{f}_{i,s}\|\|\mathbf{g}_{i,s}\|$, while $\left| \frac{d\,t_s(s)}{d\gamma_i} \right|$ is proportional to $\|\mathbf{f}_{i,s}\mathbf{g}_{i,s}\|$. It is clear that “large” $\mathbf{f}_{i,s}$ and $\mathbf{g}_{i,s}$ worsen both types of measure. A precise statement of the relationship is provided here, just as for the digital case, by Holder’s inequality

$||f_{i,s}\, g_{i,s}||_1 \leq ||f_{i,s}\, ||_2 ||g_{i,s}||_2$

We will demonstrate in section 7.3 that, for second-order filters, optimum integrator sensitivity implies optimum dynamic range.

We thus do not expect the goals of minimizing integrator sensitivity and maximizing dynamic range to conflict, but rather that improving sensitivity wil1 improve dynamic range.

**Integrator Sensitivity in Ladders**

The good sensitivity behaviour of ladders with respect to their reactive components at reflection zeros can be seen as guaranteeing good performance with respect to their “integrators”, namely their LC components. Design by ladder simulation seeks to obtain the same low sensitivities to active-RC integrators. Thus a good solution to the problem of minimizing integrator sensitivities should be at least as good as an LC simulation.

## 7.2 A Condition for Optimum Integrator Sensitivity

It was shown in section 6.7 that the sum of all integrator sensitivities is invariant with realization. The reason some designs are better than others is that the performance of the overall system is determined, not by the sum $\sum S_{\gamma_i}^{t_s(s)}$ but by some more complicated aggregate measure, like $\sum | S_{\gamma_i}^{t_s(s)} |^p$. We show in [3] that the resulting problem of optimizing the aggregate function under the constraint of a constant sensitivity sum is solved, for a wide variety of reasonable aggregate measures, by forcing all integrator sensitivities to be equal.

In general, a constraint of the form

$$\sum_i x_i = k$$

where ***k*** is a constant, results in the choice $x_1 = x_2 = \cdots = k/n$ minimizing any function of the form

$$\sum_i |x_i|^{p}, \ p \geqq 1$$

as well as functions such as

$$\sum_i |\mathrm{Re}(x_i)|^{p}$$

In our case, where the $x_i$ are various sensitivities, this means that forcing sensitivities to be equal ***simultaneously*** optimizes most reasonable sensitivity meas-***ures!*** We therefore seek systems in which, for all $i$,

$$S^{t_s(s)}_{\gamma_i} = -\frac{s}{nt_s(s)} d\frac{t_s(s)}{ds} \qquad (7\text{-}1)$$

For convenience, we will call any lower bound on an aggregate sensitivity measure implied by the constant sensitivity sum (6-27) a “Frequency-Scaling Lower Bound” (FSLB) and call the filter (if any) that attains this bound an FSLB filter. The name is chosen because the lower bound derives from the simple fact that changing all integrator gains together just results in frequency scaling,

We will use IF synthesis to derive a construction for FSLB systems when they exist, and produce as a side-effect several ***new*** and interesting results concerning their existence.

When equal sensitivities ***cannot*** be obtained one is forced to accept approximate equality. Systems may approximate equality in several different senses, so that “almost FSLB” systems need not be optimum for the wide range of different sensitivity aggregates that FSLB systems simultaneously optimize.

## 7.3 Equalizing Sensitivities

Combining the formula for optimum sensitivity (7-l) with that for integrator sensitivity in terms of $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ (4-11) yields

$$\mathbf{f}_{i,s}\mathbf{g}_{i,s} = -\frac{1}{n} d\frac{t_s(s)}{ds} \quad \forall i \tag{7-2}$$

Unfortunately we don’t have any way to select $\mathbf{f}_{i,s}\mathbf{g}_{i,s}$, but IF synthesis does allow us to pick arbitrary $\mathbf{f}_{i,s}$ (or $\mathbf{g}_{i,s}$). Inspection of (7-2) reveals **that all $\{\mathbf{f}_i\}$ must be factors** of $dt_s(s)/ds$, and because there are only a few factors to choose from we can simply try all of the various possibilities to see if any of them produce the appropriate $\{\mathbf{g}_i\}$.

**Example: Optimum Biquadric Filters**

. To see how all of this works we shall consider the case of a second-order low-pass filter,

$$t_s(s) = \frac{a}{s^2 + s\left(\frac{\omega_o}{Q}\right) + \omega_o^2} \triangleq \frac{a}{e_s(s)}$$

The derivative is easily found:

$$\frac{dt_s(s)}{ds} = -\frac{a\left(2s + \frac{\omega_o}{Q}\right)}{e(s)^2}$$

Now we know from (?-2) that $\mathbf{f}_{1,s}(s)$ and $\mathbf{f}_{2,s}(s)$ must divide $[dt_s(s)/ds]$, and we also know that they must be linearly independent. Choosing

$$\mathbf{f}_{1,s}(s) = 1/e_s(s)$$

and

$$\mathbf{f}_{2,s}(s)=(s+\frac{\omega_o}{2Q})/e_s(s)$$

satisfies both conditions. In fact (apart from trivial changes such as scaling and interchange of $\mathbf{f}_{1,s}$ and $\mathbf{f}_{2,s}$) this is the only choice possible.

Using IF synthesis we can now obtain the signal-flow graph realization of Fig. 7.1 and evaluate the g functions as

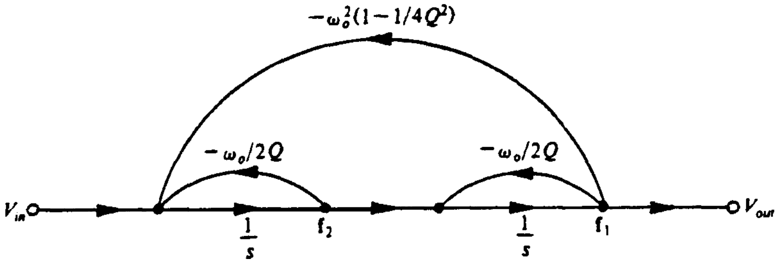


**Figure** 7.1: **Optimum LP Biquad**

$$\mathbf{g}_{1,s}=a(s+\frac{\omega_o}{2Q})/e_s(s)$$

$$\mathbf{g}_{2,s}=a/e_s(s)$$

We thus see that

$$\mathbf{f}_{1,s}\mathbf{g}_{1,s}=\mathbf{f}_{2,s}\mathbf{g}_{2,s}=-\frac{1}{n}\frac{dt_s(s)}{ds}$$

which is the condition for optimum sensitivity. Thus the realization of Fig. 7.1 is the optimum sensitivity realization of the second-order low-pass function. Note that for this structure

$$\mathbf{f}_{1,s}=a\,\mathbf{g}_{2,s}$$

and

$$\mathbf{f}_{2,s}=a\,\mathbf{g}_{1,s}$$

This can be shown to give the condition derived in [4] for a second-order system to have optimum dynamic range (after $L_2$ scaling, which does not affect integrator sensitivities). In fact biquads with. this “reciprocity” property always have equal integrator sensitivities, If (for any a)

$$\mathbf{g}_{1,s} = \alpha \mathbf{f}_{2,s}$$

and

$$\mathbf{g}_{2,s} = \alpha \mathbf{f}_{1,s}$$

then

$$\mathbf{f}_{1,s}\, \mathbf{g}_{1,s} = \mathbf{f}_{2,s} \mathbf{g}_{2,s}$$

We thus conclude that optimum integrator sensitivity produces optimum dynamic range in biquads.

The same procedure can be applied to bandpass, high-pass, all-pass and notch biquad transfer functions, and so produces optimum integrator sensitivity structures for most of the interesting biquads. It should be mentioned, however, that for certain transfer functions (primarily those with very low-Q poles) it is not possible to find two suitable factors of $[dt_s(s)/ds]$ for $\mathbf{f}_{1,s}$ and $\mathbf{f}_{2,s}$.

Table 7.1 lists FSLB choices for $\{\mathbf{f}_i\}$ for some important types of biquad.

<table>
<tr><th colspan="4">$e_s(s) \triangleq s^2 + \frac{\omega_o}{Q} s + \omega_o^2$</th></tr>
<tr><td>Type</td><td>$p_s(s)$</td><td>$\mathbf{f}_{1,s}(s) \cdot e_s(s)$</td><td>$\mathbf{f}_{2,s}(s) \cdot e_s(s)$</td></tr>
<tr><td>LP</td><td>$1$</td><td>$1$</td><td>$s + \frac{\omega_o}{2Q}$</td></tr>
<tr><td>BP</td><td>s</td><td>$s + \omega_a$</td><td>$s - \omega_o$</td></tr>
<tr><td>HP</td><td>$s^2$</td><td>s</td><td>s $+ 2Q\, \omega_o$</td></tr>
</table>

**Table 7.1: Optimum Biquads**

## 7.4 Relationship to LC Ladders

In general an $n^{th}$ order filter with equal integrator sensitivities is insensitive to integrator gain errors at the reflection zeros (i.e. frequencies of maximum transmission) $\omega_r$ because

$$S_{\gamma_i}^{|t_s(j\omega_r)|}=\mathrm{Re}\left[S_{\gamma_i}^{t_s(j\omega_r)}\right]$$

$$=\mathrm{Re}\left[\frac{-j\omega_r}{n}\frac{1}{t_s(j\omega)}\frac{dt_s(j\omega)}{d(j\omega)}\right]_{\omega=\omega_r}$$

$$=\mathrm{Re}\left[-\frac{\omega_r}{n}\frac{d\ln t_s(j\omega)}{d\omega}\right]_{\omega=\omega_r}$$

Expressing $t_s(j\omega)$ as

$$t_s(j\omega)=|t_s(j\omega)|e^{j\varphi(\omega)}$$

results in

$$S_{\gamma_i}^{|t_s(j\omega_r)|}=\frac{-\omega_r}{n}\mathrm{Re}\left[\frac{d\ln|t_s(j\omega)|}{d\omega}+j\frac{d\varphi(\omega)}{d\omega}\right]_{\omega=\omega_r}$$

But since $\left.\frac{d\ln|t_s(j\omega)|}{d\omega}\right|_{\omega=\omega_r}=0$ it follows that

$$S_{\gamma_i}^{|t_s(j\omega_r)|}=0$$

This is the same properly that LC ladders which are designed for maximum power transfer between resistive terminations have. In the following we show that our optimum biquads either simulate the corresponding LC ladders (in which case they have the same sensitivity performance) or are new structures that exhibit lower sensitivities than ladders.

In the signal flow graph of Fig. 7.1 it turns out that the signals at the two nodes labelled $f_{1,s}$ and $f_{2,s}$ simulate the capacitor voltage and inductor current of the doubly terminated low-pass ladder in Fig. 7.2 which has maximum power transfer at its reflection zeros.

On the other hand, the signal flow graph of the optimum bandpass biquad, shown in Fig. 7.3, does ***not*** simulate a doubly-terminated bandpass ladder. To further investigate this point we demonstrate in Fig. 7.4, through a Thevinin equivalence, that for the bandpass case a doubly-terminated realization is no better than a singly-terminated one. The optimum structure of Fig. 7.3 may be compared as to sensitivity with one simulating one of the ladder of Fig. 7.4: the integrator sensitivity products $(f_{i,s}g_{i,s})$ are as shown in Table 7.2.

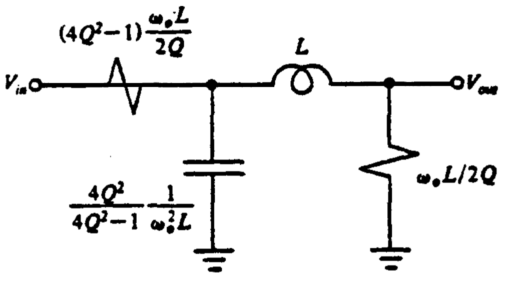

$(4Q^2-1)\frac{\omega_o L}{2Q}$
$L$
$V_{in}$
$V_{out}$
$\frac{4Q^2}{4Q^2-1}\frac{1}{\omega_o^2 L}$
$\omega_o L/2Q$

Figure 7.2: Insensitive **Lowpass Ladder**

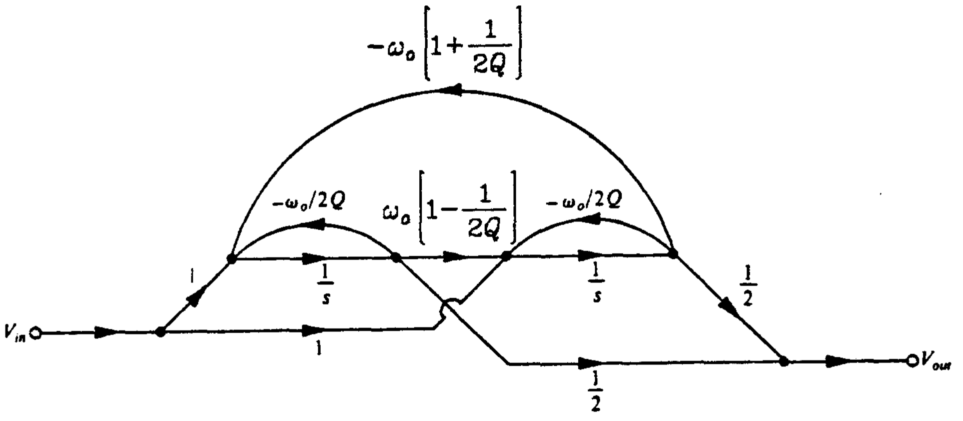


Figure 7.3: Bandpass FSLB Filter

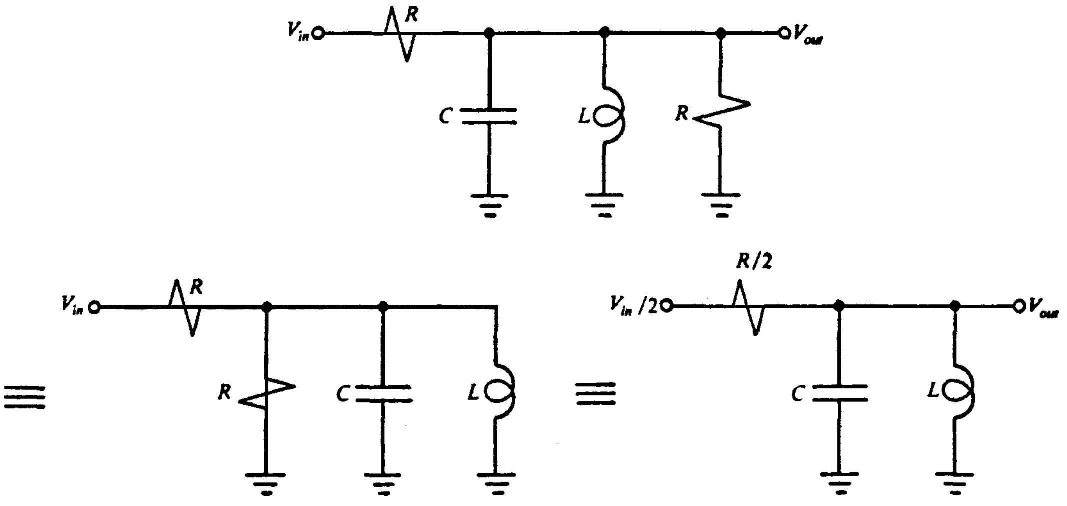


**Figure 7.4: Equivalent BP Ladders**

| Circuit | $f_{1,s}g_{1,s}$ | $f_{2,s}g_{2,s}$ |
|---|---|---|
| | | |
| Fig. 7.3 | $(s^2-\omega_0^2)/2e_s(s)^2$ | $(s^2-\omega_0^2)/2e_s(s)^2$ |
| Fig. 7.4 | $-\omega_0^2/e_s(s)^2$ | $s^2/e_s(s)^2$ |

**Table 7.2: Sensitivity Comparison: Ladder vs. Optimum.**

It is interesting to note that the two structures have equal sensitivity at the reflection zero $s\pm j\omega_o$, which is where the doubly terminated structure is known to be good. Our new structure becomes superior away from $j\omega_0$ (according to any aggregate sensitivity measure of the types discussed earlier).

## 7.5 When FSLB is and is not Attainable

The second-order case investigated above was quite special, in that for most interesting functions an FSLB realization was found to exist. In fact, while one may concoct higher-order transfer functions for which FSLB realizations are possible, they appear not to exist for most transfer functions. We will show some interesting results in this area.

As a matter of passing interest, there is only one canonic state-space realization for a first-order transfer function, and it is FSLB.

### 7.5.1 Low-Q Second-order

We mentioned above that some second-order transfer functions with very low pole-Q's did not admit of FSLB realization. Since they illustrate one of two ways in which FSLB filters can fail to exist, we will explore an example in detail.

The simplest example is the function $t_s(s) \triangleq \frac{s}{s^2-1}$; it is an unusual kind of function to choose to synthesize because it is unstable, but this has no effect on synthesis. The IF synthesis procedure does not contain assumptions about the detailed behaviour of the “operators” $s^{-1}$ which it combines to form given functions. It is therefore better to choose a physically uninteresting example that will make our development clear than to choose a stable one that involves messier algebra.

The derivative $t_s'(s) = -\frac{s^2+1}{(s^2-1)^2}$ must be factored two different ways to provide two independent $\{\mathbf{f}_i\}$. Only two factors with real coefficients exist however, $1/e_s(s)$ and $s^2/e_s(s)$, and one of them is improper (i.e. does not have a numerator of lower order than its denominator) and so not a possible $\{\mathbf{f}_i\}$. There is therefore no FSLB realization of this transfer function.

This result is novel in that it clearly shows that no FSLB realization of a particular transfer function can exist. It is also interesting in that an FSLB realization *is* obtained if complex coefficients are permitted, in which case $\{\mathbf{f}_{1,s},\mathbf{f}_{2,s}\}=\{\frac{1}{e_s(s)},\frac{s+j}{e_s(s)}\}$ is FSLB. The physical meaning of transfer functions with complex coefficients is discussed in chapter 9.

### 7.5.2 **OrderAbove 2**

At second order, the majority of useful transfer functions were capable of FSLB realizations and a few special ones were not. At higher order, unfortunately, the situation appears to be reversed. For some cases (e.g. $3^{rd}$ order LP Butterworth, $t_s(s)=\frac{1}{s^3+2s^2+2s+1}$) the problem is like that shown for $t_s(s)=\frac{s}{s^2-1}$ above: that there simply do not exist $n$ independent factors of $t_s(s)$ with real coefficients. For most transfer functions investigated by the author, however, the problem is different: that the $\{\mathbf{g}_i\}$ resulting from choosing $\{\mathbf{f}_i\}$ as factors of $t'(s)$ are not the complementary factors, so that $\{\mathbf{f}_{i,s}\mathbf{g}_{i,s}\}$ are not all equal. A design illustrating this point for an $8^{th}$ order transfer function appears in chapter 8. The $\{\mathbf{g}_i\}$ that are obtained there are close to, but not equal to, the complementary factors of $t_s'(s)$.

## 7.6 On FSLB-Realizability

We have presented a result that tells us how to synthesize a filter that attains the frequency-scaling lower bound on integrator sensitivity for an arbitrary transfer function if it is possible. We know of second-order cases where it does exist, but also know that it doesn't always exist. Further, we don't know too much about the topology of the resulting system, and so cannot tell whether to expect "dense" systems or not. We also have little insight into the reasons for the fact that some transfer functions "work" while others don't.

This section tries to solve these problems. It shows how to generate all FSLB systems known to the author, both showing what transfer functions "work" and what topologies optimally realize them. While the resulting transfer functions do not include all interesting ones (or perhaps even any) they are often “similar”. This may mean that the topologies that we derive here, which are often quite sparse, will provide respectable performance for general filters.

The section starts by showing a way of “coupling” identical sub-filters to maintain optimum integrator sensitivity. It then sketches a number of “building blocks” that may be coupled in this way, and presents a way of recognising functions of the form that may be derived this way.

**Theorem:** A transfer function $t_k$ has a realization attaining FSLB if it may be written in the form

$$t_{k,s} = \frac{(t_{k-1,s})^n}{(1+c\,(t_{k-1,s})^n)} \quad n>0$$

where there is an FSLB realization of $t_{k-1}$.

**Proof:** An implementation of $t_k$ is depicted in Figure 7.5

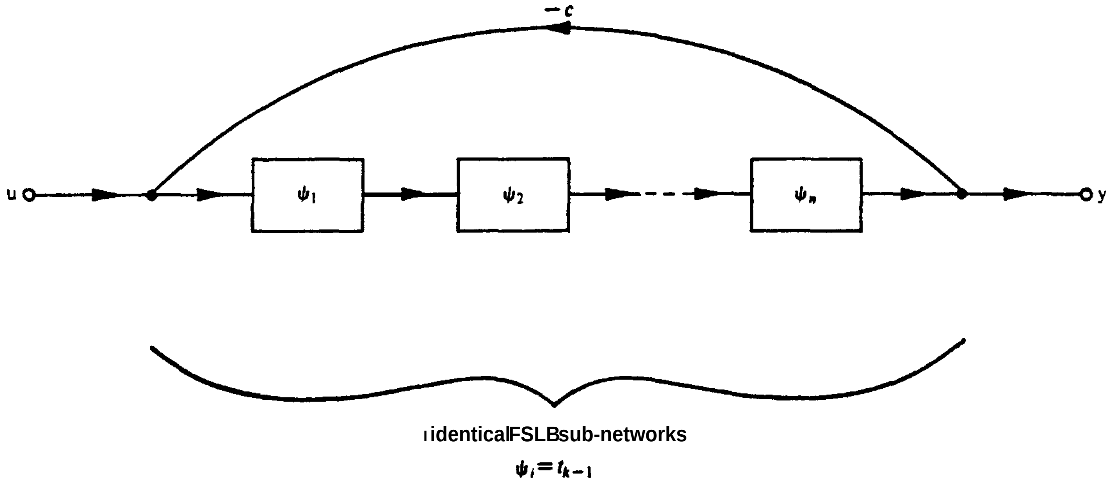


**Figure 7.5: An FSLB Topology**

To prove that this is an FSLB realization, we have to show that all capacitor sensitivities are equal. The circuit is composed of a number of identical FSLB sections (FSLB realizations of $t_{k-1}$, which exist by hypothesis). The interconnection topology arranges that the sensitivity of the overall transfer function $t_k$ to each section is the same, and this together with the fact that all sections are identical and FSLB makes $t_k$ FSLB.

More formally,

$$S^{t_k}_{C_{l,m}} = S^{t_k}_{\psi_l} \cdot S^{\psi_l}_{C_{l,m}} \tag{7-3}$$

where $\psi_l$ is the transfer function of the $l^{th}$ section in the circuit, and $C_{l,m}$ is the $m^{th}$ capacitor in the FSLB realization of that section.

Now we may investigate the second term of (7-3).

$$S^{\psi_l}_{C_{l,m}} = S^{\psi_n}_{C_{n,k}}$$

i.e. all sections have equal sensitivity to corresponding capacitors, because all sections are realized the same way. Also

$$S^{\psi_l}_{C_{l,m}} = S^{\psi_l}_{C_{l,n}}$$

i.e. all capacitor sensitivities within sections are equal because the realizations are FSLB. Thus all sensitivities of sections to all of their capacitors are equal. This means that the second term in (7-3) is the same for every integrator in the overall circuit.

Now we may investigate the first term: but all of the component $\psi_l$ are simply cascaded,

$$t_k = \frac{1}{c + \prod_l \frac{1}{\psi_l}}$$

So that

$$S_{\psi_l}^{t_k}=S_{\psi_m}^{t_k} \quad \forall k,m$$

Thus the first term in (7-1) is the same for all capacitors, and the overall transfer function $t_k$ has equal sensitivity to all integrators and must attain FSLB.

**Consequences**

We will show a few transfer functions and corresponding topologies that may be generated using this construction.

First note that the canonic realization of $t_{0,s}(s)\triangleq\frac{1}{s}$, an integrator, certainly has equal sensitivities to all of its integrators, since there is only one. Applying the theorem with n=l

$$t_{1,s}(s)=\frac{1}{s+c_1}$$

must have an FSLB realization as shown in figure 7.6. Again, since there is only one integrator, this is obviously FSLB.

Applying the theorem again with n=2 yields a function of the form

$$t_{2,s}(s)=\frac{1}{(s+c_1)^2+c_2}$$

which is the general second-order lowpass transfer function. Thus the topology of Figure 7.1 is an FSLB 2nd order Iowpass biquad.

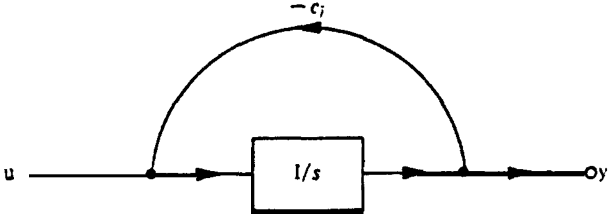


**Figure 7.6: FSLB Realization of $\frac{1}{s+c_1}$**

Figure 7.7 shows some FSLB topologies generated in this way together with s-plane diagrams showing possible resulting pole locations. in this way together with s-plane diagrams showing possible resulting pole locations.

There are a couple of special cases of the interconnection scheme above that are important enough to be worth mentioning. They are:

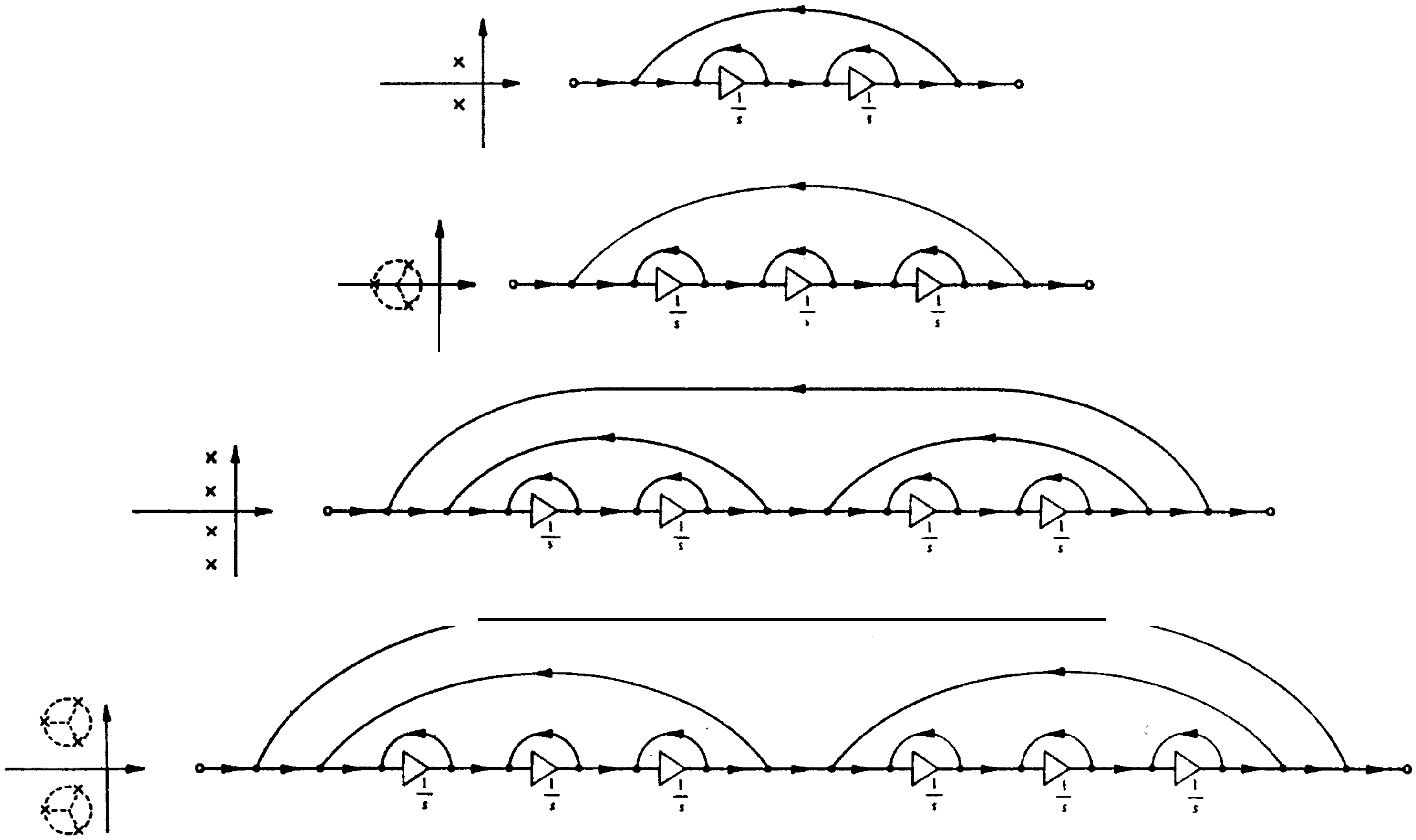


**Figure 7.7: Some FSLB Filters**

1. Cascade: functions of the form $t_{k,s}=(t_{k-1,s})^n$ are FSLB when realized by a cascade of FSLB sections. Thus functions with multiple roots are best realized by cascade connection of good realizations of sub-filters with single roots, and no coupling is required. When the sub-functions are similar, rather than exactly identical, as is often true of the biquadratic sections making up a good cascade filter design, then we might expect a (sub-optimum) good filter.

2. Overall Feedback: functions of the form $t_k(s)=\frac{t_{k-1}(s)}{t_{k-1}(s)-1}$ are generated by the case n=l of the theorem. This feedback may be used to split multiple poles, as in the example above that generated second-order lowpass filters.

There are also several other interesting manipulations of FSLB sections that produce FSLB results. Two simple ones are:

1. Gain: changing the required gain, $t_k=ct_{k-1}$ does not affect integrator sensitivity.

2. Adding a highpass component: functions of the form $t_k=t_{k-1}+d$ have the FSLB realization shown in figure 7.8 when an FSLB $t_{k-1}$ exists. Thus the first-order highpass function $\frac{s}{s+1}=1-\frac{1}{s+1}$ is an FSLB building block.

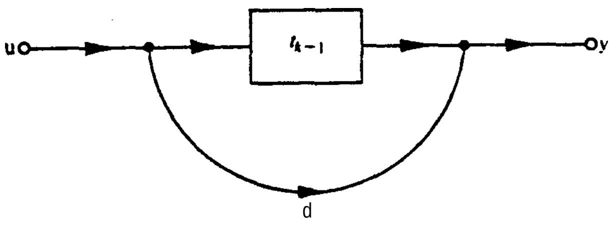


**Figure 7.8: FSLB circuit with Highpass Component**

Many other ways of combining FSLB sections into an FSLB structure may exist: the only other one known to the author is that shown in figure 7.9. Signal-flow graph analysis reveals that the transfer function of the overall circuit is

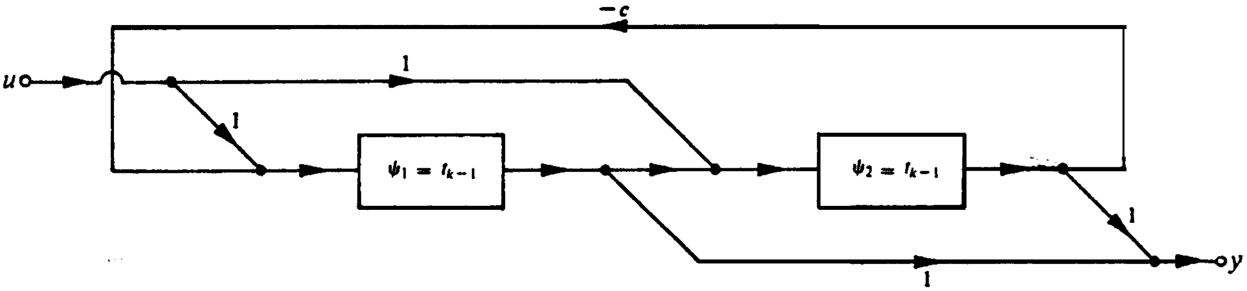


**Figure 7.9: FSLB Two-input Two-output Loop**

$$t_{k,s}(s)=\frac{\psi_{1,s}+\psi_{2,s}+\psi_{1,s}\psi_{2,s}(1-c)}{1+\psi_{1,s}\psi_{2,s}c} \qquad (7\text{-}4)$$

where $\psi_1=\psi_2=t_{k-1}$ are FSLB-realizable functions so that

$$t_{k,s}=\frac{2t_{k-1,s}+t^2_{k-1,s}(1-c)}{1+ct^2_{k-1}}$$

It is the fact that $\psi_1$ and $\psi_2$ appear symmetrically in expression ('7-4) that makes the topology retain the FSLB character of its components. It does not seem to be possible to extend this interconnection to three or more sections, although the $t_{k-1}$ may of course be of arbitrary order. When first-order sections are used for $\psi_i$ and a highpass component is added, bandpass and notch second-order sections may be obtained.

### 7.6.1 **Sub-functions in FSLB Filters**

As a contribution to understanding these FSLB filters, we will investigate their intermediate functions. There are really two distinct cases above: the cascade arrangement with feedback and the two-input two-output loop. We will show how the intermediate functions for each may be related to their component filters.

Let us define numerators and denominators of the transfer functions used to build up FSLB systems:

$$t_{k,s}(s) \triangleq \frac{p_{k,s}(s)}{e_{k,s}(s)}$$

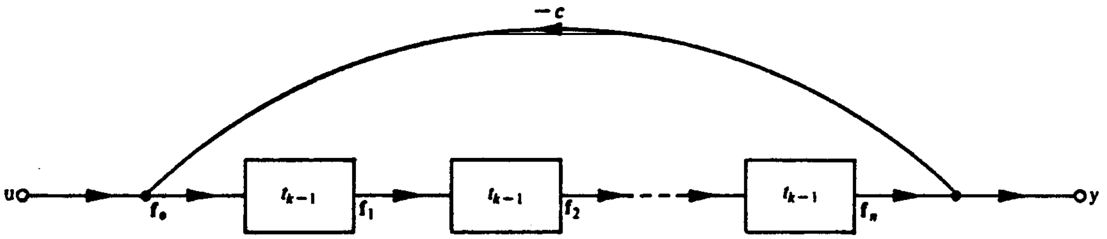


**Figure** 7.10: **FSLB Topology with Intermediate Functions**

Now the topology of Figure 7.11 has $n+$ 1 internal nodes, with transfer functions to these nodes which we shall call $\mathbf{f}_{k,0,s}, \mathbf{f}_{k,1,s} \cdots \mathbf{f}_{k,n,s}$ by analogy to intermediate functions using integrators as building blocks. Straightforward signal-flow graph analysis shows that all of these functions have denominator

$$e_{k,s} = e^{n}_{k-1,s} + c p^{n}_{k-1,s}$$

and that

$$\mathbf{f}_{k,0,s} = e^{n}_{k-1,s} / e_{k,s}$$

$$\mathbf{f}_{k,1,s} = e^{n-1}_{k-1,s} p_{k\ ,s} / e_{k,s}$$

and, in general

$$\mathbf{f}_{k,i,s} = e^{n-i}_{k-1,s} p^{i}_{k-1,s} / e_{k,s}$$

with the last function forming the output:

$$\mathbf{f}_{k,n,s}=p^{n}_{k-1,s}/e_s(s)=t_{k,s}$$

The second major topology has two intermediate transfer functions (cf. Figure 7.11)

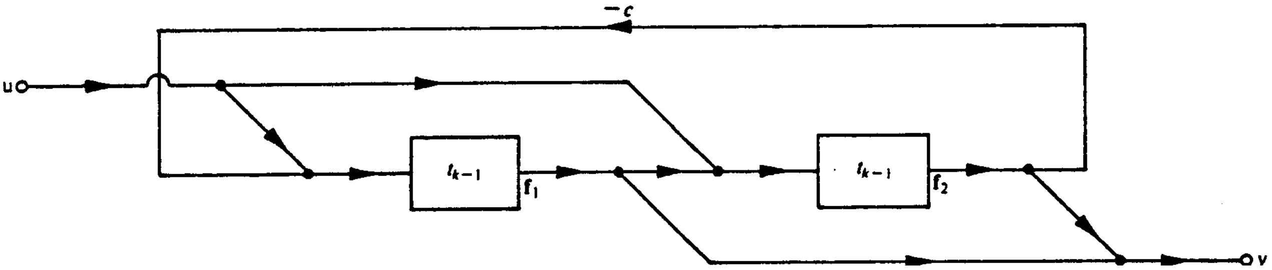


**Figure ‘7.11: FSLB Structure with Intermediate Functions**

with denominator

$$e_{k,s}=e^{2}_{k-1,s}+cp^{2}_{k-1,s}$$

and numerators:

$$\mathbf{f}_{k,1,s}=p_{k-1,s}e_{k-1,s}-cp^{2}_{k-1,s}/e_{k,s}$$

$$\mathbf{f}_{s,k,2}=p_{k-1}e_{k-1}+p^{2}_{k-1}/e_k$$

with output

$$t_k=\mathbf{f}_{s,k,1}+\mathbf{f}_{s,k,2}$$

$$=[2p_{k-1}e_{k-1}+(1-c)p^{2}_{k-1}]/e_k$$

We have presented these results for completeness, since they characterize structures originally developed in terms of intermdediate function synthesis by their intermediate functions. Inquiry into the relationships among $\{\mathbf{f}_i\}$ generated **as** shown above is an obvious area for further research.

### 7.6.2 Composition of FSLB-realizable functions

The importance of these results on generation of FSLB systems is that they show two inherently very good filter topologies, and show how to combine good building blocks to get good systems. The resulting systems are inherently quite sparse and so practical.

The essential point about these systems is that they are obtained by composition of functions. One way of describing this result (where for convenience we use loss functions $h_s(s) \triangleq \frac{1}{t_s(s)}$) is this:

**Theorem:** A loss function $h_s(s)$ has an FSLB realization if it may be written in the form

$$h_s(s) = h_{a,s}(h_{b,s}(s))$$

where $h_a$ and $h_b$ both have FSLB realizations. Furthermore, the loss functions

$$h_{1.s}(s) = c_1(s + c_2)^n$$

and

$$h_{2.s}(s) = \frac{s^2 + \left[\frac{\omega_o}{Q}\right] s + \omega_o^2}{s}$$

are FSLB-realizable.

**Proof Outline:**

1. A system implementing $h_s(s)$ may be constructed by replacing integrators in an FSLB system realizing $h_a$ by FSLB sub-sections implementing $h_b$.
2. We have already shown FSLB implementations of $h_1$ and $h_2$.

### 7.6.3 Testing for FSLB-Realizability

We have a way of generating some FSLB-realizable functions: it would be useful to have some simple test that characterizes FSLB-realizable functions so that we could tell whether it was worth trying to "decompose" a given function into its FSLB constituents. This section presents an unusual-looking result that has some bearing on this ideal and may be a useful seed for a stronger result.

**Theorem:** Any function $h_s(s)$ of the class generated by arbitrary composition of

$$h_{1,s}(s) \triangleq c_1(s+c_2)^n$$

and

$$h_{2,s}(s) \triangleq s+\frac{1}{s}$$

(i.e. by composition of any two functions already in the class) is pure real at zeros of its derivative $d\frac{h_s(s)}{ds}$.

**Notes:**

1. This is exactly the class of FSLB-realizable functions that our composition theorem generated.
2. Notice the recurrence of the derivative $d\frac{h_s(s)}{ds}$: *we* know that it is of interest in FSLB synthesis from our earlier result that the intermediate functions of an FSLB filter must be factors of the derivative. This suggestive re-appearance of the frequency derivative partly motivates presentation of this result.

**Proof:** The proof is an induction: first we verify that the condition holds for $h_{1,s}$ and $h_{2,s}$; then that it holds for any composition of two functions for each of which it holds.

1. $\frac{dh_1}{ds}$ has **n-l zeros** at $s=-c_2$, **where** $h_1(-c_2)=0$, which is certainly pure real. When **n=l** there are no ***zeros*** of the derivative at all, and the condition is still satisfied.

$\frac{dh_2}{ds}=\frac{s^2-1}{s^2}$ has derivative zeros at s $=\pm 1$, and $h_s$ *(s)* has real coefficients and so is real at both those locations.

2. Now assuming

$$h_{a,s}'(s_o)=0=>\mathrm{Im}(h_{a,s}(s_o))=0 \qquad (7\text{-}5)$$

and

$$h_{b,s}'(s_o)=0=>\mathrm{Im}(h_{b,s}(s_o))=0 \qquad (7\text{-}6)$$

and that $h_{a,s}$ and $h_{b,s}$ have real coefficients, we need to show that

$$[h_{a,s}(h_{b,s}(s))]'|_{s=s_o}=0=>\mathrm{Im}(h_{a,s}(h_{b,s}(s_o)))=0$$

and that $h_{a,s}(h_{b,s}(s))$ has real coefficients. It is straightforward that compositions of rational functions with real coefficients are rational functions with real coefficients, so the trick is to show the first part.

Applying the chain rule:

$$LHS \triangleq [h_{a,s}(h_{b,s}(s))]'|_{s=s_o}$$

$$=h_{a,s}'(h_{b,s}(s_o))\cdot h_{b,s}'(s_o) \qquad (7\text{-}7)$$

So that the fact that the LHS is zero implies that one or other of the terms in (7-7) must be zero.

If the first term is zero, $h_{a,s}'(h_{b,s}(s_o))=0$, substituting $h_{b,s}(s_o)$ for the Variable in (7-5) gives that

$$h_{a,s}'(h_{b,s}(s_o))=0 \Rightarrow \mathrm{Im}(h_{a,s}(h_{b,s}(s_o)))=0$$

as required.

If the second term is zero, $h_{b,s}'(s_o)=0$, *we* have by (7-3) that

$$\mathrm{Im}(h_{b,s}(s_o))=0$$

But $h_{a,s}$ has real coefficients, so that

$$\mathrm{Im}(h_{b,s}(s_o))=0 \Rightarrow \mathrm{Im}(h_{a,s}(h_{b,s}(s_o)))=0$$

as required.

**Application:** We can use this result for three types of thing: to scan functions to see whether they might be FSLB (though the test is neither necessary nor sufficient); in the hope that FSLB functions other than those generated by our present set of rules (if any) might obey this condition; and in investigating non-FSLB functions.

As an interesting example, let us look again at the function $t_s(s)=\frac{s}{s^2-1}$, which we investigated in section 7.3.2 above because it was not FSLB-realizable (at least with real-valued integrators, cf. chapter 9). The zeros of the derivative $h_s'$ occur at $s=\pm j$, at which points $h_s(\pm j)=\pm\frac{j}{-2}$ is pure ***imaginary*.** This suggests that we could extend the class of FSLB-realizable functions by using the complex-vaIued integrators of chapter 9.

## 7.7 Summary and Conclusions

We have presented a variety of results concerning "FSLB" realizations of transfer functions, which have (when they exist) minimum sensitivities to their integrators. The sensitivity condition may be interpreted as meaning that these filters tend to average errors in any of their integrators over the entire system, a phenomenon that we use to derive a new approximation technique in chapter 10.

FSLB filters are important because they minimize a critical sensitivity and because they do so without compromising other sensitivities or dynamic range. We have also shown, in several different ways, that there are FSLB structures. This is important because these structures are quite sparse.

## 7.8 References

[1 L.B. Jackson, "Roundofl Noise *Bounds* Derived ***from Coefficient Sensitivities for Digital Filters***" IEEE Trans. Circuits and Systems, CAS-23, no. 8, Aug. 1976

[2] A. Fettweis, ***"Roundoff Noise and Attenuation Sensitivity in Digital Filters with Fixed- Point Arithmetic"*** IEEE Trans. Circuit Theory, CT-20, pp. 174-175, Mar. 1973.

[3] Snelgrove, W.M. and Sedra, A.S. ***State- Variable biquads with optimum integrator sensitivities,*** IEE Proc., Vol. 128, Pt. G, No. 4, August 1981 pp. 173-175

[4] L.B. Jackson, A.G. Lindgren and Y. Kim, ***Optimal Synthesis of Second-order State- Space Structures for Digital Filters.*** IEEE Trans. Circuits and Systems, vol. CAS-26, pp. 149-153, Mar. 1979.

# 8. IF Design

This chapter presents a design problem and its solution by means of IF synthesis. It does a detailed example several different ways, including both new types of design and several well-known topologies. Inclusion of the older topologies both permits comparison with the new designs and shows how one may use IF synthesis to generate known topologies, and so strengthens links between IF synthesis and prior art.

The problem is an $8^{th}$ order one taken from the technical Iiterature [1] whereas our previous examples have been of low order for purposes of clarity. Since filter design problems are often trivial at low orders a medium-order filter is more Iikely to interest a filter designer.

## 8.1 The Filter

The filter to be investigated meets an arithmetically symmetric set of specifications: it has a passband from 1kHz to 1.4kHz with 0.4dB of ripple and stopbands with 5OdB of attenuation below '7OOHz and above 17OOHz. We will work throughout the chapter with a frequency-normalized version of the filter in which the upper passband edge is scaled to 1 rad/sec. Table 8.1 lists the roots and leading coefficients of the various polynomials describing the transfer function. The list includes natural modes, transmission zeros, reflection zeros, and the numerator of the derivative of the transfer function (which we will use in a novel type of design).

| polynomial: | $e_s$ | $p_s$ | $f_s$ | $e'_s p_s - e_s p'_s$ |
|---|---|---|---|---|
| leading coefficient: | 1210.19 | 5.31436 | 3896.135 | 12862.8 |
| list of roots: | -.0681354±j.916348<br>-.06388±j.788556<br>-.0232875±j.710193<br>-.0262875±j1.00448 | 0<br>0<br>±j1.25008<br>kj.399736 | ±j.909524<br>±j.98919467<br>±j.796951<br>±j.723907 | -0.0357517±j.741621<br>-.0611769±j.851543<br>-.039558±j.9689114<br>.030864±j1.42125<br>.0057±j.3135<br>-.937406<br>.95568<br>0 |

**Table 8.1: Transfer Function for Example**

## 8.2 A Ladder Simulation

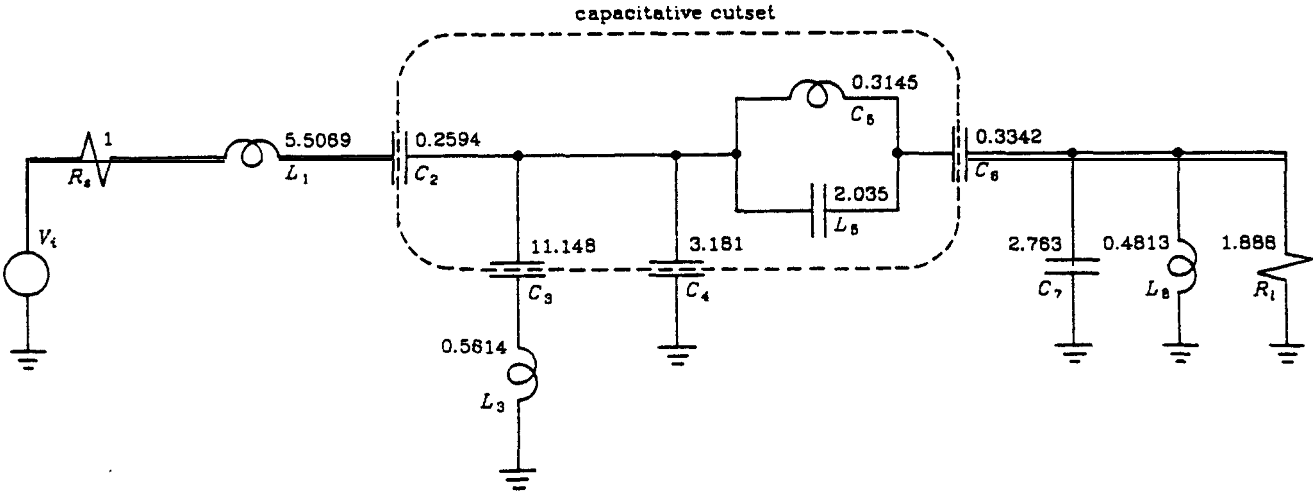


**Figure 8.1: LC Ladder to be Simulated**

Figure 8.1 shows a doubly-terminated lossless ladder, designed for maximum power-transfer between input and output, realizing the desired transfer function. It is the circuit given in [l]. This circuit contains ten reactive elements, two more than the canonic n. The simulation technique of [l] had an integrator for each of these eIements so as to maintain a one-to-one correspondence between circuit and simulation; it needed also to simulate some “parasitic” circuit elements with “reciprocators”, which were approximations to reactances of nominal value zero. The performance of these circuits was compromised by the reciprocators, which had poor high-frequency performance because of their short (nominally zero) time-constants.

We can simulate this circuit, as suggested in chapter 3, by choosing to simulate any independent set of eight of the natural “state variables”: inductor currents and capacitor voltages. Many ladder simulation strategies appear in the literature, some of which simulate other functions related to the ladder (e.g. wave variables) [2,3,4,5,6]. We could easily use IF synthesis to investigate any of the canonic techniques. This example will clearly show that it is important to simulate the ***right*** set of variables in a ladder, or all the advantages of ladder simulation will be lost. In particular, we will first choose to simulate a pIausible-looking set of states and will analyze the resulting system for its noise and integrator-sensitivity performance: they will both turn out to be quite poor, whereas the same figures for a differently-chosen set of states will be good.

By analyzing the ladder of figure 8.1 on FILTOR2 we may find its “intermediate transfer functions” (in terms of their poles and zeros), i.e. the transfer functions from ladder input to capacitor voltages and inductor currents. Table 8.2 lists the roots of the numerators of these functions: an interested reader might notice that many of the roots of these polynomials are quite close to roots of the derivative of $t_s$ (which we list again in the table for convenience), an interesting observation in view of the results we obtained for FSLB filters in chapter '7. The leading coefficients are of no real interest, since we will be scaling $\{\mathbf{f}_i\}$ for good dynamic range.

| state | numerator roots of intermediate function | numerator roots of derivative of $t_s(s)$ |
|---|---|---|
| $I_{L1}$ | -.022528±j.73095<br>-.04425±j.854146<br>-.02402±j.9829337<br>0 | -0.0357517±j.741621<br>-0.0611769±j.851543<br>-0.039558±j.9689114<br>0 |
| $V_{C2}$ | $I_{L1}/s$ | |
| $I_{L3}$ | -.0467423±j.7641159<br>-.04381265±j.957335<br>0 | |
| $V_{C3}$ | $I_{L3}/s$ | |
| $V_{C4}$ | ±j.39974<br>-.046742±j.76412<br>-.043813±j.95733 | 0.0057±j.3135 |
| $V_{C5}$ | 0, 0<br>±j.39974<br>-.095866±j.86193 | 0.95568 |
| $I_{L5}$ | $V_{C5}/s$ | |
| $V_{C6}$ | ij.399736<br>-.095866±j.86193<br>±j1.25008 | 0.030864±j1.42125 |
| $V_{C7}=t_s(s)$ | 0, 0<br>±j.39974<br>±j1.25008 | -0.9375 |
| $I_{L8}$ | $\frac{t_s(s)}{s}$ | |

**Table 6.2: Transmission Zeros From Input to States for Ladder**

### Selecting States to Simuiate

It is not possible to choose states to simulate arbitrarily: not all choices produce independent $\{\mathbf{f}_i\}$, and some choices simulate "extra" unwanted natural modes of the ladder.

In general, we will select $n$ suitable independent states out of those available so long as we do not simulate all of the voltages or currents in any one-reactance-kind cutsets or tiesets.

The internal structure of a ladder can contain natural modes (usually at DC) that are cancelled by transmission zeros in the input-output transfer function: this section explains their physical meaning.

The "extra" elements of an LC ladder can be thought of as providing redundancy, and thereby potentially improving performance (or at least performance measures), just as the simple example in section 5.4 obtained improved performance by introducing an extra state. Since the utility of extra states is controversial we shall avoid them in this chapter.

If we were to choose all of the capacitors in a capacitative loop (tieset), for instance $\{C_4,C_5,C_6,C_7\}$ in figure 8.1, Kirchoff's voltage law would give a linear dependence among states, $V_{C4}+V_{C5}+V_{C6}+V_{C7}=0$. Dually, choosing to simulate all of the currents in a cutset consisting wholly of inductors would produce a dependency among $\{\mathbf{f}_i\}$ by Kirchoff's current law.

It would also be a mistake to simulate all of the voltages in a capacitative cutset, for instance $\{C_2,C_3,C_4,C_6\}$ in figure 8.1, because by doing so we wouId be simulating an uncontrollable and unobservable state of the system. Notice, for example, that the central section of the ladder in figure 8.1 is not connected to ground at DC so that the charge on that section of circuit is in principle arbitrary. This charge cannot be changed by any input signal (i.e. is uncontrollable) and cannot be measured from the output (i.e. is unobservable). A complete simulation of this ladder would include a natural mode at DC (to model this charge) cancelled by a transmission zero (so that it has no effect on the input-output transfer function). We would prefer simply not to simulate this state, and thus eliminate the possibility of internal DC instability.

The dual of the capacitative-cutset problem is that the DC value of a current circulating in an inductive tieset is uncontrollable and unobservable.

To get a canonic system simulating only the useful (i.e. observable) part of this ladder we must throw away one state for each of these constraints. Thus, for instance, one cannot solve both problems in the ladder of figure 8.1 just by removing $C_4$ from the simulation even though its removal breaks both the sets causing trouble, because any resulting simulation would still have enough information to reconstruct the unobservable state (charge on the capacitative cutset) because $V_{C4}$ may be computed as $V_{C4}=-V_{C5}-V_{C6}-V_{C7}$.

### A First Choice

As a first attempt at simulating this filter, let us choose to simulate everything except $V_{C4}$ and $V_{C2}$: this obeys the rules above. Transfer functions from the input to the remaining 6 states may then be chosen as $\{\mathbf{f}_i\}$, and the methods of chapter 6 may then be used to scale $\{\mathbf{f}_i\}$ to have equal $L_2$ norms (so that rms signal levels at integrator outputs will be equal, cf. section 4.7).

“Dot”(cf. section 6.9} was used to do this, assuming that the input signal was white noise with a spectral power density of $1V^2/(rad/sec)$†. Intermediate-function synthesis by the methods of chapter 6 yields:

[1] “dot” uses the two-sided norm definition $||\mathbf{f}_i||_2 \overset{\Delta}{=} \left[\int_{-\infty}^{\infty} |\mathbf{f}_{i,s}(j\omega)|^2 d\omega\right]^{1/2}$ for consistency with the “complex filters” of chapter 9. For this reason we consider both positive and negative frequencies when computing noise band-width and spectral density. The filter designs of this section are therefore all scaled to give 1Vrms outputs with this two-sided deflnition of input signal spectral density set to $1V^2/(rad/sec)$, which would correspond in the more conventional one-sided approach to $2V^2/(rad/sec)$.

$$A=\begin{vmatrix} -0.1816 & 0 & 2.41 & 5.718 & 0 & 6.109 & 2.012 & 0 \\ 0 & 0 & -0.202 & 1.552 & 0 & 1.512 & 0.5463 & 0 \\ 0 & 0.7911 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.0815 & -0.1675 & 0 & 0 & -0.9235 & 0 & 0.0313 & 0.1459 \\ 0 & 0 & 0 & 0.9065 & 0 & 0 & 0 & 0 \\ -0.0594 & -0.1618 & 0 & 0 & 0.7734 & 0 & 0.0303 & 0.1409 \\ -0.0199 & -0.5417 & 0 & 0 & 0.2589 & 0 & -0.1616 & -0.8454 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.8426 & 0 \end{vmatrix} \quad b=\begin{vmatrix} -0.2814 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{vmatrix}$$

$$\mathbf{c}^T=[0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; 0 \;\; .7848 \;\; 0] \; d=0$$

There is already evidence that this is not a very good design: rows 1 and 2 of the A-matrix contain fairly large (i.e. larger than 1 in magnitude) elements in columns 4 and 6. The arguments of section 4.11 suggest that a subtraction of nearly equal $\mathbf{f}_4$ and $\mathbf{f}_6$ may be being used to form inputs $s\mathbf{f}_1$ and $s\mathbf{f}_2$.

We may compute $\{\mathbf{g}_i\}$ functions from our $\{\mathbf{f}_i\}$: together these two sets of functions tell us about noise and sensitivity. In particular, we can look at (figure 8.2) $\sum_i |\mathbf{g}_{i,s}(j\omega)|^2$, the total output noise power that we should expect at the output for white noise at integrator inputs. The resulting passband noise level may be seen to be about **6OdB** above integrator input noise.

We can use the **W** matrix described in section 4.8 to look at the total noise contributions of individual integrators: defining

$$\mathbf{W}_{ij} \triangleq \mathbf{g}_i \cdot \mathbf{g}_j \triangleq \int_{-\infty}^{\infty} \overline{\mathbf{g}}_{i,s}(j\omega)\mathbf{g}_{j,s}(j\omega)d\omega$$

gives us a definition of an inner product . on $\{\mathbf{g}_i\}$ that degenerates to a useful norm (cf. section 4.8). The diagonal elements of W are the rms noise levels that we should expect each integrator to contribute to the output if integrators have independent white input noise with a (two-sided, cf. footnote for $||\mathbf{f}_i||_2$ above) spectral density $1V^2/(rad/sec)$.

Investigation of the K and W matrices casts further suspicion on states 4 and 6: in particular,

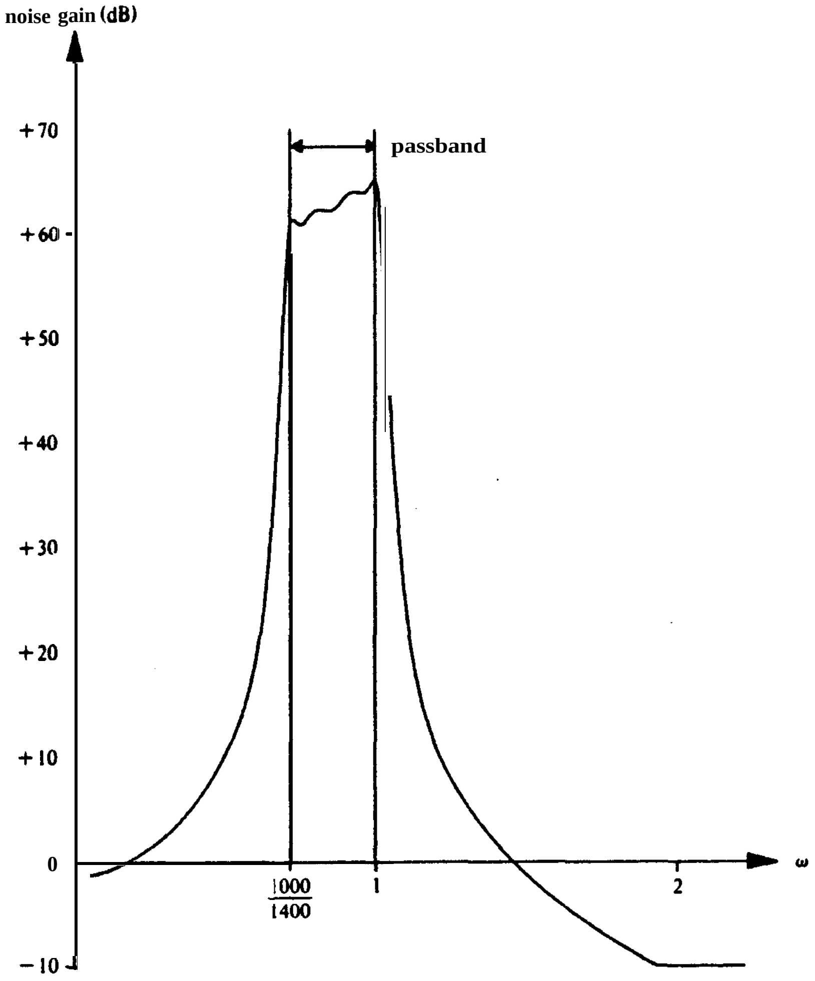


**Figure 8.2: Total Output Noise for First Ladder Simulation**

$W_{ii}$ = [ 7.76 9.76 66.2 345.4 9.76 406.5 49.7 7.6 ]

and so the total output noise for this $L_2$ scaled filter is about $\sum_i W_{ii} \cong 902.9$. This total may also be evaluated as the area under the curve of figure 6.2 (extended to negative frequency and converted from decibels). We will soon see that this is more than 1OdB above the optimum. Note that states 4 and 6 generate a lot of noise: we might suspect from this and the large magnitudes of $\mathbf{A}_{14}$ and $\mathbf{A}_{16}$ that states $\mathbf{f}_4$ and $\mathbf{f}_6$ are too similar. Using our inner product to evaluate an angle between these functions gives

$$\vartheta(\mathbf{f}_4,\mathbf{f}_6)=\cos^{-1}\frac{\mathbf{f}_4\bullet\mathbf{f}_6}{\|\mathbf{f}_4\|\|\mathbf{f}_6\|}\cong 21°$$

This suggests that there may be excessive correlation between these states (which correspond to $V_{C5}$ and $V_{C8}$ in figure 8.1).

We can also look at the sensitivity performance of this filter in keeping with our emphasis on integrators we will plot some functions related to $S_{\gamma}^{t_s}$.

Figure 8.3 shows eight plots, one for each integrator, of the function

$$S_{\gamma_i}(\omega)=\left|\frac{f_{i,s}(j\omega)g_{i,s}(j\omega)}{t_s'(j\omega)}\right|$$

which is proportional to the sensitivity magnitude $|S_{\gamma_i}^{t_s(j\omega)}|$ normalized to the optimum given by (7-2). This measure of integrator sensitivity has the advantage of enabling one to see how close a design is to being FSLB. Because of (7-2) we know that an FSLB filter, if it existed, would have $S_{\gamma_i}(\omega)\equiv\frac{1}{n}=\frac{1}{8}\ \forall\ i,\omega$. Figure 8.3 clearly shows that this particular ladder simulation is nowhere near to having optimum performance: in fact the sensitivities to $\gamma_4$ and $\gamma_6$ (of which we have already been made suspicious by inspection of A, K and W) are between 10 and 15 times worse than this optimum in the passband.

Section 8.5 below will show how this normalized measure relates to more conventional ones for this problem.

**Aggregates**

Inspection of figure 8.3 allows a designer to understand the behaviour and meaning of some of the different aggregate measures of integrator sensitivity discussed in chapter 4, all of which are just means of summarizing all eight $S_{\gamma_i}(\omega)$ in a single curve or number. A fairly obvious approach is to pick the worst case over $i$ at each frequency, which leads to

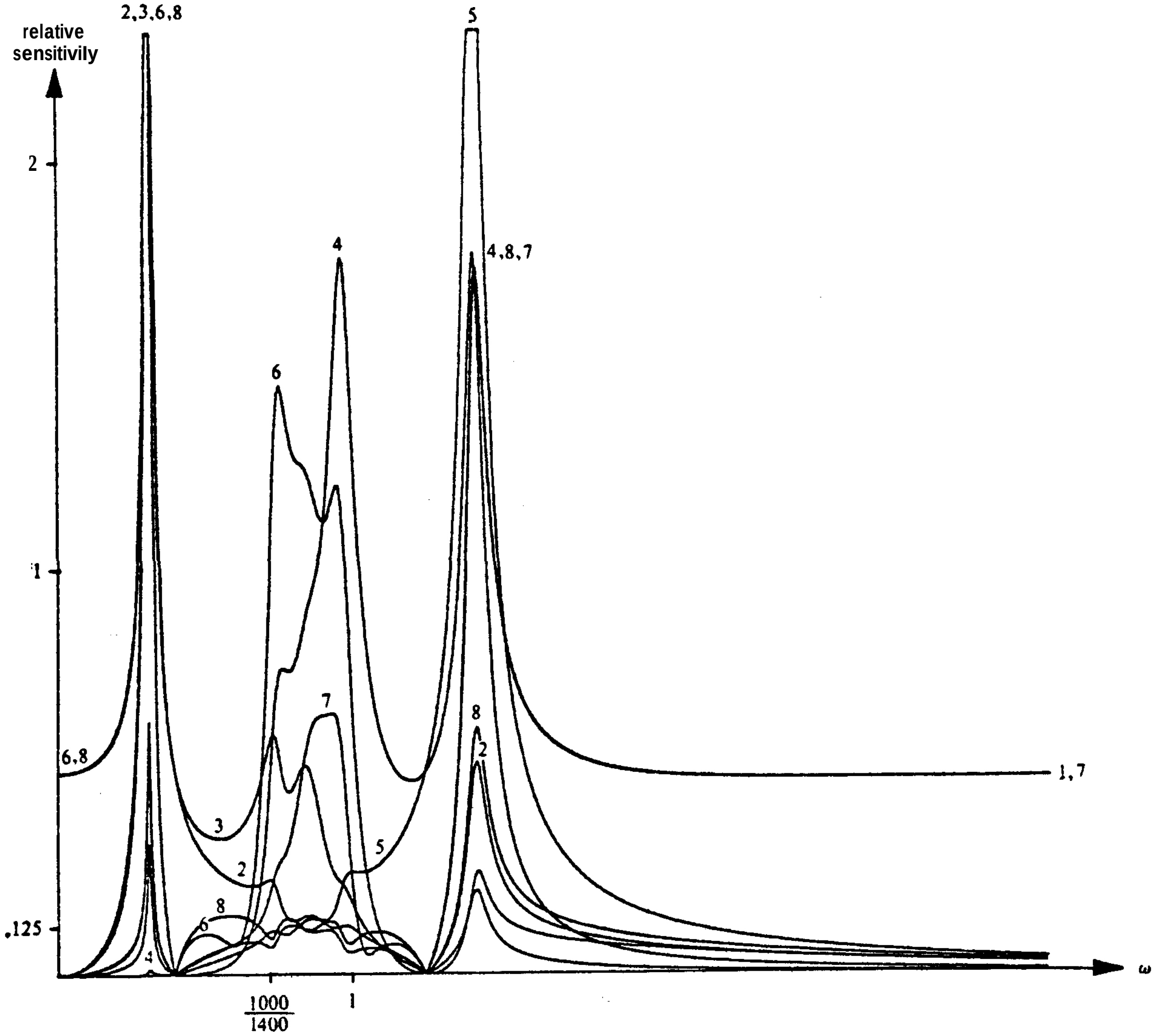


**Figure 8.3: Normalized Sensitivity Plots for First Ladder Simulation**

$$S_{\gamma,\infty}(\omega) \triangleq \max_{i} S_{\gamma_i}(\omega)$$

$$\geq 1/n$$

For simplicity, this is the kind of curve we will plot for all the other designs to be done for this filter.

A statistical aggregate, like

$$S_{\gamma,2}(\omega) \triangleq \sqrt{\sum_i |S_{\gamma_i}(\omega)|^2}$$

$$\geqq 1/n$$

measures the variability of $t_s$ under random perturbations of $\gamma_i$. Its shape will often be similar to that of the $S_{\gamma,\infty}$, since when a single integrator dominates overall sensitivity at each frequency the two measures converge.

An average magnitude measure, like

$$S_{\gamma,1}(\omega) \triangleq \frac{1}{n} \sum_i |S_{\gamma_i}(\omega)|$$

$$\geqq 1/n$$

distinguishes between filters for which the sensitivity functions are different only in magnitude, but highly correlated, and those in which these functions are very different (or opposite in sign}. For an FSLB filter,

$$\mathbf{f}_{i,s}\mathbf{g}_{i,s} = \frac{-1}{n} t'_s \quad \forall\, i$$

i.e. all $S_{\gamma_i}\frac{\mathbf{f}_{i,s}\mathbf{g}_{i,s}}{t_s'}$ are exactly equal, which we believe to be optimum. A design in which the $\mathbf{f}_{i,s}\mathbf{g}_{i,s}$ all lie in the same direction (when seen as a vector in the vector space of possible $\mathbf{f}_{i,s}\mathbf{g}_{i,s}$) has $S_{\gamma,1} = \frac{1}{n}$ and is a reasonable suboptimum. Note, for instance, that figure 8.3 shows that at high frequencies the system has $|S_{\gamma_1}(\omega)| = |S_{\gamma_7}(\omega)| = 1/2$ and all other $S_{\gamma_i} \cong 0$. We can deduce that $\mathbf{f}_{1,s}\mathbf{g}_{1,s} \cong \mathbf{f}_{7,s}\mathbf{g}_{7,s} \cong \frac{-1}{2} t_s'$ at these frequencies: at least there we are not adding large, nearly equal and opposite, terms together to get the invariant sum $\sum_i \mathbf{f}_{i,s}\mathbf{g}_{i,s}$.

This particular ladder simulation is a long way from the optimum $\frac{1}{n}$=.125 for any of these aggregate figures over the passband and most of the stopband.

We conclude that the filter, despite the fact that it is a ladder simulation, is a long way from lower-bound performance in the passband. Further, it looks as if its problem is that $\mathbf{x}_4$ (simulating $V_{C5}$) and $\mathbf{x}_6$ (simulating $V_{C6}$) are too similar.

One does not necessarily get good filters by blindly simulating ladders.

## 8.3 An Improved Ladder Simulation

It seems that the way to improve this simulation would be to replace one or both of the overly-similar $\mathbf{f}_4$ and $\mathbf{f}_6$ with something else (while still obeying the basic rules of section 8.2).

Several of the rows in our first A matrix appeared to be forming weighted sums of $\mathbf{f}_4, \mathbf{f}_6$, and $\mathbf{f}_7$. This suggests that they might be trying to form $V_{C5}+V_{C6}+V_{C7}=-V_{C4}$ by summation, which is one of the states that we threw out in order to get a canonic system. If we therefore replace the troublesome $\mathbf{f}_6(V_{C6})$ with $V_{C4}$ we might expect a better filter.

The result is somewhat better: it has total output noise (after $L_2$ scaling) of $\sum_i \mathbf{W}_{ii} \cong 200$, which is about 7dB better than our first design. The largest element of *A in this* design is $\mathbf{A}_{16}=2.62$, and the next *worst* is $\mathbf{A}_{13}=2.4$. Since these are both greater than 1, we may suspect (cf. section 4.11) that now our new $\mathbf{f}_3$ and $\mathbf{f}_6$ are too similar: in fact the angle between them is only 18°.* States 3 and 6 also contribute most of the total noise:

- This is even worse than before, but the filter is a little better because the worst states are now slightly further away from the rest of the states.

$$\mathbf{W}_{ii}=[7.78\ \ 9.76\ \ 66.24\ \ 15.378\ \ 9.76\ \ 74.84\ \ 8.63\ \ 7.8]$$

Rather than continuing on and looking at sensitivity, Iet us simply replace $\mathbf{f}_3$ with something more useful.

## 8-4 A Good Ladder Simulation

Choosing to have $\mathbf{f}_3$ simulate $V_{C2}$ rather than $V_{C3}$ gives a system with

$$\mathbf{W}_{ii}=[7.78\ \ 9.76\ \ 7.83\ \ 10.75\ \ 9.76\ \ 10.86\ \ 8.17\ \ 7.8]$$

and $\sum_i \mathbf{W}_{ii} \cong 72.7$. This is 5dB better than our second design, and in fact the technique of [Mull'761 states that the best possible output noise level for this problem is about 67.4 (cf. section 8.6 below) : this ladder simulation is within 0.4dB of theoretically optimum dynamic range!

The system description required to get this set of $\{\mathbf{f}_i\}$ *is:*

$$A=\begin{vmatrix} -0.1816 & 0 & -0.8288 & 0 & 0 & 0.1801 & 0 & 0 \\ 0 & 0 & 0.0695 & -0.0455 & 0 & 0.8533 & -0.0164 & 0 \\ 0.8445 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.0615 & -0.1675 & 0 & 0 & -0.9235 & 0 & 0.0313 & 0.1459 \\ 0 & 0 & 0 & 0.9005 & 0 & 0 & 0 & 0 \\ -0.3023 & -0.8235 & 0 & 0 & -0.1893 & 0 & -0.00744 & -0.0345 \\ -0.0199 & -0.0542 & 0 & 0 & 0.2509 & 0 & -0.1816 & -0.8454 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.3426 & 0 \end{vmatrix} \quad b=\begin{vmatrix} -0.2814 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{vmatrix}$$

$$\mathbf{c}^T=\left[0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0.7846\ \ 0\right] \qquad d=0$$

now all $\mathbf{A}_{ij}$ are less than 1 in magnitude, which we believe to be good.

Figure 8.4 plots $\sum_i |\mathbf{g}_i(j\omega)|^2$ for this ladder, and may be interpreted to mean that passband output noise for this filter will be about 4OdB above the level of input noise. Thus if we used integrators with $12nV/\sqrt{Hz}$ of input noise to

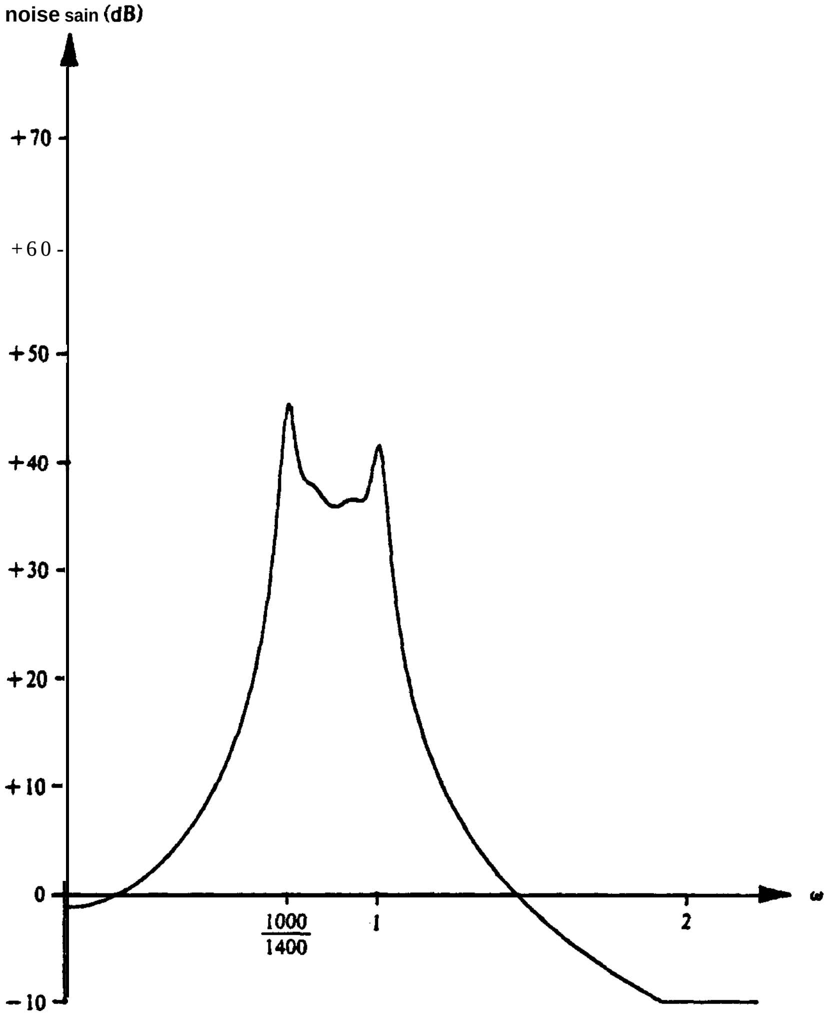


**Figure 8.4: Total Output Noise for Good Ladder Simulation**

implement this filter we could expect about $1.2\mu v/\sqrt{Hz}$ at the output.

Figure 8.5 plots $S_{\gamma,\infty}(\omega)$ for this filter. Notice that it is near minimum (1/8) over the passband and that it is $\cong 1/2$ over much of the stopband. We may interpret these features as meaning that the function is approximately equally sensitive to each of its integrators in the passband and usually dominated by a pair of integrators in the stopband (because a good second-order system would have $S_{\gamma,\infty}$=0.5, which is what we have over most of the stopband).

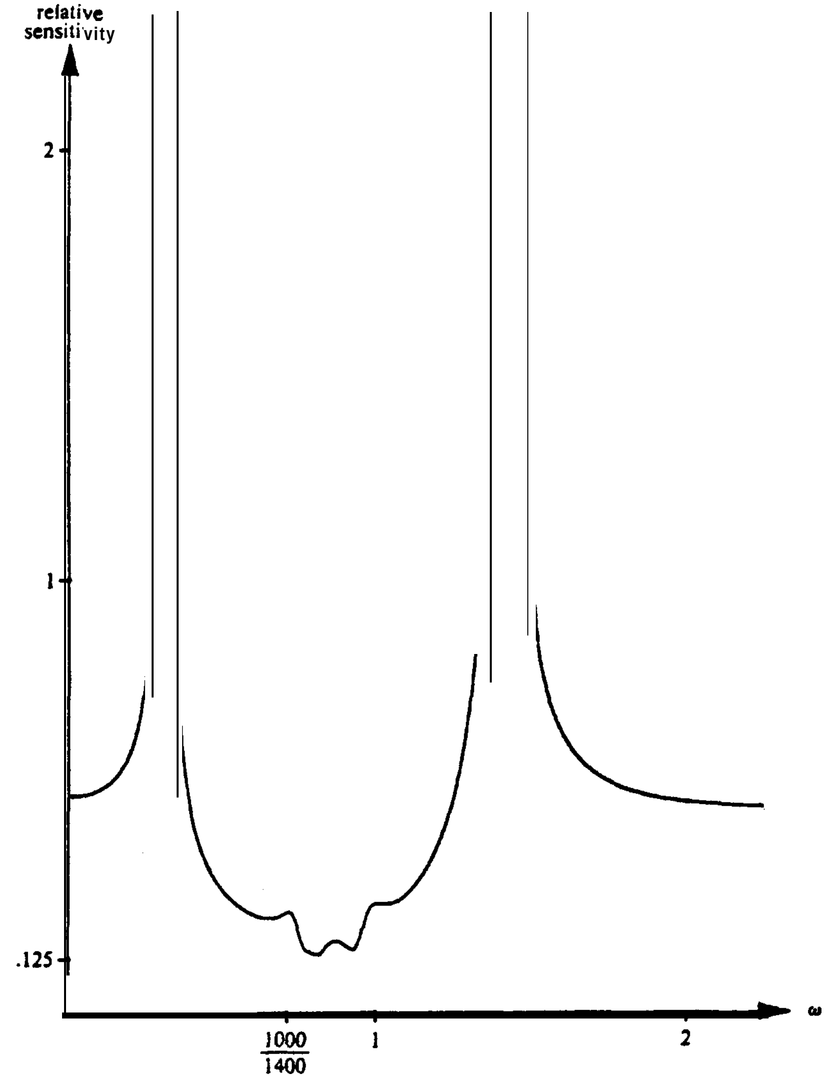


**Figure 8.5 Worst-case Integrator Sensitivity for Good Ladder Simulation**

There are, however, high peaks (which actually go off this graph): one at $\omega \cong .314$ where $S_{\gamma,\infty} \cong 5.7$ and the upper-stopband one at $\omega \cong 1.43$ where $S_{\gamma,\infty} = 2.74$. These sensitivity maxima turn out to occur at local ***minima of*** the attenuation function for the filter, i.e. where there is the smallest amount of stopband margin to protect the design from breaking specifications. Thus it seems that ladder simulations reserve their worst performance for the most sensitive part of the stopband, an unfortunate problem that we shall shortly see that cascade designs share.

It is a feature of the type of sensitivity plot that we have chosen to use, one normalized to the theoretical lower bound on performance, that sensitivity peaks appear at stopband attenuation minima for these filters. The classical sensitivity figure $S_{\gamma_i}^{t_s}$ does not peak at these points, but the lower bound dips (because the derivative $t_s'$ is small at a stopband minimum). It would be wrong, however, to treat the peaks as an artifice of the measure because there are some structures which do not exhibit them, and are therefore much better than ladder and cascade structures at the most critical stopband frequencies.

**Structure of the Simulation**

**The** system implementing the $\{\mathbf{f}_i\}$ for our good ladder simulation has several desirable properties: we have already seen that it has near-minimum noise and passband integrator sensitivity and that no large coefficients appear in **A** (i.e. no integrators have very short time-constants). It also does not need a zero-forming network (corresponding to the $\mathbf{c}^T$ vector) because only one element of $\mathbf{c}^T$ is non-zero, so that we may simply take the system output as state $\mathbf{x}_7$; this useful saving happened because we chose $\mathbf{f}_7 = t_s$. It also happens that of the **b** vector only $\mathbf{b}_1$ is non-zero, which makes the input easy to apply; this happened because only $\mathbf{f}_1$ was of order $n-1=7$. This kind of saving is usually easy to arrange when choosing $\{\mathbf{f}_i\}$.

The system simulating the ladder is fairly sparse, with only 27 of the 81 elements non-zero. This comes about because of the structure of the original ladder, and in particular because one element of each ‘Yank” (LC resonator) simply integrates the output of the other (if a capacitor and an inductor are in parallel, the inductor current is the integral of capacitor voltage), and therefore only has a single entry in its row (i.e. does not have to perform a summation as well as integration). The functions $\mathbf{f}_3, \mathbf{f}_5$ and $\mathbf{f}_8$ in this structure may be seen to be derived in this way from $\mathbf{f}_1, \mathbf{f}_4$ and $\mathbf{f}_7$, respectively. In general it is fairly safe to make about half the $\{\mathbf{f}_i\}$ simply integrals of the other half, because $\mathbf{f}_i$ and its integral are orthogonal (under any inner product of the same form as the one we chose for this example). The weakness of companion form will be shown to result from the fact that it goes too far and integrates a single function n -1 times, and that the result of integrating an $\mathbf{f}_i$ twice is very strongly correlated with the original $\mathbf{f}_i$, especially in a narrow-band case.

**Summary**

We have demonstrated a canonic simulation of an LC ladder with near-optimum performance in the passband; we have also shown that it is fairly easy to diagnose and correct structural faults in filter designs by means of the vector-space approach offered by intermediate-function synthesis.

## 8.5 Sensitivities to Magnitude and Phase

This chapter has been (and will continue) comparing filters on the basis of the magnitude of a complex-valued sensitivity measure normalized to the lower bound on performance (which we called the "frequency scaling lower bound") discussed in chapter 7.

Filter designers are generally accustomed to investigating $S_{|x|}^{|t_s|} = \frac{\partial \ln |t_s(j\omega)|}{\partial \ln |x|}$ and $\frac{\partial \ln |t_s(j\omega)|}{\partial \varphi(x)}$, the sensitivity of filter magnitude response to magnitude and phase errors in transmittances.* To illustrate the relationship between this approach and the one we take we show in figure 6.6 three functions of $S_{\gamma_1}^{t_s}$, which is the (complex) sensitivity of $t_s$ to an integrator's gain, for our best ladder design. The magnitude $|S_{\gamma_1}^{t_s}|$ is shown, and is the kind of function our measure $S_{\gamma_1}(\omega)$ investigates, except that $S_\gamma$ normalizes this sensitivity to its frequency-scaling lower bound; The real part $\mathrm{Re}(S_{\gamma_1}^{t_s})$ is shown, which may be

* It has been customary to deal with phase errors in terms of sensitivity to component "Q-factors" or "dissipation factors", but it is easy enough to relate this older work to phase, which is naturally related to the logarithmic definition usually used for sensitivity.

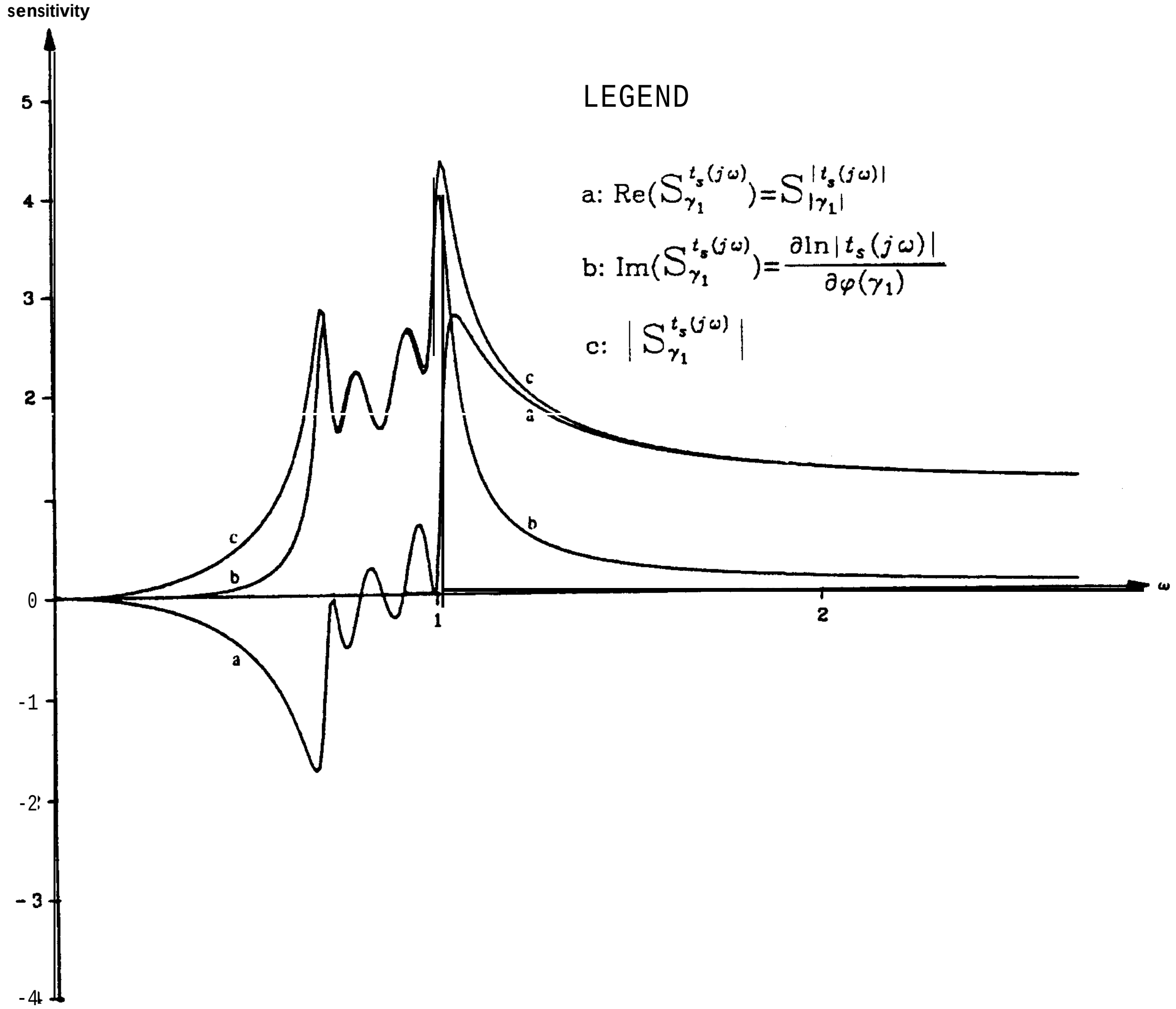


**Figure 8.6: Classical Sensitivity for Ladder Simulation**

interpreted (cf. section 4.3) as $S^{|t_s|}_{|\gamma_1|}$; and the imaginary part $\text{Im}(S^{t_s}_{\gamma_1})$ is shown, which is $\frac{\partial|t_s(j\omega)|}{\partial\varphi(\gamma_1)}$ (equation 4-9).

Note that $S^{|t_s|}_{|\gamma_1|}$ goes to zero at the four reflection zeros, just as the sensitivity of a doubly terminated ladder to a reactive element must, and that it is also zero at another passband frequency (a local maximum of $|t_s|$). Note also that throughout the passband sensitivity to phase is relatively large.

The large magnitude of $\frac{\partial \ln|t_s|}{\partial\varphi(\gamma_1)}$ relative to that of $S^{|t_s|}_{|\gamma_1|}$ appears at first to suggest that the argument for ladder structures might contain a fallacy: if filters existed with lower $\frac{\partial \ln|t_s|}{\partial\varphi(\gamma_i)}$ they might be preferable to these even if $S^{|t_s|}_{|\gamma_i|}$ were somewhat higher, since a practical ladder simulation looks likely to have its sensitivity performance dominated by its phase sensitivity rather than by its magnitude sensitivity. We can see from the normalized measure $S_{\gamma_1}$, however, that phase sensitivity cannot be improved very much: an FSLB filter has the lowest possible sensitivity to phase (because equalizing the sensitivities a *fortiori* equalizes sensitivity to phase) and this filter is close to FSLB in the passband.

Our earlier figure 8.5 showed that this ladder was close to FSLB (cf. chapter '7') in the passband, and the argument of chapter 7' is that this necessarily implies features like low magnitude sensitivity in the passband, and that the phase-sensitivity performance of almost-FSLB filters is almost as good as is possible. The filter designer should therefore expect to see curves Iike those of figure 8.6 in any “good” filter, i.e. in any filter for which $S_{\gamma_i} \cong \frac{1}{n}$.

It appears that the justification for thinking that this ladder simulation is good is more clearly deducible from our $S_\gamma$ measures than from the more conventional magnitude and phase sensitivities.

## *8.6* Noise Bound

The derivation in [12] for optimum $\sum_i \mathbf{K}_{ii}\mathbf{W}_{ii}$ (i.e. minimum noise for a scaled filter) relied on the invariance of the eigenvalues of KW (cf. section 4.8.1). We can take KW for any of our filters and find these eigenvalues (or “principal values” or “second order modes”); they are

$$\mu_i^2 = [0.1605\ 0.17029\ 2.6562\ 2.6945\ 15.125\ 15.279\ 32.232\ 32.0591$$

From these we can use the formulae in [12] to find out that

$$\sum_i \mathbf{K}_{ii}\mathbf{W}_{ii} \geq \frac{1}{n}\left(\sum_i \mu_i\right)^2 = 67.4$$

Our best ladder filter $(\sum_i \mathbf{K}_{ii}\mathbf{W}_{ii} = 72.7)$ clearly came quite close.

## 8.7 A Cascade Design

The next popular type of design to try, in order to demonstrate IF synthesis, is a cascade type.

A cascade contains sections with individual transfer functions $\Gamma_1, \Gamma_2,$ etc., connected one after another so that $t_s = \prod_j \Gamma_j$. In principle the structure of each section could be arbitrary, but for this example we will assume that a two-integrator loop is to be used with zeros to be formed by summing the two outputs, either at a dedicated summer or (more economically) directly at the inputs to integrators in later stages. The manner of summation does not affect sensitivity to integrators (our dominant concern).

The $\{\mathbf{f}_i\}$ appearing in the $k^{th}$ section are of the form

$$\mathbf{f}_i = \left[\prod_{j<k} \Gamma_j\right] \frac{1}{e_{k,s}} \tag{8-1}$$

$$\mathbf{f}_{i+1} = \left[\prod_{j<k} \Gamma_j\right] \frac{s}{e_{k,s}}$$

where $e_{k,s}$ is the denominator of the $k^{th}$ section, which (if it is second order} contains two integrators. A first-order $e_k$ only needs the first of these two $\{\mathbf{f}_i\}$. Note that in the second-order case one of the two intermediate functions is just the integral of the other, so that half of the system's integrators clearly need only one input: we suggested at the end of section 8.4 that this was usually safe enough.

Figure 8.7 is a sketch of the s-plane, and shows a particular choice of factors $\Gamma_j$ for $t_s$, i.e. a “pairing and ordering” [8,13]: each singularity is labelled with the index number of the section ($\Gamma$) which realizes it.

This design, following the rule of thumb developed in [8,10,13] has natural modes paired with their nearest transmission zeros, with high-Q natural modes taking priority, and orders the sections to have successively higher Q-factors from start to finish so that the front-end of the filter is relatively low-Q. Lowpass and highpass sections are alternated. Choosing $\{f_i\}$ this way and $L_2$ scaling as usual gives a system with:

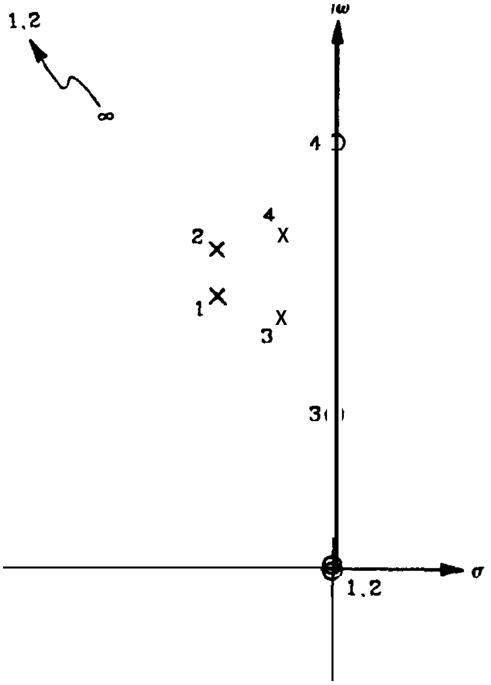


**Figure** 8.7: A Pairing **and Orderin**

$$A=\begin{vmatrix} 0 & 0.7911 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.7911 & -0.1278 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.8553 & 0 & 0 & 0 & 0 \\ 0 & 0.2039 & -0.9871 & -0.1363 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.7683 & 0 & 0 \\ 0 & 0 & 0 & 0.1794 & -0.6572 & -0.046\% & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.9036 \\ 0 & 0 & 0 & 0.0669 & -0.222 & -0.0231 & -1.117 & -0.0525 \end{vmatrix} \quad b=\begin{vmatrix} 0 \\ 0.2016 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{vmatrix}$$

$c^T$=[ 0 0 0 0.0825 -0.2066 -0.0214 0.5681 -0.0488 ] d=0

$W_{ii}$=[ 13.35 15.15 16.18 13.5 15.32 16.42 19.71 15.74 ]

$$\sum_i W_{ii} = 125.5$$

This system is clearly quite sparse, as we would expect of a cascade, and has about 3dB more noise than the optimum.

### Partial Sums

An interesting observation to make about this filter concerns zero formation. A circuit like a KHN biquad [7] or a Tow-Thomas biquad[14] using output summing forms notches by summing its internal states (and, possibly, the input), and then passes a single result on to the next stage. If interpreted as suggested in chapter 3, the "cascade" structure we have here delays that summation until it is needed, i.e. does summations at the inputs of later integrators. One is free, of course, to implement these zero-forming networks either way: using an op-amp to form sums that a Miller integrator's virtual ground could equally well have formed is wasteful; but if an &term sum will be needed by several states, it probably saves net cost to form it only once (at a cost of 8 resistors and one op-amp) and then distribute it (at a cost of one resistor each) to all of the integrators needing it.

A version of this procedure, related to feedback structure rather than to zero formation, is illustrated by the SFG of figure 8.8. Inspection of the cascade system above shows that row 8 of A is directly proportional to $c^T$ except at element 7, from whence can be derived the equivalence of figure 8.8. This unusual structure derives an internal feedback from the overall system output, and avoids using a 5-term summation to form $\dot{x}_8$ Notice, however, that this saving is obtained at the cost of a cancellation in forming $A_{87}$: there are two paths from the output of integrator 7 to the input of integrator 8, one with a gain of 0.61 through $y$ and a direct path with a gain of -1.73. The result is an increase in the sensitivity of $A_{87}$ to elements of the second SFG of figure 8.8 in return for a reduced number of circuit elements.

Generally, within a given state-space structure there will often be a number of different ways to form the summations implied by the $\{A, b, c, d\}$ system. It

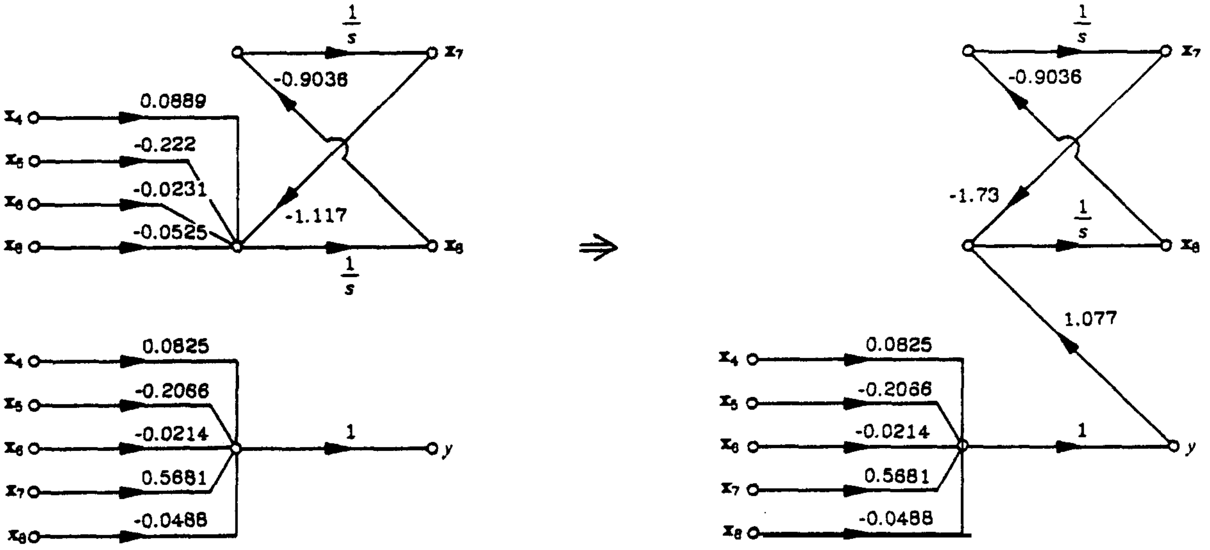


**Figure 8.8: Re-using a Sum**

may be advantageous to form and re-use partial sums when several rows of the system equations contain several common terms. We regard this as a design question to be settled after the deeper, state-space, structure has been chosen.

## 8.8 A Cascade Design with Higher Dynamic Range

The rule of thumb from the literature [10] used above to determine section ordering lacks symmetry with respect to input and output. The dual rules of $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$, and of the signal and noise gains investigated in [9], suggest that if a particular design is good then its "dual" must also be. By "dual" we mean here a new system derived by interchanging the $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ of a system, which interchanges inputs and outputs in a state-space structure.

We therefore propose a different rule: that the highest-Q sections of a cascade filter be in the middle of the structure and the low-Q ones on the outside. This rule is symmetric with respect to a reversal of the order of sections in a filter while the previous rule is not. This symmetry suggests, as does the close duality of $\{f_i\}$ and $\{g_i\}$, that if any particular structure is good then its reverse will also be. We still maintain the usual “pairing” rule and associate poles with their nearest zeros, and we still alternate highpass and lowpass sections.

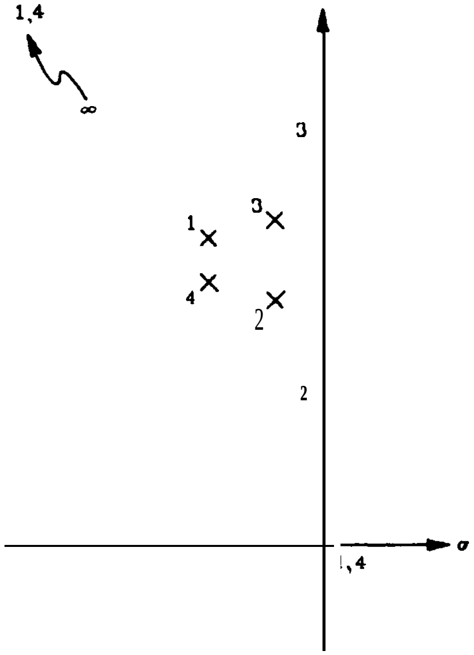
This modified ordering rule is reminiscent of the rule for ordering of “removals” in an LC synthesis given in [10] which places tanks (LC resonators) that form notches close to bandedges near the middle of the filter and those forming notches at extreme frequencies on the outside. The symmetry here is obviously implied by the reciprocal nature of a ladder synthesis, because interchanging input and output in a reciprocal network makes no difference to sensitivity.

In order to experimentally investigate the effect of this more appealing section ordering we will look at the ordering shown in figure 8.9.

With $\{f_i\}$ selected in the same general way as before we obtain a scaled system with

$$W_{ii}=[\ 16.89\ 14.2\ 12.78\ 13.57\ 11\text{-}82\ 11.9\ 13.35\ 15.15\ ]$$

for which the total output noise figure is $\sum_i W_{ii} \cong 109.7$. This system clearly has significantly better dynamic range than does its predecessor.



F’igure 8.9: Improved Ordering

Figure 8.10 shows the expected output noise spectrum $\sum_i |g_i|^2$ for this design, which can be seen to be slightly "peakier" than that of our best ladder, with a maximum value of about 5OdB.

Figure 8.11 shows sensitivity $S_{\gamma,\infty}(\omega)$ for this cascade design. Two important features stand out: that $S_{\gamma,\infty} \cong 1/2$ over the passband and most of the stopband, but that the sensitivity peaks (just as it did for the ladder) at about the

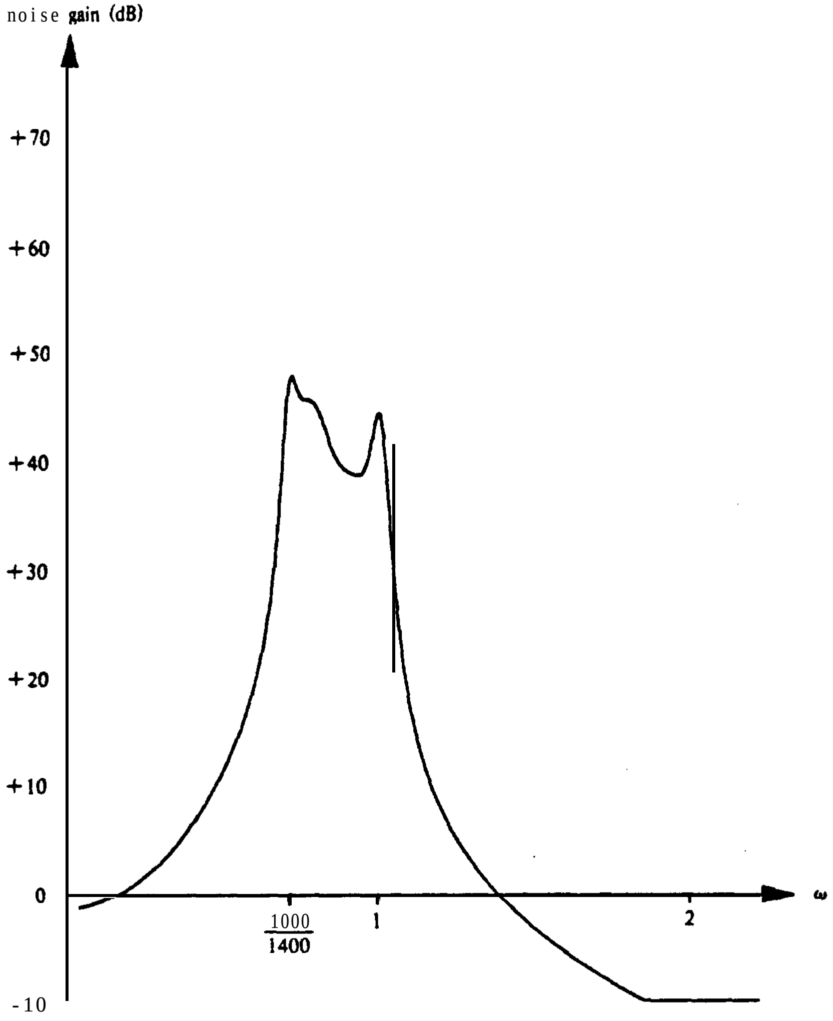


**Figure 8.10: Output Noise for Improved Cascade**

stopband minima.

## 8.9 Another Cascade Design

It happens that there is a cascade design with marginally better dynamic range yet: namely one with the pairing and ordering shown in figure 8.12, for which (after scaling} $\sum_i W_{ii} \cong 103.4$.

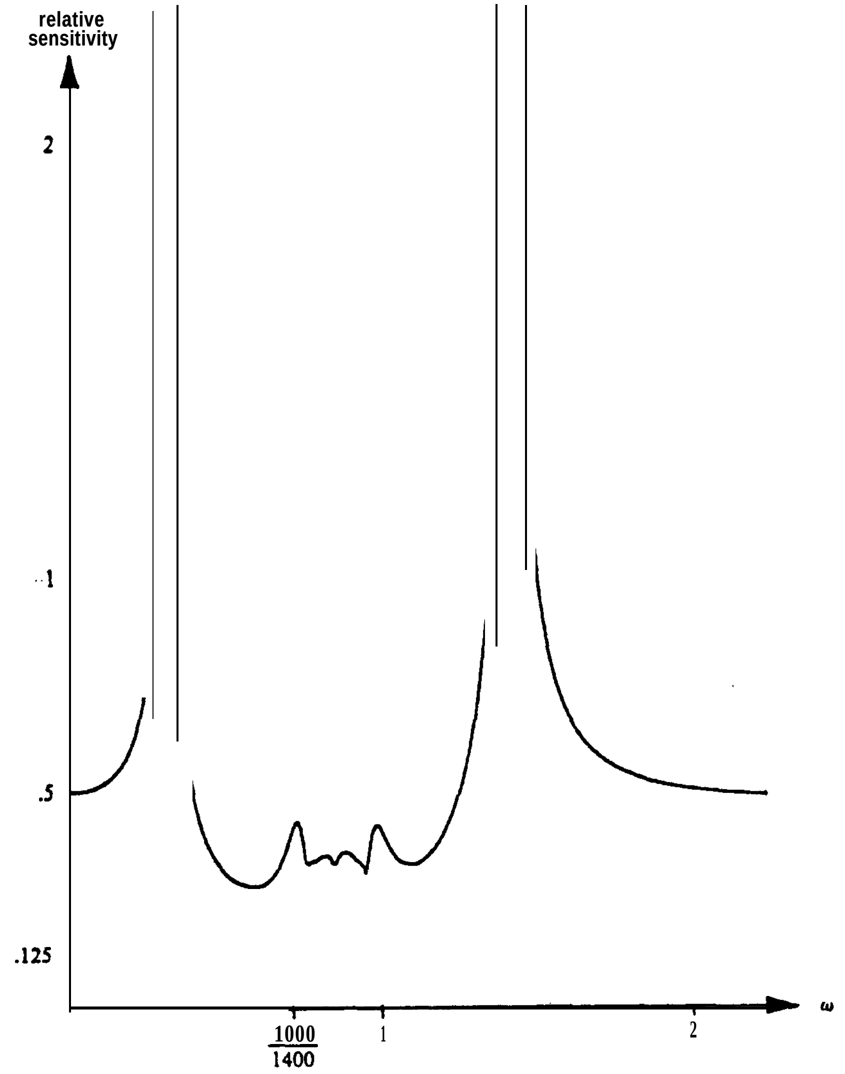


**Figure 8.11: Normalized Sensitivity for Improved Cascade**

This differs from our previous design only in that the low-Q poles are here assigned to low-pass and high-pass sections where the previous design assigned them both to bandpass sections. This (better) design adheres more closely to the pairing rule recommended in section 8.8 above.

The noise spectrum for this design is shown in figure 8.13 and $S_{\gamma,\infty}$ appears in figure 8.14. The most notable thing about these curves is that they are very similar to those for the previous cascade design: the designs are obviously pretty similar.

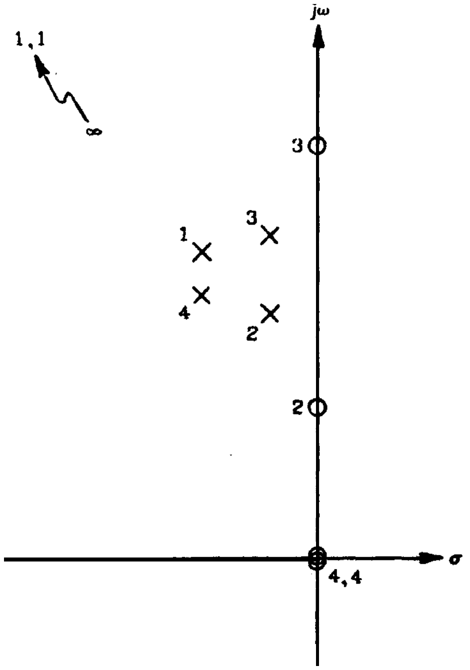


**Figure 8.12: Best Pairing and Ordering**

## 8.10 Summary: Cascade

Three cascade designs have been touched on at different levels of detail. A "duality" condition suggested an alteration in the manner of section ordering, and the resulting designs indeed outperformed a conventional design. Zero formation within the cascade can be done in different ways within a basic structure, with some straightforward trade-offs between component count and sensitivity.

The passband sensitivity figure $S_{\gamma,\infty}$ of the cascade designs was about 1/2, whereas the ladder circuits approached the limiting $S_{\gamma,\infty}=1/n$. This just means that in a cascade design, at any given passband frequency, two integrators (i.e. one section) dominate performance while in a ladder all n participate about equally. It therefore seems that (in the passband) cascade design suffer from a disadvantage of a factor $\frac{n}{2}$ relative to coupled designs.

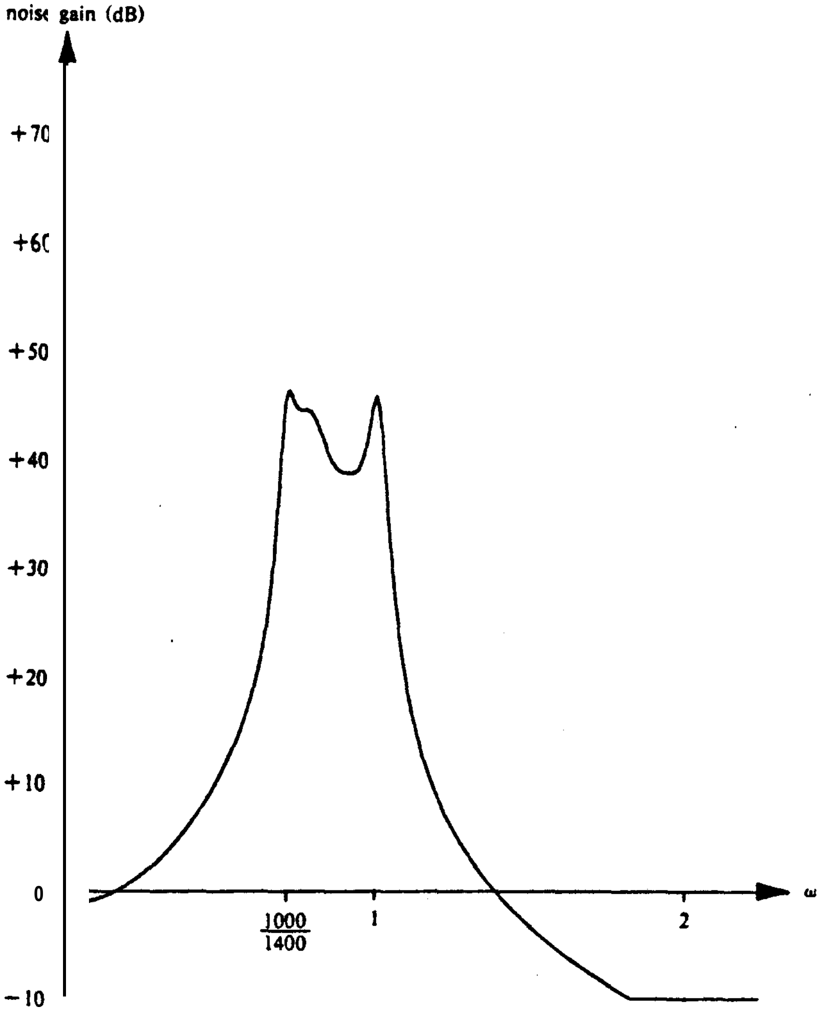


**Figure 8.13: Noise for Best Cascade**

Stopband sensitivities for cascade and ladder structures were generally similar: over much of the stopband $S_{\gamma,\infty} \cong 1/2$, reflecting the fact that stopband performance at these frequencies is determined by 2 integrators. The fact that $S_{\gamma,\infty}=1/2$ at transmission zeros may be directly attributed to the “decoupled tuning” property for which cascade and ladder designs are noted in their stopbands: only a single resonator (i.e. 2 integrators) determines $t_s$ at these points. It follows that no structure with decoupled tuning can possibly do better than a cascade at transmission zeros.

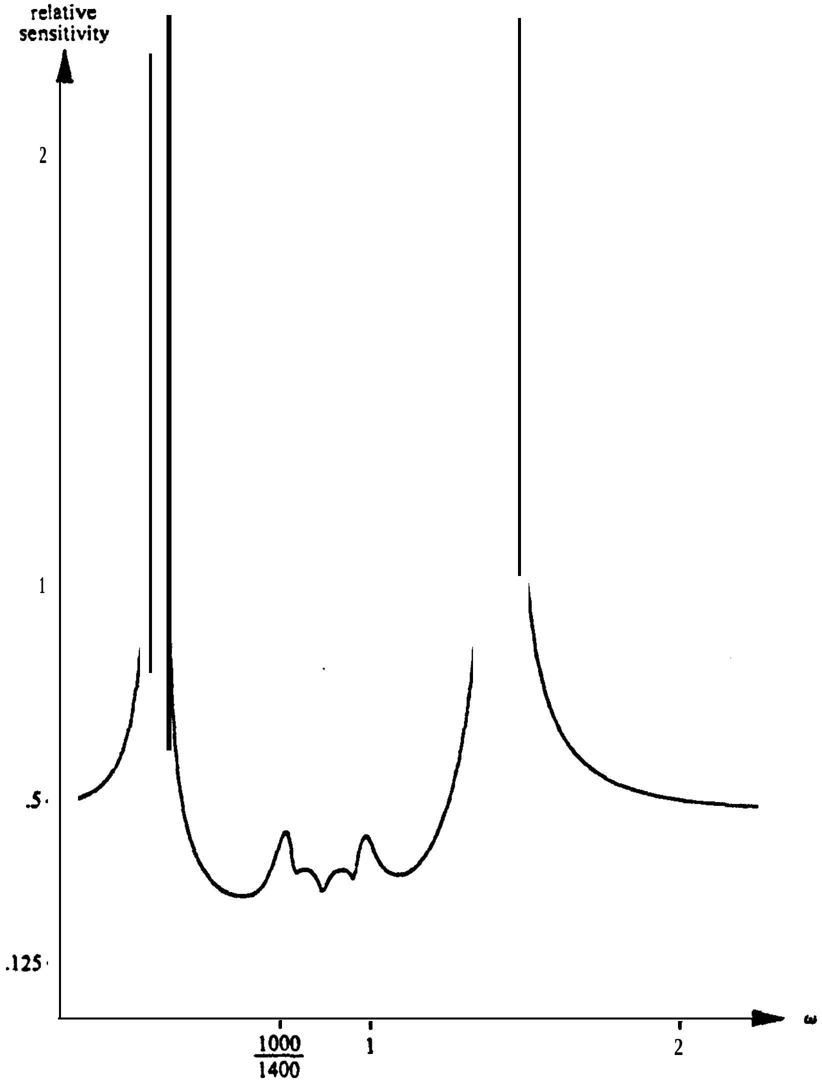


Figure 8.14: Sensitivity for Best Cascade

**The worst** stopband feature of these designs, however, is that they have stopband sensitivity peaks at stopband minima.

## 8.11 Decoupled Tuning and Stopband Sensitivity

Both cascade and ladder filters are structured so that only a single pair of integrators is involved in setting the frequency of each transmission zero. It is easy to demonstrate that this decoupling of transmission zeros necessarily implies poor performance at stopband minima.

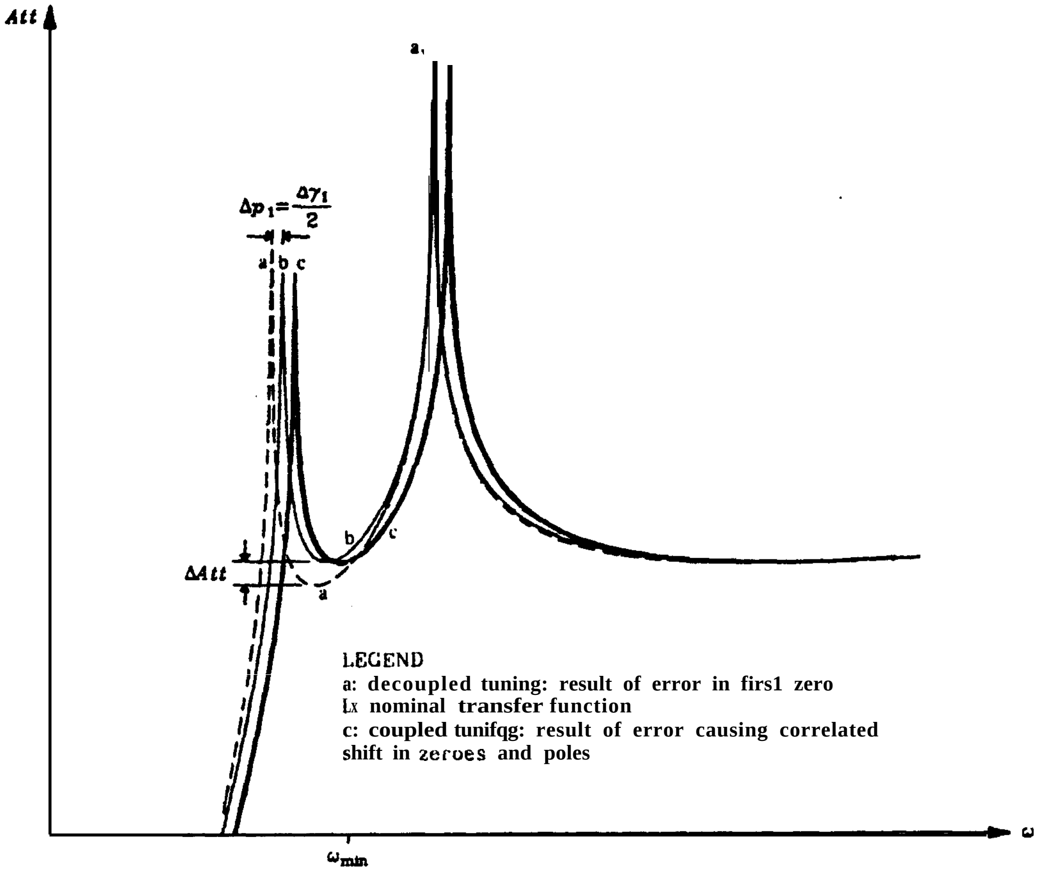


**Figure 8.15: Decoupled Tuning and Stopband Sensitivity**

Figure 8.15 shows the effects for two different filter structures of varying an integrator gain: in a structure with decoupled tuning only one transmission zero moves while in a coupled structure both move together. The difference between the two structures is most evident at the frequency of minimum stop-band attenuation: in the coupled structure there is no effect on attenuation, while in the decoupled structure (e.g. cascade or ladder) there is a large effect..

## 8.12 Companion Form

For a companion-form filter, figure 8.16,

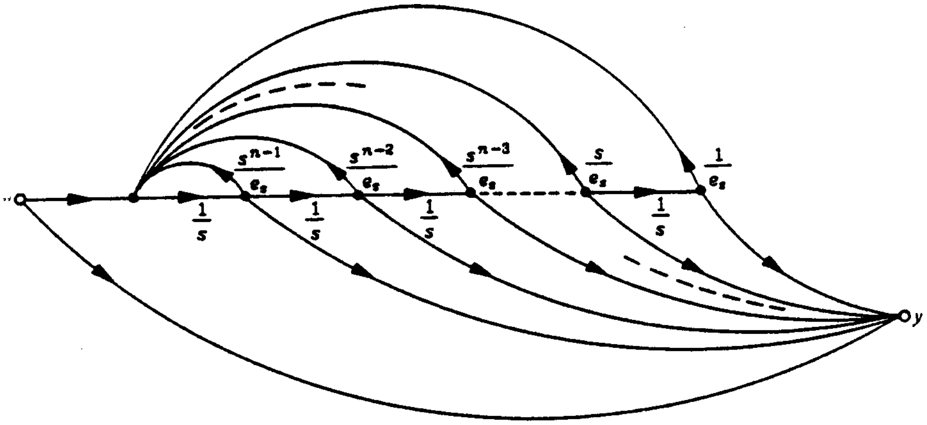


**Figure 8.16: Companion-form Filter**

the $\{f_i\}$ are just $\{\frac{s^{i-1}}{e_s}\}$. We can $L_2$ scale these and look at $\sum_i |g_i|^2$ (figure 8.17) and $S_{y,\infty}(\omega)$ (figure 8.18). Notice that these figures are very much higher than those for any of the filters we have seen so far. Noise gain is about 120dB, 80dB worse than that for the ladders, while the sensitivity figure is around 40 in the passband, over 300 times worse than its optimum. At no frequency does this structure compete with either ladder or cascade.

The reason for this poor performance is not hard to find: some of the $\{f_i\}$ are very close to others, and so we have "near-dependencies". In fact, $K_{13} \cong K_{24} \cong K_{i,i+2} \cong 0.98$, which means that every $f_i$ is 98% correlated with $f_{i+2}$ and $f_{i-2}$, or separated by an angle of only about 13° from its second derivative. This

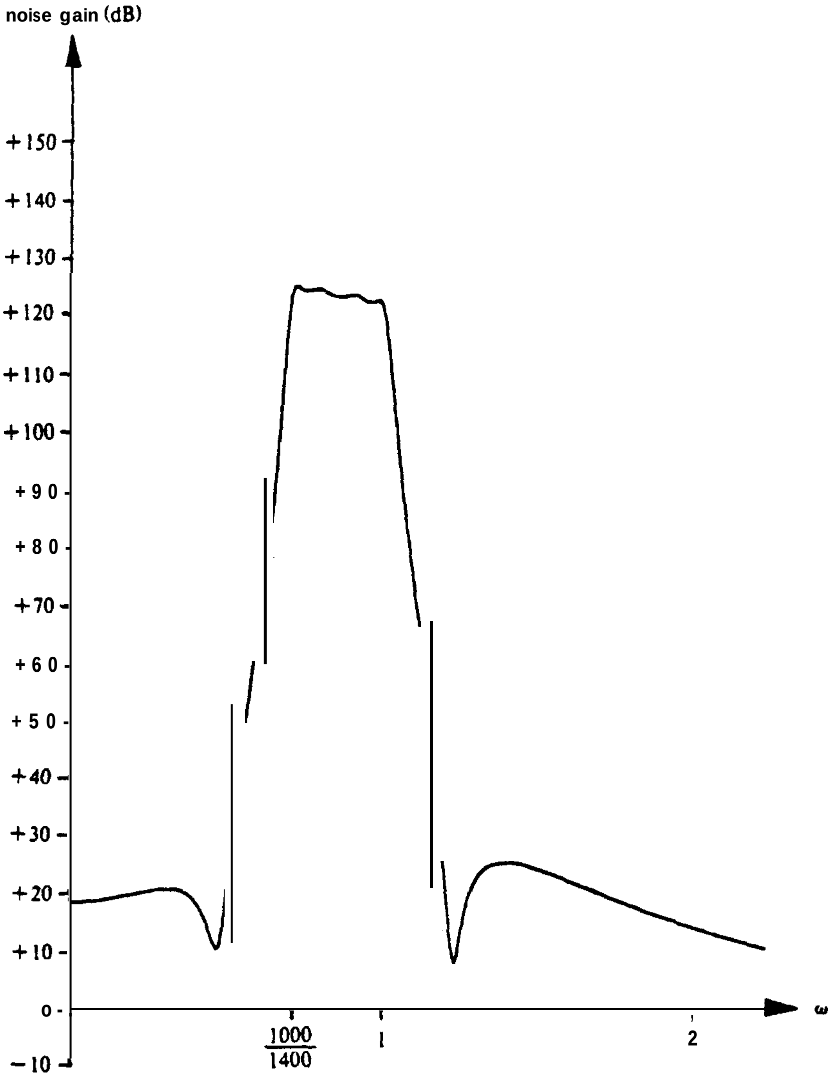


**Figure 8.17: Noise for Companion Form**

confirms our earlier comment that it was usually unsafe to design systems that simply integrate any function two or more times. The correlation becomes progressively stronger as bandwidth is decreased.

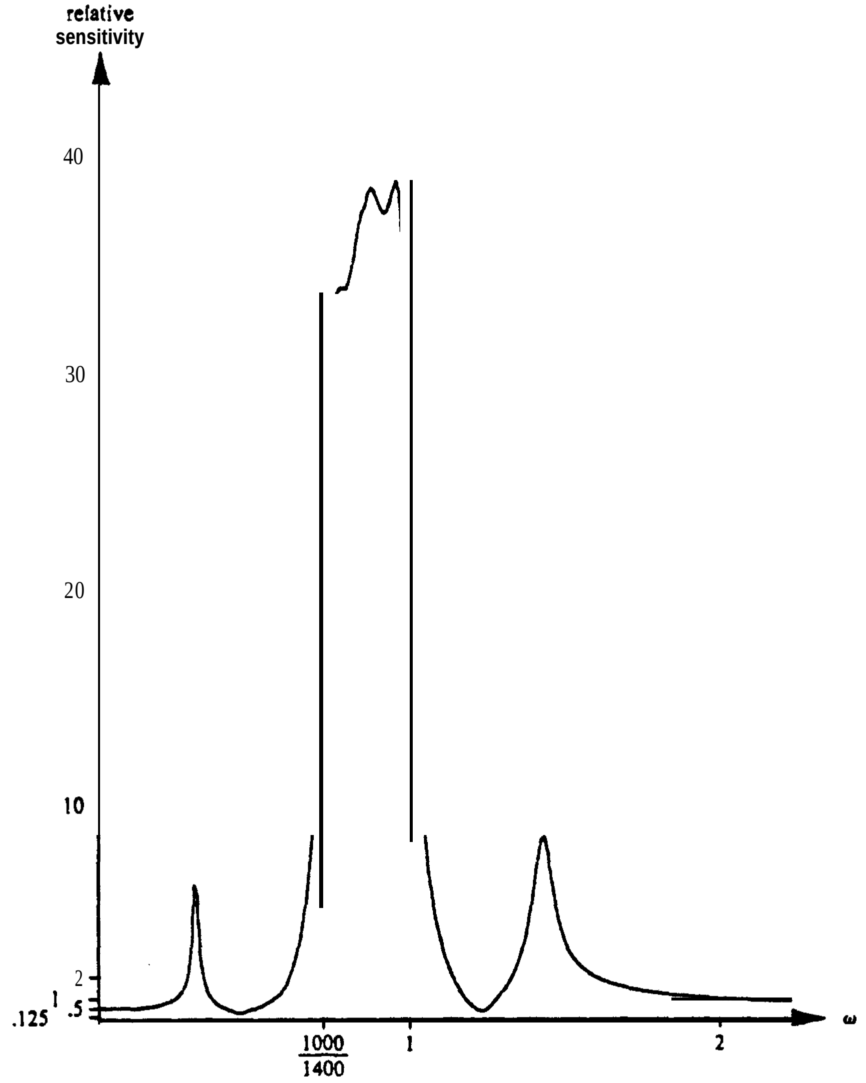


Figure 8.18: Sensitivity for Companion Form

## 8.13 &am-Schmidt Orthonormalization

Chapter 5 discussed "orthonormal" filters, which are free of the problem of near-dependencies among $\{\mathbf{f}_i\}$. Gram-Schmidt orthonormalization [11], of which equations (5-2) and (5-3) of chapter 5 were a case, is an algorithm that produces n orthonormal vectors $\{\tilde{\mathbf{f}}_i\}$ from any n independent $\{\mathbf{f}_i\}$. The procedure modifies each vector in turn by subtracting from it its projection on vectors already treated, then normalizing it.

$$\forall i \quad \{ \quad \tilde{\mathbf{f}}_i \leftarrow \mathbf{f}_i - \sum_{j<i} (\mathbf{f}_i \cdot \tilde{\mathbf{f}}_j) \tilde{\mathbf{f}}_j$$

$$\tilde{\mathbf{f}}_i \leftarrow \frac{\tilde{\mathbf{f}}_i}{\|\tilde{\mathbf{f}}_i\|} \quad \}$$

Since the $\{\mathbf{f}_i\}$ for companion form are independent and we have an inner product, we can use the Gram-Schmidt procedure to find a new system with orthonormal intermediate functions from the (extremely poor) set for companion form. The resulting functions are shown in table 8.3.

| function | leading<br>coefficient | list of<br>roots |
|---|---|---|
| $\mathbf{f}_1$ | 0.00165 | - |
| $\mathbf{f}_2$ | 0.002065 | 0 |
| $\mathbf{f}_3$ | 0.01105 | ±j0.8033 |
| $\mathbf{f}_4$ | 0.01277 | 0, Aj.82457 |
| $\mathbf{f}_5$ | 0.0877 | *j0.7414, j0.93742 |
| $\mathbf{f}_6$ | 0.08044 | 0, ±j0.75018,±j0.94929 |
| $\mathbf{f}_7$ | 0.30171 | ±j.7126,±j.8335,±j.9848 |
| $\mathbf{f}_8$ | 0.34 | 0,±j0.7225,±j0.8561,±j0.9943 |

**Table 8.3: Orthonormal Intermediate Functions**

It is interesting to note that these $\{\mathbf{f}_i\}$ have notches in the filter's passband. Since sensitivities are proportional to $\{\mathbf{f}_i\}$ (chapter 4) it follows that various sensitivities go to zero in the passband. Note that both magnitude and phase sensitivity are thus forced to zero, where only magnitude sensitivity is zero in a ladder or FSLB structure. The net effect is, however, not as good as this would suggest because not all sensitivities are forced to zero simultaneously and those that are not zero become larger than they would be for an FSLB structure.

The total resulting noise is $\sum_i W_{ii}=100$, slightly better than the cascade design and within about 2dB of the optimum: recall (section 5.6) that we have shown that orthonormal filters are at most a factor of n worse than optimum. The structure of the resulting system (Table 8.4} is sparse and suggestive: notice that this is a type of “leap-frog” structure.

$$\mathbf{A}=\left|\begin{matrix} 0 & 0.8032 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.0032 & 0 & 0.1861 & 0 & 0 & 0 & 0 & 0 \\ 0 & -0.1861 & 0 & 0.8651 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.8651 & 0 & 0.1886 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.1886 & 0 & 0.8417 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.8417 & 0 & 0.2668 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.2666 & 0 & 0.8874 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.8874 & -0.3332 \end{matrix}\right| \quad \mathbf{b}=\left|\begin{matrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0.34 \end{matrix}\right|$$

$$\mathbf{c}^T=[0.7434 \;\; 0 \;\; -0.2492 \;\; 0 \;\; -0.0292 \;\; 0 \;\; 0.0146 \;\; 0] \quad d=O$$

**Table 8.4: An Orthonormal System**

The system may be interpreted as a simulation of the (singly terminated) pole-forming ladder filter of figure 8.19, with transmission zeros formed by output summing.

The output noise spectrum (figure 8.20) and normalized sensitivity curve (figure 8.21) are comparable to those for a good cascade design: a little better in the passband and upper stopband and worse in the lower stopband.

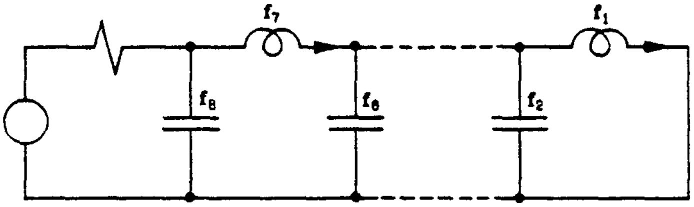


**Figure 8-19: Pole-forming LC Ladder with Orthonormal $\{f_i\}$**

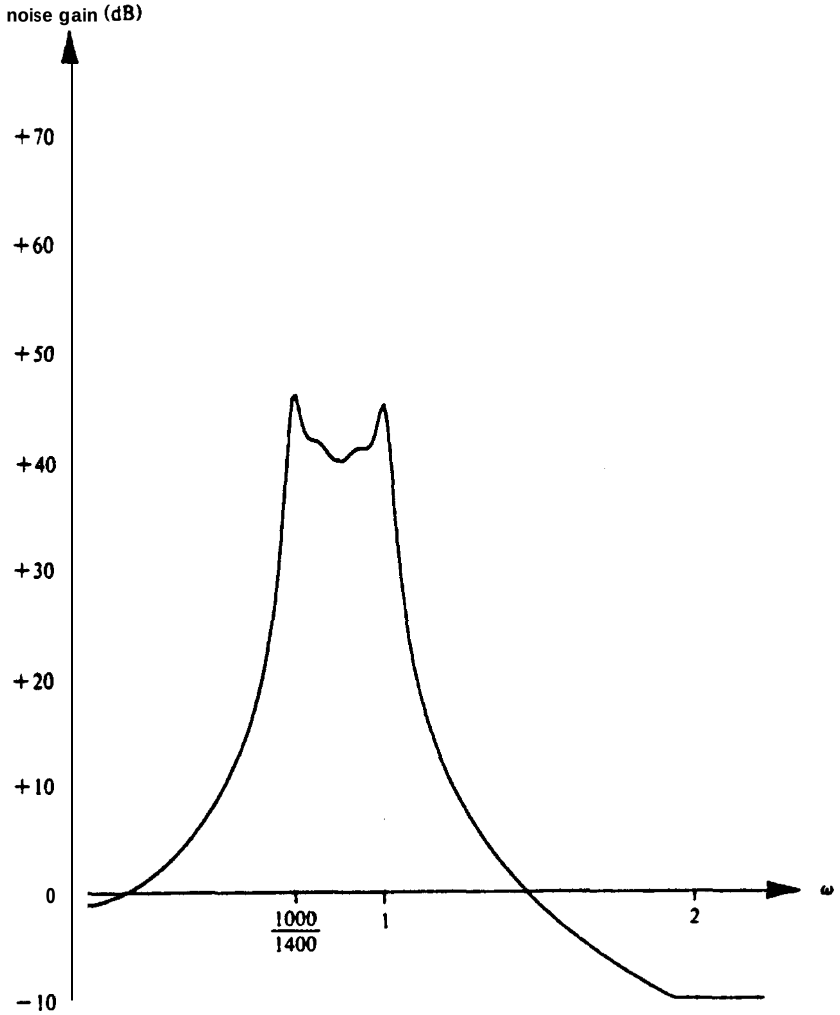


**Figure 8.20: Noise for Orthonormal System**

## 8.14 Derivative-Based Design

We know (chapter 7) that if an FSLB realization of a filter exists its $\{\mathbf{f}_i\}$ must divide $t'_s$. We can choose various sets $\{\mathbf{f}_i\}$ by combining the prime factors of $t'_s$ and investigate the resulting systems.

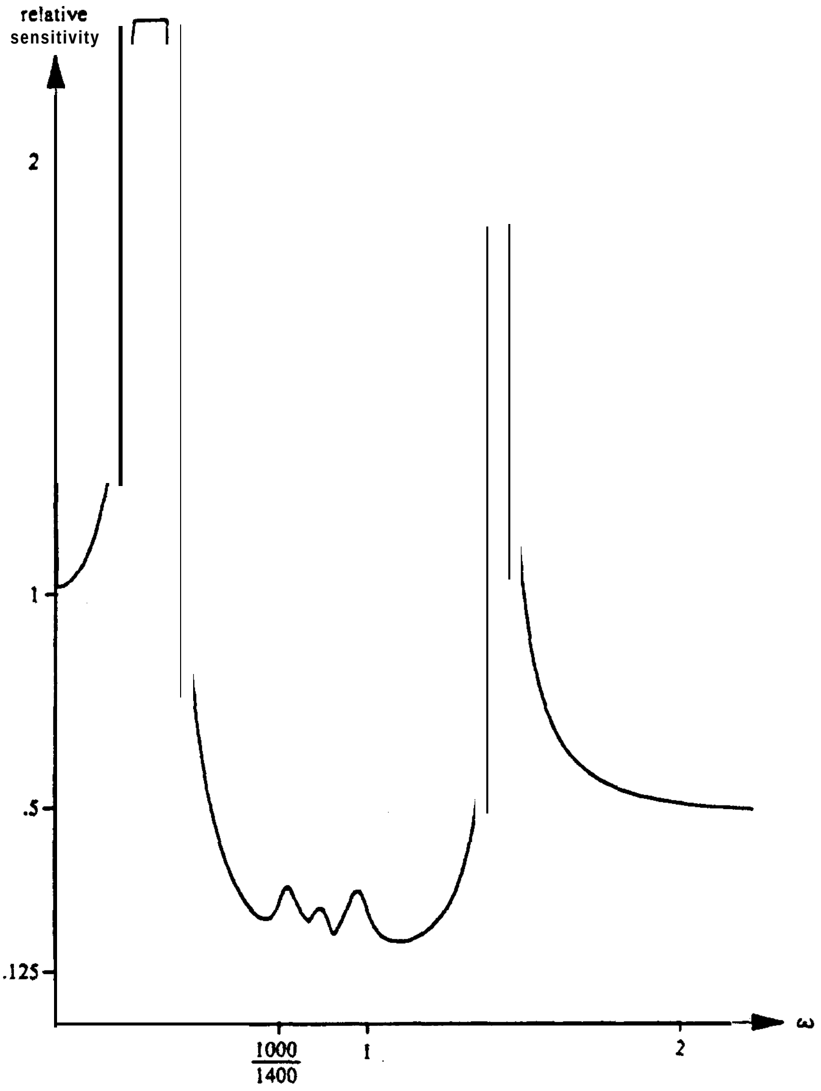


**Figure 8.21: Sensitivity for Orthonormal System**

Figure 8.22 is a sketch of the s-plane showing roots of both $t_s$ and $t'_s$. It shows with each root of the numerator of $t'_s$ a list of the index numbers $i$ of the $\{\mathbf{f}_i\}$ that contain that root for a particular design, and so provides a notation for a choice of $\{\mathbf{f}_i\}$.

The particuIar choice of $\{\mathbf{f}_i\}$ in figure 8.22 was made to have a number of symmetries, one of which gives a sort of "reciprocity" and one of which tends to force the system to have substructures resembling the FSLB biquads of chapter 7.

We expect some sort of "reciprocity" to be useful because we know that, because of the dual roles of $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$, taking the dual of a good system $(\{\tilde{\mathbf{f}}_i\}=\{\mathbf{g}_i\})$ should yield another good one. That in turn suggests that if a unique best system exists it must be self-dual. The $\{\mathbf{f}_i\}$ in figure 8.22 are so chosen that the factors of $t'_s$ missing from any $\mathbf{f}_i$ (which would be the factors of $\mathbf{g}_i$ if the filter happened to be FSLB) appear in $\mathbf{f}_{9-i}$, i.e. $\mathbf{f}_1\mathbf{f}_8=\mathbf{f}_2\mathbf{f}_7=\mathbf{f}_3\mathbf{f}_6=\mathbf{f}_4\mathbf{f}_5=t'_s$.

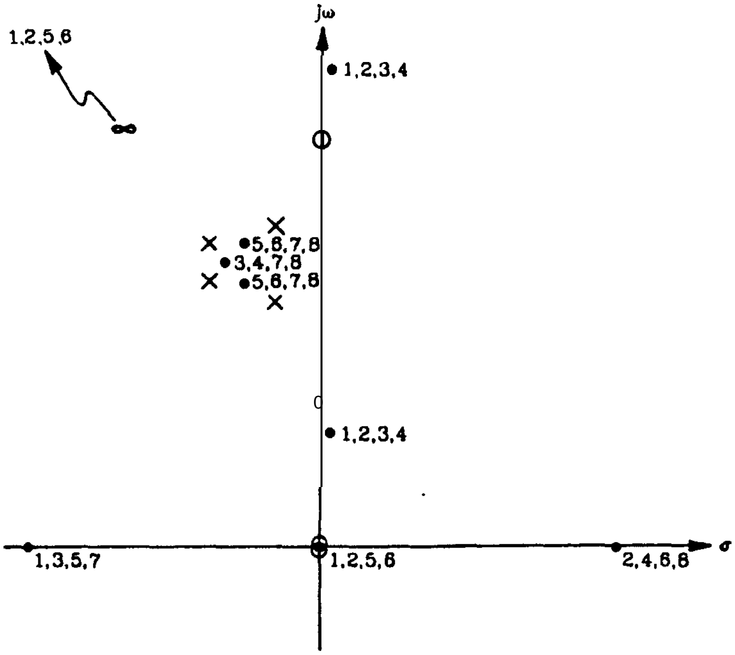


**Figure** 8.22: **Assignment of Derivative Roots to** $\{\mathbf{f}_i\}$

We have also chosen factors so that all $\{\mathbf{f}_i\}$ are bandpass in appearance, i.e. we have paired corresponding factors from above and below the bandcentre. This was done because the work in chapter 7 showed that FSLB structures of high orders were obtained from those of lower order by composition of functions; since our transfer function is generally bandpass in shape we should expect it to be composed of bandpass sub-filters.

The resulting system has, after $L_2$ scaling the $\{\mathbf{f}_i\}$ and using "dot" to find $\{\mathbf{g}_i\}$,

$W_{ii}$=[9.769 *9.771 9.482 9.475 6.532 6.527 11.21 11.191*

*so* that $\sum_{i} W_{ii} \cong 73.96$. This is very close to that of the best ladder circuit above. The output noise spectrum (figure 8.23) is also similar to that of a ladder.

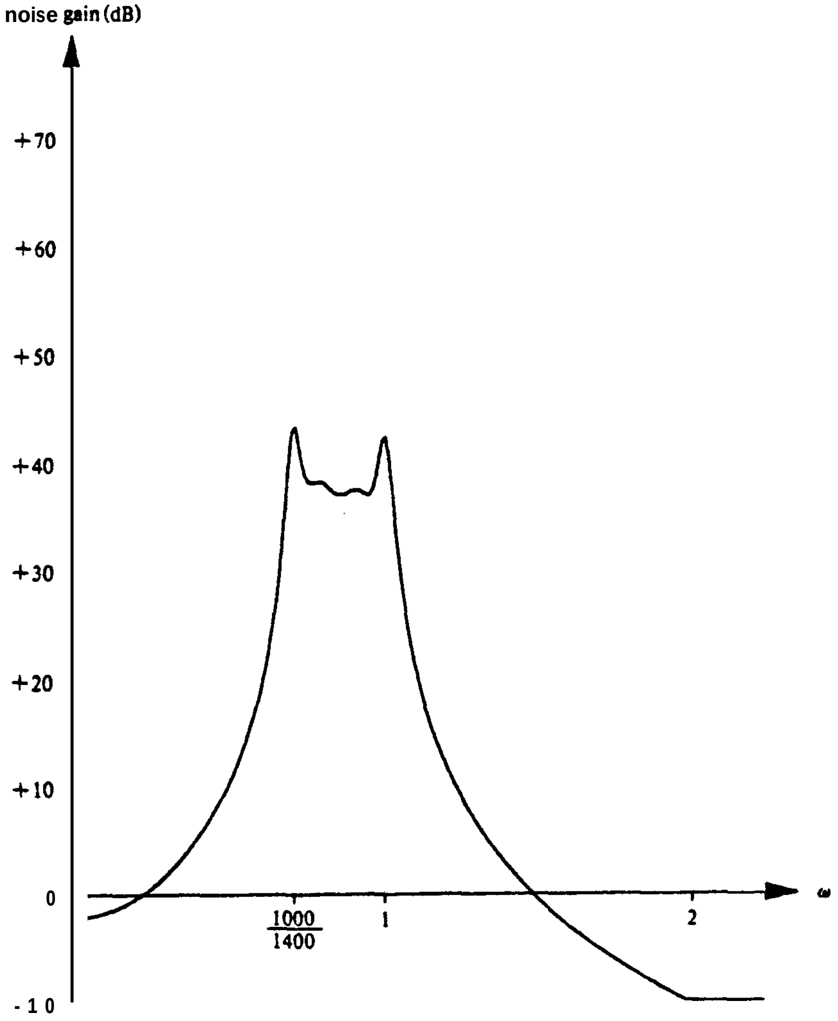


**Figure 8.23: Noise for Derivative-based Design**

Most interesting, however, is the sensitivity performance of this structure, figure 8.24. While the lower-stopband performance is mediocre, this filter is superior to any others in the upper stopband and the passband, where it is close to having FSLB performance $(S_{\gamma,\infty}=1/8)$. *It* has a small peak, $S_{\gamma,\infty}\cong 1/2$, at

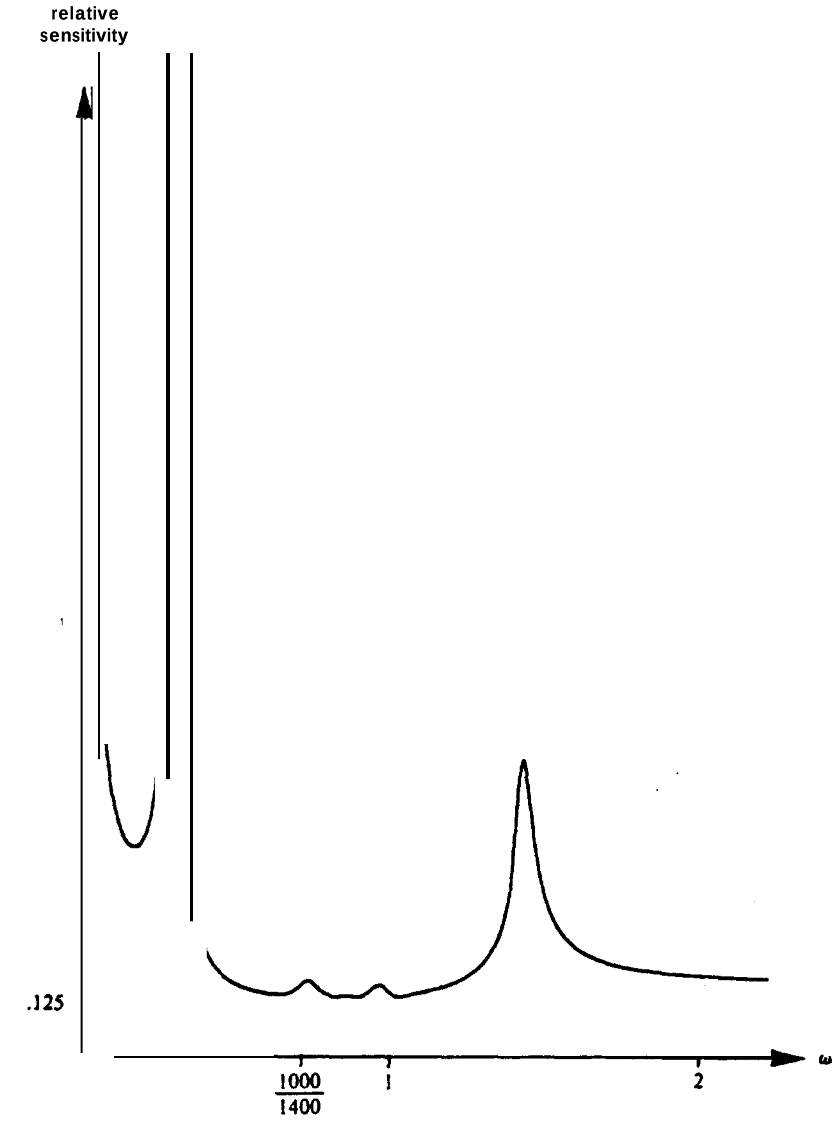


**Figure 8.24: Sensitivity of Derivative-based Design**

the upper stopband minimum. The fact that this is Iess than 1 clearly sets this filter apart from the cascade and ladder designs.

The system matrices, Table 8.5, are fairly dense (only 16 **A** entries are zero} but have strong structure, which derives from the structure imposed by the way we chose factors. Note, for example, that the first six columns of row **7** are directly proportional to the first six of row 8, so that a partial sum of these signals could be used for both integrators. This pair of integrators essentially forms a biquadratic section with a single input derived from the other six

$$A=\begin{vmatrix} 0.0265 & 0.7971 & 0.1167 & 0.1157 & 0 & 0 & 0 & 0 \\ -0.9194 & -0.1488 & 0.1155 & 0.1145 & 0 & 0 & 0 & 0 \\ -0.1223 & -0.1067 & 0.0006 & 0.8325 & 0.1194 & 0.1020 & -0.0221 & 0.0056 \\ -0.1210 & -0.1055 & -0.8658 & -0.1139 & 0.1182 & 0.1009 & -0.0219 & 0.0055 \\ 0 & 0 & 0 & 0 & 0.0265 & 0.797 & 0.1479 & 0.1466 \\ 0 & 0 & 0 & 0 & -0.91945 & -0.1488 & 0.1463 & 0.1451 \\ 0.0414 & 0.0371 & -0.0461 & -0.0402 & -0.0930 & -0.0807 & 0.0537 & 0.6072 \\ 0.0410 & 0.0367 & -0.0456 & -0.0398 & -0.0920 & -0.0799 & -0.6924 & -0.1388 \end{vmatrix} \quad b=\begin{vmatrix} 0 \\ 0 \\ 0.0092 \\ 0.0091 \\ 0.1557 \\ 0 \\ 0 \\ 0 \end{vmatrix}$$

$$c^T=[\,0.5849\ \ 0.5755\ \ -0.3479\ \ 0.0209\ \ 0.0231\ \ 0.0255\ \ 0.0010\ \ -0.0002]\quad d=O$$

**Table 8.5: A Derivative-Based System**

integrators and distributed almost equally to both of its integrators. This is basically the form of the optimum bandpass biquad derived in chapter 7.

Integrator pairs (1,2), (3,4 and (5,6) may similarly be grouped into biquads.

**Relationship to Ladders**

The fact that many of the roots of $t_s'$ lie close to roots of intermediate functions for the doubly terminated ladder (section 8.2) suggests that one could "correct" roots in the ladder simulation by moving them to nearby $t_s'$ roots. By this means it is possible to obtain an infinite class of filters with performance intermediate between that of a ladder and that of the derivative-based design.

## &15 Summary and Conclusions

We have used IF synthesis to investigate, at the **{A,b,c,d]** system matrix level, several competing designs for a non-trivial (eighth order) bandpass filter taken from the open literature. Comparisons were made of dynamic range and integrator sensitivity performance, which have been the dominant concerns of preceding chapters.

A new type of ladder simulation was developed which avoids the need to simulate “extra” states with their concomitant problems at high frequencies and near DC. It was shown that the detailed choice of which states to simulate strongly affected the performance of the simulation, but IF synthesis made it easy to experiment, and compare alternatives. The best simulations produced were shown to be close to lower bounds on noise and passband sensitivity performance.

Cascade syntheses were also done, and a new rule for section ordering inspired by the duality between $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ was tested and found to be superior to that current in the art. Good cascades appeared to be approximately a factor n/2 worse than lower bounds in passband sensitivity: this suggests a straightforward rule of thumb by means of which a designer may compare cascade and highly coupled structures.

The simple sensitivity performance figures for ladder and cascade structures suggest a new strategy for transfer function approximation, which is is developed in chapter 10.

Companion form was investigated and the cause of its weakness pointed out.

The Gram-Schmidt procedure was used to modify companion form to a structure with orthonormal states, which turned out to have sensitivity and dynamic range performance generally comparable to those of cascade designs and could be seen as a simulation of a singly-terminated pole-forming ladder.

A new type of structure was developed by choosing $\{\mathbf{f}_i\}$ from the zeros of $t'_s$. It turned out to have better passband and upper stopband performance than the doubly-terminated ladder, but to perform fairly poorly in the lower stopband. The fact that integrators are relatively well-behaved at low frequencies suggests that this type of design could be preferable to a ladder simulation. The technique of basing a design on $t'_s$ is applicable to a wider range of transfer functions than is ladder simulation because there are restrictions on the class of functions for which ladder synthesis works and because ladder simulations only try to obtain good performance near transmission maxima. This suggests that derivative-based design offers a new approach to synthesis of unusual types of transfer function.

Intermediate-function synthesis appears to be a powerful and practical design tool.

## 8.16 References

[1] P.O. Brackett and A.S. Sedra, ***Direct SFG Simulation of LC Ladder Networks with Applications to Active Fili!e?- Design,*** IEEE Trans. Circuits and Systems, vol. CAS-23, pp. 61-67, Feb. 1976

[2] P.O. Brackett, ***The OpeTationa.l Simulation of Filte'r Networks*** Ph.D. Thesis, University of Toronto, Toronto, Canada 1976

[3] A. Fettweis "Some ***Principles of Designing Digital Filters Imitating Classical Filter Structures",*** IEEE Trans. Circuit Theory, CT-18, no. 2, pp. 314-316, March 1971

[4] H.G. Dimopoulos and A.G. Constantinides, ***Linear Transfomation Active Filters*** IEEE Trans. Circuits Syst., CAS-25, pp. 845-852, Oct. 1978

[5] G.C. Temes, "***MOS Switched- Capacitor Filter's – History and State of the Art***", European Conf. Circuit Theory and Design, The Hague, The Netherlands, August 1981, pp.l76-185

[6] F.E.J. Girling and E.F. Good, ***"Active Filter - 12. The Leap Frog or Active Ladder Synthesis",*** Wireless World, pp. 341-345, July 1970

[7] W.J. Kerwin, L.P. Huelsman and R.W. Newcomb, ***"State- vatiable Synthesis for Insensitive Integrated- circuit Transfer Functions",*** IEEE J. Solid-state Circuits, SC-2, pp. 87-92, Sept. 1967

[8] E. Lueder ***"Optimization of the Dynamic Range and the Noise Distance of RC- Active Filters by Dynamic Programming***", Int. J. of Circuit Theory Appl., vol. 3, pp. 365-370, Dec. 1975

[9] W-M. Snelgrove and AS. Sedra, *"Optimization of Dynamic Range in Cascade Active Filters",* Proc. IEEE Int. Symp. on Circuits and Systems, New York 1978, pp. 151-155

[l0] A.S. Sedra and P.O. Brackett, ***Filter Theory? and Design: Active and Passive*** Matrix Publishers, Champagne, Illinois, 1978

[11] D.G. Luenberger, ***Optimization by Vector Space Methods,*** Wiley, 1969

[l2] C.T. Mullis and R.A. Roberts, ***Synthesis of Minimum Roundoff Noise Fixed- point Digital Filters*** IEEE Trans. Circuits and Systems, vol. CAS-23, pp. 551-562, Sept. 1976.

*[13]* Moschytz, G.S., ***Linear Integrated Networks,*** Van Nostrand Reinhold 1975

[14] J. Tow, ***"Active RC Filters – a State- Space Realization",*** Proc. IEEE, vo1. 56, pp. 1137-1139, 1968

# 9. Synthesis of Complex Filters

Intermediate-function synthesis has proven useful in the investigation of "complex filters" [1,2], by which is meant filters whose transfer functions may have complex coefficients. This type of filter is shown to be potentially useful.

## 9.1 Introduction to Complex Signals

This section presents the concept of a complex analog signal. It will be useful in defining and designing the networks required for, inter alia, single-sideband (SSB) generation. This section is tutorial in nature because this chapter applies a concept known in signal processing to circuit design, where it is not familiar.

One conventionally thinks of analog signals as being real-valued because they are represented by (real-valued) voltages on wires. If, however, we have a pair of wires at different voltages (to ground, presumably) $V_1$ and $V_2$, we may think of the pair of wires as carrying the "complex voltage" $V \triangleq V_1+jV_2$. Now, if $V_1$ and $V_2$ vary with time (i.e. are signals) so does the fictitious $V$, and we call it a "complex signal" (or a "complex-valued" signal). This procedure is no different from that we follow when we write a complex number as, e.g., $3+j4$ or **(3,4): we** are resolving the complex quantity (in a particular one of infinitely many possible ways) into an ordered pair of real numbers.

The properties of a "complex signal" in the time domain, then, are fairly straightforward: it is just an ordered pair of real signals. In the frequency domain the difference between real and complex signals is that the spectrum of a real signal must be symmetric about DC [3] while no such constraint applies to the spectra of complex signals. To take a concrete and important example, if $V_1=\cos(\omega t)$ and $V_2=\sin(\omega t)$ then by definition $V=\cos(\omega t)+j\sin(\omega t)=e^{j\omega t}$, which has power only at a positive frequency $\omega$. A real signal, like

$V_1=\cos(\omega t)=\frac{e^{j\omega t}+e^{-j\omega t}}{2}$, must have equal power at positive and negative frequencies.

We will want to do two types of things with complex signals: linear filtering and modulation. We will also want to convert real signals to complex ones and vice-versa.

**Modulation**

Modulation just involves multiplication of **a** pair of signals to produce a result. We define multiplication in the usual way, $(a+jb)(c+jd)\triangleq(ac-bd)+j(ad+bc)$, and can draw the resulting "complex modulator" as shown in figure 9.1.

We can easily check that the structure of figure 9.1 does what we would expect of. a modulator, i.e. convolves spectra, on a simple example. If $e^{j\omega t}$ and $e^{j(2\omega)t}$ arc multiplied this way one may verify that the result is $e^{j(3\omega)t}$. Complex modulation by $e^{j\omega t}$ is actually conceptually simpler than modulation by a real signal like $\cos(\omega t)$, because the former involves frequency-domain convolution with a single &function at $j\omega$ while the latter involves convolution with a pair of &functions (one at $+j\omega$ and another at $-j\omega$). Thus complex modulation involves a simple frequency shift, while "real" modulation involves adding the results of shifts both up and down in frequency.

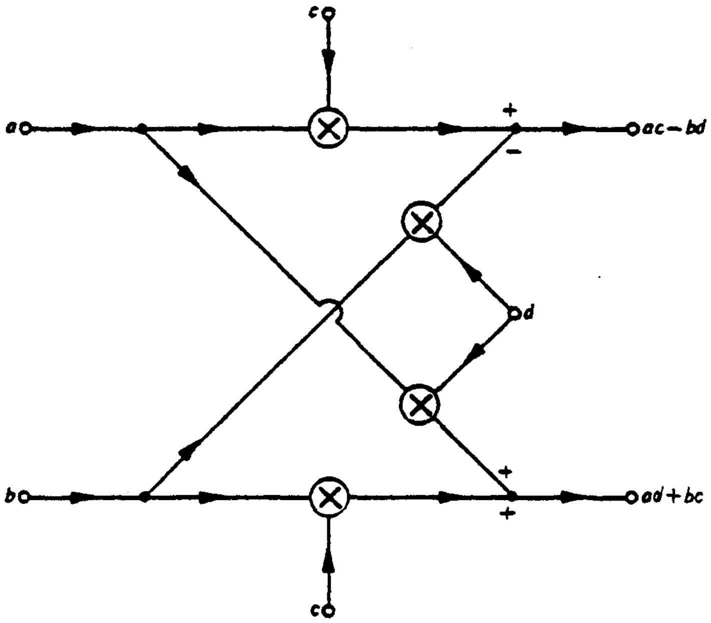
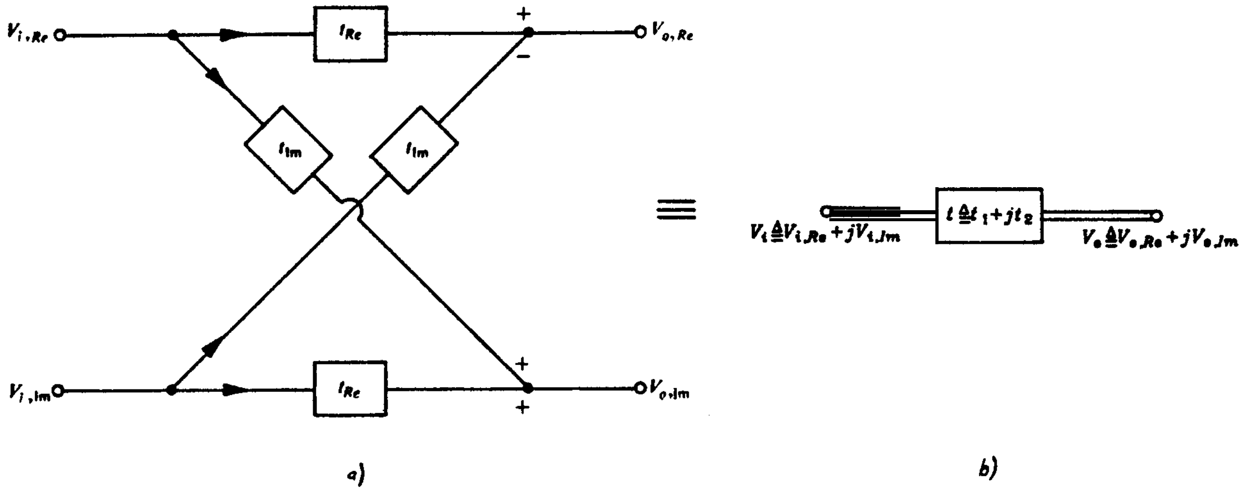


**Figure** 9.1: A Complex Modulator

**Complex Transfer** Functions

We can make two-input two-output linear systems, and so we can make systems that process complex signals. In particular, the four ordinary single-input single-output (SISO) systems in figure 9.2a process the complex signal $V_i \triangleq V_{i,Re} + jV_{i,Im}$ applied at the left to give a complex result at the right.



$V_{i,Re}$
$t_{Re}$
+
−
$V_{o,Re}$
$t_{Im}$
$t_{Im}$
$\equiv$
$V_i \triangleq V_{i,Re} + jV_{i,Im}$
$t \triangleq t_1 + jt_2$
$V_o \triangleq V_{o,Re} + jV_{o,Im}$
$V_{i,Im}$
$t_{Re}$
+
+
$V_{o,Im}$
a)
b)

**Figure 9.2: Two-Input Two-Output System Equivalent to a Complex System**

Notice that we have started to use subscripts "*Re*" and "*Im*" to distinguish the two real signals comprising a complex signal, rather than the anonymous 1 and 2 that we used earlier: this is just a mnemonic device and need not necessarily imply anything about a relationship between the signals.

It is straightforward to verify from the figure that the "complex output" $V_o \triangleq V_{o,Re} + jV_{o,Im}$ *is* just

$$V_{o,s} = t_s V_{i,s}$$

by substituting the definitions of $V_o, t_s$, and $V_i$ given in figure 9.2b. We can therefore draw the network of figure 9.2a in the shorthand form of figure 9.2b: as a system with a single (complex) input (shown as a double line to emphasize that a pair of real signals is involved) and a single (complex) output $V_o$ related by a transfer function $t_s$.

Just as signals with complex values as functions of time can have spectra asymmetric about DC, so filters like $t_s(s)$ with complex inputs and outputs can have asymmetric transfer functions. The individual SISO transfer functions $t_{Re,s}$ and $t_{Im,s}$ of figure 9.2 had to have real coefficients as functions of s and so $|t_{Re,s}(j\omega)|=|t_{Re,s}(-j\omega)|$, but $t \triangleq t_{Re,s}+jt_{Im,s}$ can have complex coefficients and there is no symmetry imposed at all.

This chapter is concerned with the design and synthesis of complex $t_s(s)$ (which we often call “complex-coefficient transfer functions” when “complex” alone might be misread to mean “complicated”) especially when the function desired is a filter.

**Converting Real Signals to and from Complex Signals**

Complex signals may be used internally in systems processing real inputs to get rea1 outputs. A real signal may be “converted” to a complex one by the trivial operation of adding an imaginary part of value zero, which has no effect on the signal’s spectrum.

A complex signal may be converted to a real one by ignoring its imaginary part (which obviously destroys information) but this does affect the spectrum: it causes a type of “aliasing between positive and negative frequency components of a signal. This issue will be discussed in section 9.3 below.

### 9.1.1 Application to SSB Generation

The block diagram of figure 9.3 shows how to construct a single-sideband (SSB} modulator* using complex signals internally. The first stage is a “positive-pass”

• **Here we are simply deriving Weaver’s second method of single-sideband generation in such a**

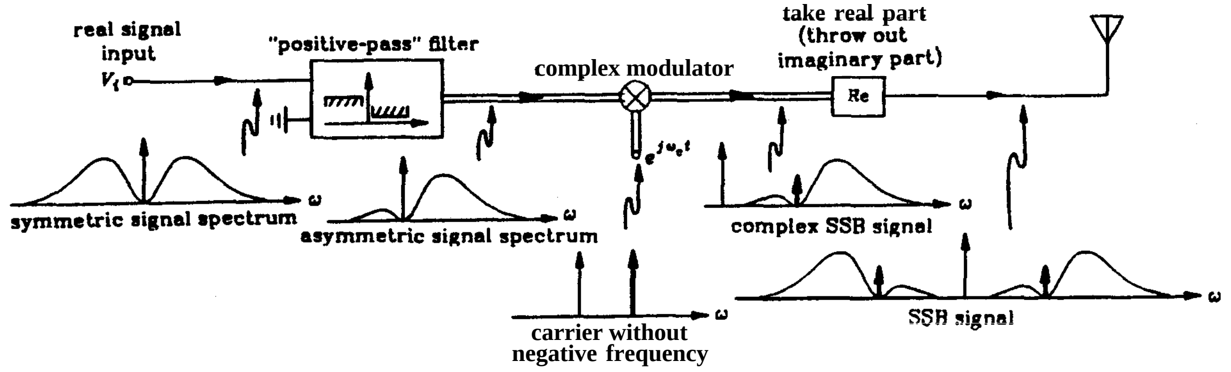


Figure 9.3: An SSB Modulator using Complex Signals

complex transfer function, i.e. a “complex filter” designed to pass positive and reject negative frequencies (which is perfectly permissible because transfer functions are not restricted to be symmetric around DC). The next part is a complex modulator, which shifts the spectrum of the complex signal at its input up by a carrier frequency $\omega_c$ by multiplying by $e^{j\omega_c t}$. Notice that the result has, at positive frequencies, the spectral asymmetry (upper sideband only) that we desire of a single-sideband signal. The final stage takes the real part of the complex SSB signal, which has the effect of adding a mirror image of the complex SSB spectrum at negative frequencies, resulting in a symmetric output spectrum.

The problem of specifying the positive-pass filter so that the overall system has given performance as an SSB modulator is straightforward. If, say, **40dB** of lower-sideband rejection is required, the input filter must suppress negative frequencies by 4OdB; if **0.5dB** of variation is permissible in the upper sideband, and frequencies from 300Hz to 3OOOHz are to be passed, a 3OO-3OOOHz passband with 0.5dB of ripple is required.

way as to motivate complex network synthesis.

### 9.1.2 Analytic Signals

A positive-pass filter like that needed for the SSB modulator above is essentially a way to aproximate the "analytic signal" [4,5] which is the result of removing negative-frequency components from an ordinary "real signal". Analytic signals have the interesting property of allowing measurements of the "instantaneous magnitude" and "instantaneous phase" of a signal.

If $\cos(\omega t)$ is applied to the input of an ideal positive-pass complex filter the output (i.e. the analytic signal) will be $V_o = e^{j\omega t} = \cos(\omega t) + j\sin(\omega t)$. Now we can measure the rms amplitude of the complex signal *at any instant t* as the rms value of the two component signals: $|V_o| = \sqrt{\cos^2(\omega t) + \sin^2(\omega t)} = 1$. By comparison, we cannot estimate the amplitude of a sinusoid from its value at any single point (unless we are also told its phase).

Similarly, we can define the "instantaneous phase", $arg(V_o) = \omega t$, and its derivative, "instantaneous frequency". These measures are sometimes used for theoretical reasons, but have also been suggested for such practical applications as bandwidth compression [4].

### 9.1.3 The Relationship between Real and Imaginary Parts of a Positive-Pass Complex Signal

We saw above that *an* ideal "positive-pass" filter presented with $\cos(\omega t)$ would produce a pair of outputs, $\cos(\omega t)$ and $\sin(\omega t)$, with exactly equal amplitude and a *90a* difference in phase. In general any signal containing only power at positive frequencies wil1 have its outputs in this relationship one to the other: the outputs of an ideal positive-pass filter (i.e. the real and imaginary parts of an analytic signal) are related by the Hilbert transform.

This phase and magnitude relationship comes about just because it applies to every Fourier component of an arbitrary signal. If a signal is

$$\sigma_t(t)=\int_0^\infty(\sigma_{\omega,R}(\omega)\cos(\omega t)+\sigma_{\omega,I}(\omega)\sin(\omega t))d\omega$$

then the corresponding analytic signal is

$$\hat{\sigma}_t(t)=\int_0^\infty(\sigma_{\omega,R}(\omega)e^{j\omega t}+\sigma_{\omega,I}(\omega)e^{j\omega t-\frac{\pi}{2}})d\omega$$

which we can decompose into real and imaginary parts:

$$=\int_0^\infty(\sigma_{\omega,R}(\omega)\cos(\omega t)+\sigma_{\omega,I}(\omega)\sin(\omega t))d\omega \qquad (9\text{-}1)$$

$$+j\int_a^\infty(\sigma_{\omega,R}(\omega)\sin(\omega t)-\sigma_{\omega,I}(\omega)\cos(\omega t))d\omega$$

in which every frequency component of the real part (just the first half of (Q-l)) can be seen to correspond to a component of the imaginary part (second half of (Q-l)) with equal amplitude and whose phase lags by *90°*.

### 9.1.4 History

The concept of a complex signal is well-known in digital signal processing and communications theory [5] and is used for theoretical purposes in network theory [12,15] in particular the signals appearing in machines implementing the Fast Fourier Transform [5] are usually treated this way, and the “analytic signal” [4,5] in which real and imaginary parts are related by the Hilbert transform is used in communications theory. The SSB modulator we described above is a known derivation of Weaver’s second method of single-sideband generation [6], but the matched-allpass synthesis approach usually taken for implementation of these modulators is a very special case of the ones we propose here.

One can convert the structure of figure Q.3 into that conventionally used to implement Weaver’s second method of SSB generation by substituting into it the implementations of modulators and filters suggested by figures 9.1 and 9.2. Because the input is real-valued, the two SISO blocks processing its imaginary part suggested by figure 9.2 may be eliminated*; because only the real part of the modulator's output is used, the two multipliers involved in producing the imaginary part (cf. figure 9.1) may also be eliminated. The result is the well-known structure shown in figure 9.4.

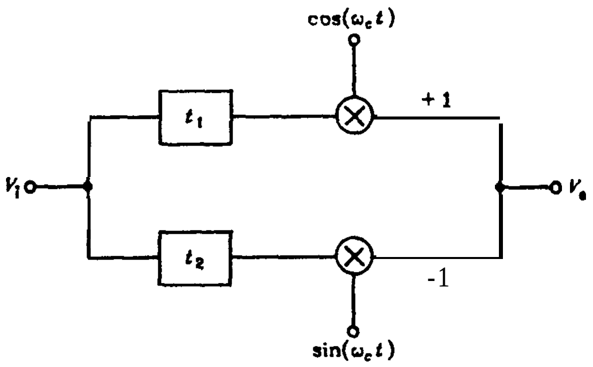


Figure 9.4: **Weaver's Second Method of SSB Generation**

The conventional approach to the approximation problem (choosing $t_1$ and $t_2$) has been to derive a pair of all-pass functions whose phase difference approximates 90' over the frequency band of interest. Because the functions are all-pass (i.e. $|t_1|=|t_2|=1$) their magnitudes must match, and because they **also** approximate 90° of relative phase, the pair $(t_1+jt_2)$ approximates a positive-pass filter

We show in this chapter that this restricted approach to approximation and synthesis is unnecessary (because much more general extensions of ordinary filter approximation and synthesis may be used) and undesirable (because we can derive networks of other forms with better performance).

* Note that the result is a one-input two-output Iinear system characterized by a complex-coefficient transfer function. We will often be more interested in this special case than in the two-input two-output case.

## 9.2 Approximation

An extensive theory is available [7] for approximation of conjugate specifications by transfer functions with real coefficients, but the corresponding theory for complex filters is limited: Lang [1] suggests an approximation technique involving a shifted bandpass design, while Tsuchiya and Shida [8] develop some theory for a special case in which natural modes are pure real but transmission and reflection zeros complex. We give here the complex equivalents of the exact formulae available in "real filter" design and a theory for transforming between complex and real filters to allow designers to use existing techniques to design complex filters.

### 9.2.1 Feldtkeller's Equation

It is generally easier to approximate characteristic functions $k_s$ (ideally 0 in passbands and $\infty$ ln stopbands) than attenuation functions $h_s$ (ideally 1 in passbands and $\infty$ in stopbands), In "real filter" design Feldtkeller's equation allows one to derive an $h_s$ from any $k_s$ such that

$$|h_s(j\omega)|^2=1+|k_s(j\omega)|^2 \tag{9-2}$$

(where the factor $\varepsilon$ sometimes used is subsumed here in $k_s$).

This much applies equally well to complex filters. In "real filters" a pair of identities is then used to produce an equation in rational functions that is identical to (9-2) over the $j\omega$ axis. For any rational function $f_s(s)$ with real coefficients

$$|f_s(s)|^2=f_s(s)\overline{f_s(s)} \tag{9-3}$$

$$=f_s(s)f_s(-s)|_{s=j\omega} \tag{9-4}$$

where $\overline{f_s(s)}$ is the conjugate of $f_s(s)$, *so* that (9-2) may be re-written

$$h_s(s)h_s(-s)=1+k_s(s)k_s(-s) \tag{9-5}$$

Now this is not directly applicable to complex filters, because (9-4) only holds when $f$ has. real coefficients. (9-3) still applies, but cannot be used directly because $\overline{f_s(s)}$ is not a rational function of $s$, which is why (9-4) was needed to analytically continue (9-2) from the imaginary axis to the whole s-plane. To solve this problem we define a function $\overline{f_s}$, *so* that $\overline{f_s}(s)$ (note that this is *not* $\overline{f_s(s)}$) has coefficients conjugate to those of $f_s(s)$. ***We*** then have

$$\overline{f_s(s)}=\overline{f_s}(\overline{s})$$

and on $s=j\omega, \overline{s}=-s$, so

$$|f_s(s)|^2{}_{s=j\omega}=f_s(s)\overline{f_s}(-s)|_{s=j\omega}$$

and now the right-hand-side is rational.* Our new version of (9-2), then, is

$$h_s\overline{h_s}(-s)=1+k_s(s)\overline{k_s}(-s) \tag{9-6}$$

from which, by looking at the numerator and denominator polynomials

$$h_s \triangleq \frac{e_s}{p_s}$$

$$k_s \triangleq \frac{f_s}{p_s}$$

we can derive a formula in polynomials

$$e_s(s)\overline{e_s}(-s)=f_s(s)\overline{f_s}(-s)+p_s(s)\overline{p_s}(-s) \tag{9-7}$$

This formula is suitable for computation of $e_s(s)\overline{e_s}(-s)$ from a given $k_s$, and produces roots symmetrical about the $j\omega$ axis, so that pole-sorting may be used to

* Kuh and Rohrer [15] use compIex power definitions like this for their discussion of complex one-port networks.

construct a stable $e_s$.

Note that (9-6) degenerates to (9-2) for the real case.

For discrete-time filters we may similarly derive

$$h_z(z)\overline{h_z}(z^{-1)}=1+k_z(z)\overline{k_z}(z^{-1)} \tag{9-8}$$

**Example**

Take

$$f_s(s)\triangleq s-j$$

$$p_s\triangleq s+j$$

Then

$$\overline{f_s}(s)=s+j$$

$$\overline{p_s}(s)=s-j$$

and

$$e_s\overline{e_s}(-s)=(s+j)(-s-j)+(s-j)(-s+j)$$

$$=2(1-s)(1+s)$$

and we can sort roots to get a stable

$$e_s=\sqrt{2}(1+s)$$

This is a first-order approximation to an SSB filter specification It may be implemented as a first-order filter with two outputs: a “real part” of $\frac{s}{\sqrt{2}(s+1)}$

and an “imaginary part” of $\frac{1}{\sqrt{2}(s+1)}$.

The fact that $e_s$ happened to have real coefficients in the above example is a coincidence arising from the symmetrical relationship between $f_s$ and $p_s$. This symmetrical case is the one studied by Tsuchiya and Shida [8], although much of their work can be applied more generally.

## 9.2.2 Complex Filters from Real Prototypes

***The*** availability of a form of Feldtkeller’s equation for complex filters makes it possible to get transfer functions from given characteristic functions. Characteristic functions may be obtained by doing simultaneous approximation on passbands and stopbands after the fashion of [9]; by using classical approximations (e.g. Chebyshev polynomials); or by using an approximator on stopband specifications together with closed-form formulae giving $f_s(s)$ to get an ideal passband given any $p_s$.

### Arithmetically Symmetric Complex Filters

If the response $\tilde{t}_s(\tilde{s})$ of an ordinary “real” filter is translated by some $j\omega_0$ (i.e. $t_s(s)=\tilde{t}_s(s-j\omega_0)$) the resulting transfer function (cf. figure 9.4) has arithmetic symmetry about $s=j\omega_0$ because the “real” prototype $\tilde{t}_s$ was symmetric about DC **(s=O).** This offers a particularly simple approach to the approximation of complex transfer functions that is applicable when arithmetic symmetry is acceptable.

This design procedure, which is analogous to that involved [7] in using the lowpass-to-bandpass transformation, involves these steps:

i. Given specifications are “symmetrized” about some $\omega_0$ by Ending a (more ,conservative) set of specifications with arithmetic symmetry.

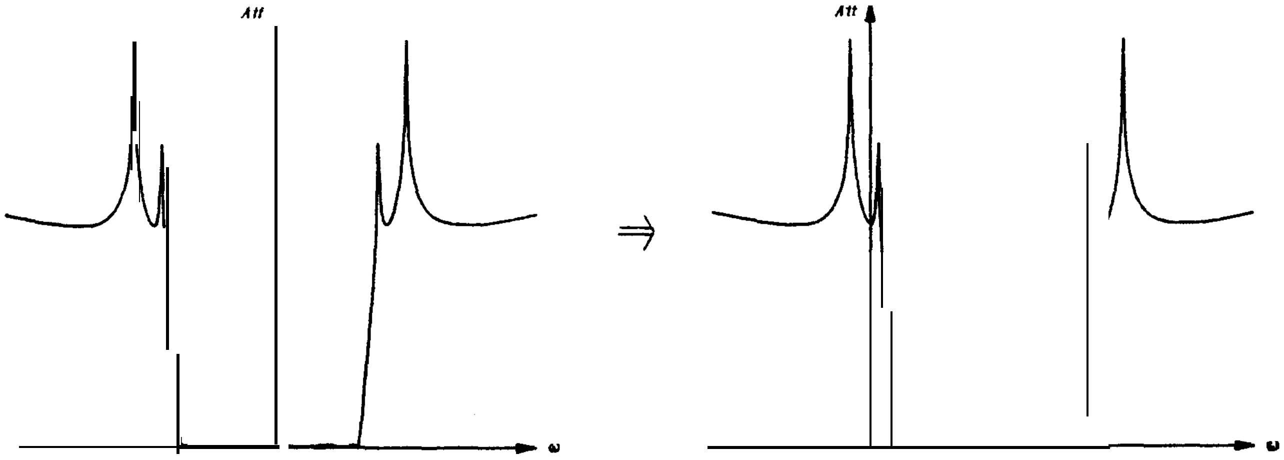


**Figure 9.5: Obtaining a Complex Filter by Translation of a** Real **Prototype**

ii. The symmetric specifications are then shifted down by $\omega_0$ to generate an ordinary real-filter design problem.

iii. A "real" filter is obtained meeting the specifications of "ii.", and

iv. the real filter prototype design is shifted back up to $\omega_0$ by applying the transformation $s=s+j\omega_0$ to its poles and zeros.

The final step of this procedure will appear later to have a simple correspondence with a network transformation in "LCX" filters, just as the lowpass-to-bandpass transformation could be applied directly to an LC network. This is the principal reason for our interest in "shifted real" complex filters.

**Example**

**The** function $\tilde{t}_s(\tilde{s})=\frac{1}{\tilde{s}+1}$ is a first-order Butter-worth (real) filter with a 3db passband from -lrad/s to lrad/s. Shifting all poles and zeros up by $j$ gives $t_s=\frac{1}{s+1-j}$, which has an arithmetically symmetric (about s=j) response with a (3dB) passband from DC to +2rad/s.

## Asymmetric Complex Filters

This section gives a transformation that may be used to convert a “real filter” of order N into a complex filter of order $\frac{N}{2}$ whose positive-frequency behaviour comes from the real prototype’s $j\omega$ axis performance. This may be used to design complex filters using well-known techniques for designing real ones or to derive design aids for complex filters equivalent to those available for real filters. The principal advantage of this technique over the “shifted-real” approach given above lies in its higher efficiency in meeting asymmetric specifications.

We first present the transformation in its simplest form, and show that it has the stated property, then show how to use it to do design.

## Real-to-Complex

The mapping

$$s=\frac{\tilde{s}^2}{j}$$

changes a function $\tilde{h}_s(\tilde{s})$ into an $h_s$ such that

$$|\tilde{h}_s(\tilde{s})|^2=|h_s|^2$$

on $\tilde{s}=j\tilde{\omega}$. Positive frequencies in $.s$ map to the imaginary axis in $\tilde{s}$ while negative frequencies map to real $\tilde{s}$. Under certain conditions on $\tilde{h}_s(s)$ a stable $h_s$ may be obtained, and so the transformation may be used to design complex filters with stated magnitude performance on positive frequencies. We show later how to transform an arbitrary complex filter problem to this form, so that this can be used to do design over the whole imaginary axis. Let us first derive the effect of the transformation.

We have to assume a special structure for $\tilde{h}_s(\tilde{s})$: that all roots of $\tilde{h}_s(\tilde{s})$ have complex conjugates. This means in practice that the order N of $\tilde{h}_s(\tilde{s})$ must be even, which is not too surprising since the transformation divides order by 2. The restriction also requires that real-axis transmission zeros in $\tilde{h}_s$ be of even multiplicity. As a further point, real-axis natural modes in $\tilde{h}_s$ *are* not permitted because they would transform to imaginary-axis roots in $h_s$ and cause marginal instability.

Assume $\tilde{h}_s(\tilde{s})$ may be written

$$\tilde{h}_s(\tilde{s})=c\frac{\prod_1^{N/2}(\tilde{s}-\tilde{e}_i)\cdot\prod_1^{N/2}(\tilde{s}-\bar{\tilde{e}}_i)}{\prod_1^{M/2}(\tilde{s}-\tilde{p}_i)\cdot\prod_1^{M/2}(\tilde{s}-\bar{\tilde{p}}_i)}$$

Then

$$\tilde{h}_s(\tilde{s})\tilde{h}_s(-\tilde{s})=c^2\frac{\prod_1^{N/2}(\tilde{s}-\tilde{e}_i)}{\prod_1^{M/2}(\tilde{s}-\tilde{p}_i)}\frac{\prod_1^{N/2}(\tilde{s}-\bar{\tilde{e}}_i)}{\prod_1^{M/2}(\tilde{s}-\bar{\tilde{p}}_i)}\frac{\prod_1^{N/2}(-\tilde{s}-\tilde{e}_i)}{\prod_1^{M/2}(-\tilde{s}-\tilde{p}_i)}\frac{\prod_1^{N/2}(-\tilde{s}-\bar{\tilde{e}}_i)}{\prod_1^{M/2}(-\tilde{s}-\bar{\tilde{p}}_i)}$$

collecting first and third terms in numerator and denominator, and likewise second and fourth,

$$=c^2\frac{\prod_1^{N/2}(-\tilde{s}^2+\tilde{e}_i^2)\cdot\prod_1^{N/2}(-\tilde{s}^2+\bar{\tilde{e}}_i^2)}{\prod_1^{M/2}(-\tilde{s}^2+\tilde{p}_i^2)\cdot\prod_1^{M/2}(-\tilde{s}^2+\bar{\tilde{p}}_i^2)}$$

and now substituting $sj=\tilde{s}^2$

$$=c^2\frac{\prod_1^{N/2}(-sj+\tilde{e}_i^2)\cdot\prod_1^{N/2}(-sj+\bar{\tilde{e}}_i^2)}{\prod_1^{M/2}(-sj+\tilde{p}_i^2)\cdot\prod_1^{M/2}(-sj+\bar{\tilde{p}}_i^2)}$$

now multiplying terms of the first products (both in numerator and denominator) by j and dividing the second set of terms by j:

$$=c^2\frac{\prod_1^{N/2}(s+j\tilde{e}_i^2)\cdot\prod_1^{N/2}(-s-j\overline{\tilde{e}_i^2})}{\prod_1^{M/2}(s+j\tilde{p}_i^2)\cdot\prod_1^{M/2}(-s-j\overline{\tilde{p}_i^2})}$$

$$=h_s\overline{h_s}(-s)$$

where

$$h_s\triangleq c\frac{\prod_1^{N/2}(s+j\tilde{e}_i^2)}{\prod_1^{M/2}(s+j\tilde{p}_i^2)}$$

$$\triangleq c\frac{\prod_1^{N/2}(s-e_i)}{\prod_1^{N/2}(s-p_i)}$$

The choice of roots for $h_s$ should be made for stability.

The above shows what we wanted: that substituting $s=\frac{\tilde{s}^2}{j}$ preserves the behaviour of transfer function magnitude.

Example

This section gives a simple example of complex filter design using the transformation just obtained. The technique used here has a restriction, in that it may not be used to approximate specifications that go down to $-j\infty$. A more general approach that does not contain this restriction appears later.

We wish to design a filter (for an SSB application) meeting specifications as shown in figure 9.6:

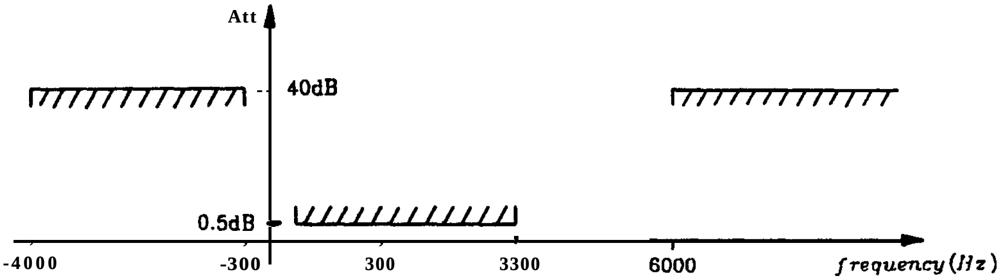


**Figure 9.6: Complex Specifications for SSB Prefilter**

we first shift them to positive frequencies and normalize with the Iinear transformation $p = \frac{(s + 4000 \cdot 2\Pi)}{7300 \cdot 2\Pi}$, getting a new set of specifications (figure 9.7).

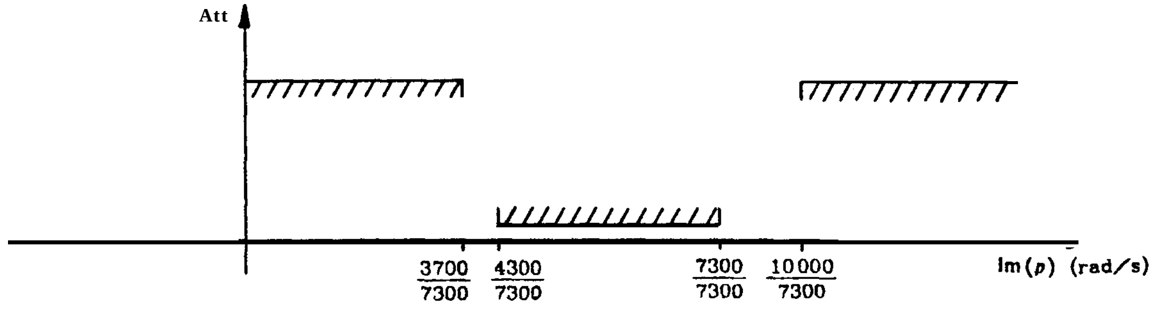


**Figure 9.7: Complex SSB Specifications Shifted to Positive Frequencies**

Next we pre-warp by taking $\Omega=\sqrt{\omega}$, and solve the resulting "real-filter" problem in $\tilde{s}$ using FILTOR2. The specifications and $8^{th}$ order transfer function resulting

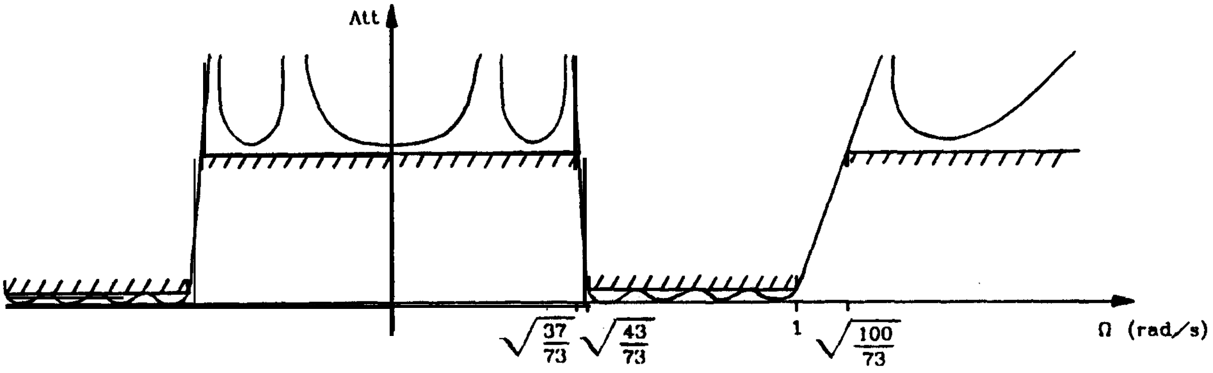


**Figure 9.8: Real Filter Prototype for SSB Prefilter**

**are** diagrammed in figure 9.0. We can now apply our frequency transformation to produce a complex filter solving the problem of figure 9.7: this is done simply by transforming roots $\tilde{e}_i$ according to the rule $e_i = \frac{\tilde{e}_i^2}{j}$. As a final step, the roots obtained may be shifted down in frequency to solve the original problem, as shown in figure 9.9.

All of this was done using the “real filter” approximator in FILTOR2 [10], and the results obtained are:

| polynomial | $p_s$ | $f_s(s)$ | $e_s$ |
|---|---|---|---|
| leading coefficient | 50.801 | 15988. | 5584.8 |
| roots: (normalized to 73OOHz) | -j.20518 | j.04911 | -.017533+j.O38052 |
| | -j.O51848 | j.12327 | -.06854+j. 10648 |
| | j.9011 | j.28484 | -. 10938+j.28667 |
| | ∞ | j.43072 | -.049279+j.45674 |

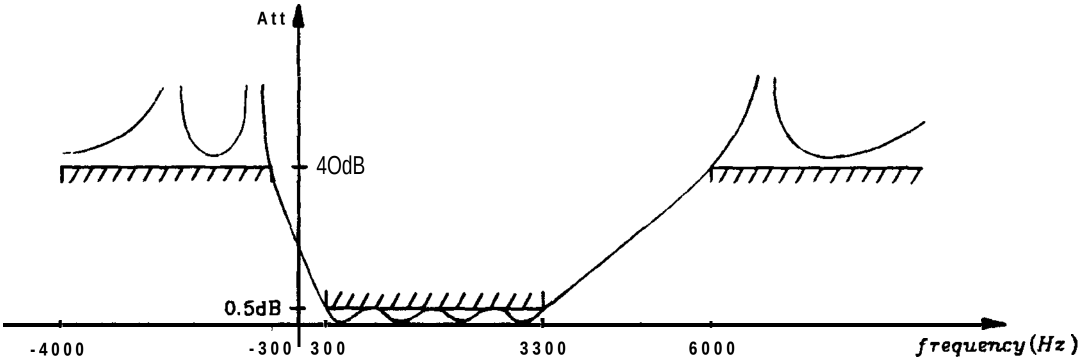


**Figure 9.9: Attenuation Function Meeting the Specifications of Figure 9.6**

### 9.2.3 Another Transformation

The scheme above required that we shift specifications up so that all specified points were at positive frequencies, because the real-to-complex transformation only attends to positive $\omega$. This section shows how to map any transition band onto the entire negative frequency axis, so that the complex-to-real transformation given above may be used to design arbitrary complex filters. In particular this transformation reduces any single-passband complex problem to a normalized lowpass.

If the lower transition band extends from, say, jc up to $ja$, the transformation

$$p \triangleq \frac{s-ja}{s-jc} \cdot \alpha j$$

maps that transition band onto the entire negative imaginary axis in p. The remaining portion of the specifications is mapped entirely onto the positive imaginary axis in $p$, and the lower passband edge $ja$ is mapped to $p=0$. The constant a may be used to do normalization: if the upper passband edge is at $s=jb$ it may be mapped to p = j by:

$$p=\frac{s-ja}{s-jc} \cdot j \cdot \frac{b-c}{b-a}$$

This maps any single-passband complex filter to one with a normalized positive-frequency passband. The transformation

$$p=\frac{\tilde{s}^2}{j}$$

presented above may then be used to obtain a lowpass “real filter” problem, which may then be solved by conventional means.

**Example**

**We** want a filter with a passband extending from DC to +3OOOHz and with stopbands extending up from 4OOOHz and down from -1OOOHz. Passband ripple is to be O.ldB and we want 40dB of stopband attenuation.

The parameters a,b and c above are respectively 0, +3OOOHz and -1OOOHz. Applying the transformation results in a specification with a passband from 0 to $j$ and a stopband from $\frac{16}{15}j$ on up. Applying the complex-to-real transformation then gives us a lowpass design problem with $\Omega_s=\sqrt{\frac{16}{15}}$. This is best solved with an elliptic filter of order 9, but odd order real filters correspond to complex filters with $\sqrt{s}$ terms in their frequency responses, so we choose a tenth-order prototype. This in turn will produce a fifth-order complex filter.

### 9.2.4 **A More Direct Approach**

Equiripple passbands are obtained in closed form for “real filters” by means of a mapping to a variable z:

$$z^2 \triangleq \frac{\tilde{s}^2+1}{\tilde{s}^2+\tilde{a}^2}$$

and we obtain our complex filters by means of another transformation

$$js \triangleq \tilde{s}^2$$

These two may be collapsed into a single transformation for the purpose of designing a complex filter. The result is simply

$$z^2 \triangleq \frac{s-jb}{s-ja}$$

which allows us to design a complex filter, given an arbitrary $p_s$, to have an ideal equiripple passband between $s=ja$ and $s=jb$.

This technique avoids the problem implicit above that we were first required to shift the problem to positive frequencies, and in fact provides us with the same power that the z transformation gives for “real filters”. With this and the generalization above of Feldtkeller’s equation approximation for complex filters becomes no harder than that for real filters.

Interpretation of the variable z in terms of the complex s-plane is a little different than that for the “real” i-plane. The main difference is that the area in z that maps to the real axis of $\tilde{s}$, and is therefore usually uninteresting, maps to negative frequency in s, which is very interesting.

“Pole-placers” for real filters often work in the variable $\gamma \triangleq ln(z)$, where the closed-form formulae for $f_s(s)$ are particularly convenient and numerical conditioning is good. The same variable 7 may be used for complex filters with the same result. This allows one to re-write a standard pole-placer for complex filters The different significance of the portion of z corresponding to negative frequency in s turns out to make the approximation problem in 7 look like a problem for a low-pass real filter.

Note that our transformation produces filters in z with twice the order of the final filter in s.

**Example**

Given $p_s \triangleq s+j$, obtain an equiripple passband between j0 and j 1.

Using the transformation $z^2 \triangleq \frac{s-j}{s}$ we get (after the fashion described in [Sedra78])

$$p_s(s)\overline{p_s}(-s)=(s+j)(-s-j)$$

transforming the root $s=-j$ to $z^2=2$

$$p_z(z)p_z(-z)=(z^2-2)^2$$

and choosing only LHP roots for $p_z$ so as to get the maximum number of ripples,

$$p_z(z)=(z+\sqrt{2})^2$$

$$=z^2+2\sqrt{2}z+2$$

and using the closed-form solution for equiripple $f_s(s)$

$$f_z(z)=Ev(p_z(z))$$

$$=z^2+2$$

and transforming the reflection ***zero*** $z=j\sqrt{2}$ back to $s$,

$$f_s(s)=c_f\,(s-\frac{j}{3})$$

where the leading coefficient contains the usual parameter $\varepsilon$ that sets passband ripple. Choosing $\varepsilon=1$, which gives about 3 decibels of passband ripple, gives $c_f$ =3 (obtained by setting$|\, k_s(0) \,|=\varepsilon$).

We can now use (9-7) to obtain $e_s$ and complete the design:

$$e_s\overline{e}_s(-s)=9(s-\frac{j}{3})(-s+\frac{j}{3})+(s+j)(-s-j)$$

$$=8(s+\frac{\sqrt{3}-j}{4})(-s+\frac{\sqrt{3}+j}{4})$$

$$e_s=2\sqrt{2}(s+\frac{(\sqrt{3}-j)}{4})$$

## 9.3 The Effects of Relative Gain Error

Complex filters implemented as two-output systems contain an inherent gain-matching requirement: that gain mismatches between the two outputs cause positive and negative frequencies to “leak” into one another. In analog technologies, where precise matching is often difficult to obtain, this is the critical effect that limits the performance of complex filters.

As an illustrative ***extreme*** case, turning one output of the system in figure 9.2 entirely off produces a single-output filter with positive and negative-frequency responses exactly conjugate: $t(j\omega)=\overline{t(-j\omega)}$.

We can derive a formula expressing this problem by adding an error term $\varepsilon t_{\mathrm{Re}}$ to a complex transfer function $t=t_{\mathrm{Re}}+jt_{\mathrm{Im}}$. First we must write $t_{Re,s}$ in terms of the complex $t_s$, which we can do by manipulating the definition

$$t_s(j\omega)=t_{Re,s}(j\omega)+jt_{Im,s}(j\omega)$$

and, taking the conjugates of poIynomia1 coefficients on both sides of the equation

$$\overline{t_s}=\overline{t_{Re,s}}-j\overline{t_{Im,s}}$$

but $t_{Re,s}$ and $t_{Im,s}$ have real coefficients, so $\overline{t_{Re,s}}=t_{Re,s}$ and $\overline{t_{Im,s}}=t_{Im,s}$. We therefore have

$$\overline{t_{Re,s}}(j\omega)=t_{Re,s}(j\omega)-jt_{Im,s}(j\omega)$$

so that we can add expressions for $t_s$ and $\overline{t_s}$ to get

$$t_{Re,s}(j\omega)=\frac{t_s(j\omega)+\overline{t_s}(j\omega)}{2}$$

We said in section 9.2.1 above that $\overline{t_s}(j\omega)=\overline{t_s(-j\omega)}$, so we can write $t_{Re,s}$ as a sum of positive- and negative- frequency terms as follows:

$$t_{Re,s}(j\omega)=\frac{t_s(j\omega)+\overline{t_s(-j\omega)}}{2}$$

Now that we have an expression for $t_{\mathrm{Re}}$ of a suitable form, let the real-part output of figure 9.2 suffer a gain error $\varepsilon$, i.e. create a new $\tilde{t}_{\mathrm{Re}}=(1+\varepsilon)t_{\mathrm{Re}}$. The resulting new complex transfer function is

$$\tilde{t}\triangleq\tilde{t}_{\mathrm{Re}}+j\tilde{t}_{\mathrm{Im}}$$

$$\tilde{t}_s(j\omega)=(1+\varepsilon)t_{Re,s}(j\omega)+jt_{Im,s}(j\omega)$$

$$=t_s(j\omega)+\varepsilon t_{Re,s}(j\omega)$$

$$=t_s(j\omega)+\frac{\varepsilon}{2}\left[t_s(j\omega)+\overline{t_s(-j\omega)}\right]$$

$$=(1+\frac{\varepsilon}{2})t_s(j\omega)+\frac{\varepsilon}{2}\overline{t_s(-j\omega)} \qquad (9\text{-}9)$$

Correlated gain errors in the two sides simply produce a gain change in $t_s(s)$ – which is relatively unimportant – but gain mismatch produces the “aliasing” term in $t_s(-j\omega)$ of (9-9). As an example of its effect, let us suppose that $t_{\mathrm{Re}}$ and $t_{\mathrm{Im}}$ become mismatched by 1%: then at some stopband frequency jw we will find $\overline{t(-j\omega)}$ – where - jw might be in the passband $(|t_s(-jw)|\cong 1)$ - added into $t_s(j\omega)\cong 0$ with a loss of only $20\log(.01/2)\cong 46$dB. This clearly limits stopband performance.

This problem is inherent to the structure of quadrature SSB modulators. The same gain matching error in the two multipliers or carriers of figure 9.1a will have the same effect. The effect is therefore well-known in SSB modulator design. If one chooses to tune out this leakage it suffices to adjust either carrier amplitude or gain on either channel of the complex filter.

Relative phase between the two outputs is also important: equation (9-9) can be applied just by taking $\varepsilon$ to be a complex value. For small phase errors $\varphi$, $\varepsilon \cong j\varphi$: thus a 1° (.017 radian) phase error allows $20\log(.017/2) \cong 42\text{dB}$ of leakage between pass- and stopband. Again the same care must be taken with the relative phase of the carriers in a quadrature SSB modulator.

## 9.4 Application of IF Synthesis

The intermediate-function synthesis method of this thesis may be applied directly to complex filter synthesis: we will show how with an example that synthesizes a first-order complex transfer function with two real-valued integrators.

Let us, for example, find a configuration of two (real-valued) integrators that realizes $t_s(s) \triangleq 1/(s+1-j)$ as a one-input two-output system. Let us further attempt to do this in such a way that the two system outputs are just the outputs of the two integrators: this will save the cost of output-summing networks.

We need to have transfer functions from the system input to the two integrator outputs of:

$$\mathbf{f}_1 = t_{\mathrm{Re}} \triangleq \frac{t_s(s)+\overline{t}(s)}{2} = \frac{s+1}{s^2+2s+2}$$

$$\mathbf{f}_2 = t_{\mathrm{Im}} \triangleq \frac{t_s(s)-\overline{t}(s)}{2j} = \frac{1}{s^2+2s+2}$$

Given $\mathbf{f}_1$ and $\mathbf{f}_2$ **we** can solve for system coefficients $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ by the methods discussed in chapters 2 and 6. The result is

$$A = \begin{bmatrix} -1 & -1 \\ 1 & -1 \end{bmatrix} \quad b = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

so that the system represented by the SFG in Fig. 9.10 would have the appropriate properties:

A circuit implementing this SFG (except for some sign changes to reduce complexity) is given in Fig. 9.11. One could, as discussed in chapter 3, use other circuits to implement either or both of these integrators: improved performance with regard to c++, for instance, could be obtained by using an Akerberg-Mossberg positive integrator [ 11]

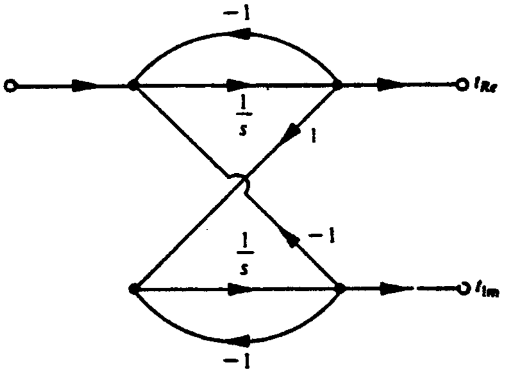


**Figure 9.10: SFG of First-order Complex Filter**

The {$g_i$} functions for this filter are:

$$g_1 = \frac{s+1+j}{s^2+2s+2}$$

$$g_2 = \frac{j\cdot(s+1+j)}{s^2+2s+2}$$

We use IF synthesis because it is easy to use it to construct and compare all of the popular topologies for filters and to invent new ones with particular properties. Thus, for example, it is of practical value, from the points of view of both economy and sensitivity, to be able to force $t_{Re}$ and $t_{Im}$ to appear at the outputs of integrators rather than as sums of outputs.

When combined with the set {$g_i$} the {$f_i$} functions may be used to measure sensitivity and dynamic range performance in complex filters just as in real ones. Thus, for instance, scaling may be accomplished by setting

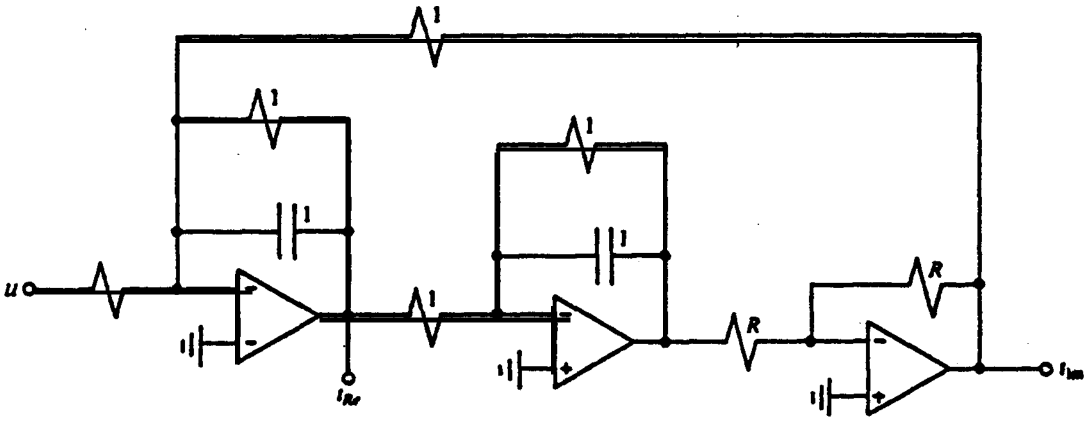


Figure 9.11: Circuit for First-order Complex Active Filter

$\|f_i\|=1 \quad \forall i$

## 9.5 Complex SFGs

Another tool useful in dealing with complex filters is the complex SFG. A complex SFG is one in which the transmittances may be complex, and thus has complex signals flowing throughout. In order to obtain a circuit implementation the complex SFG first has to be converted into an SFG whose branches have real-valued transmittances. This conversion process is done one branch at a time and will now be illustrated.

Consider a branch with a complex transmittance $(\alpha+j\beta)$ and let the input signal to this branch be $(a+jb)$. It follows that the output signal of this branch will be $(a\alpha-b\beta)+j(a\beta+\alpha b)$, which gives us the equivalence of Fig. 9.5 for converting complex SFG's to real ones:

where we have denoted*paths over which complex signals must flow by double lines.

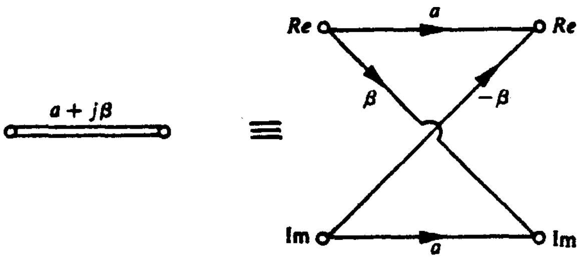


**Figure 9.12: Implementing a Complex SFG**

At this point it is worth noting that a matching requirement is implicit in the real SFG of Fig. 9.12. This is an indication of a general need for matching between the two ‘channels’ in a complex filter. Failure to achieve this matching results in aliasing between positive and negative frequencies. This causes ‘feedthrough’ from the passband into the stopband thus limiting stopband attenuation. To guard against this problem one seeks realizations which employ strong coupling between the two ‘channels’.

As an example of the application of complex SFGs consider again the synthesis of the first-order function$t_s(s) \triangleq \frac{1}{(s+1-j)}$. This function can be realized by an integrator (l/s) with **a** feedback path having a transmittance of **(-l+j).** The resulting complex SFG is shown in Fig. 9.13.

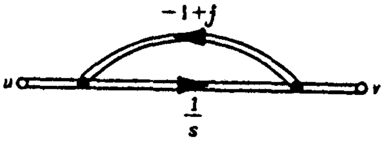


**Figure 9.13: Complex SFG of First-order System**

The conversion identity may now be used first for the integrator branch $(\alpha=\frac{1}{s},\beta=0)$ and then for the feedback branch $(\alpha=-1,\beta=+1)$. This results in the SFG of Fig. 9.3, which we previously obtained by more involved (but more general) means.

## 9.6 Three Possible Realizations

The techniques of the previous two sections have been applied to produce and compare several syntheses of the second-order transfer function given in [1]. This function is

$$t_s(s) \triangleq \frac{.062436(s+j.1495)}{(s+.16966-j1.056)(s+.10725-j.6124)}$$

and satisfies the specifications of Figure 9.14.

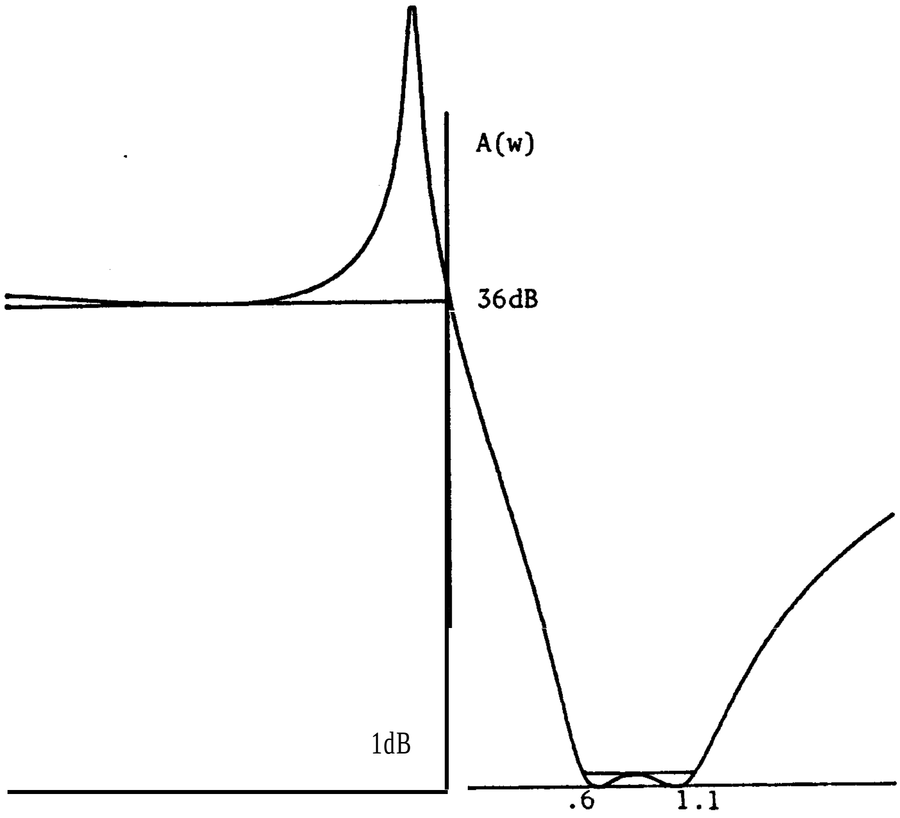


**Figure 9.14: Specifications and transfer function**

These syntheses include a cascade design, one formed by summing various outputs of a pole-forming network derived from an LC ladder, and a design derived from an LCX ladder similar to the type described by Humpherys [ 12] Details of

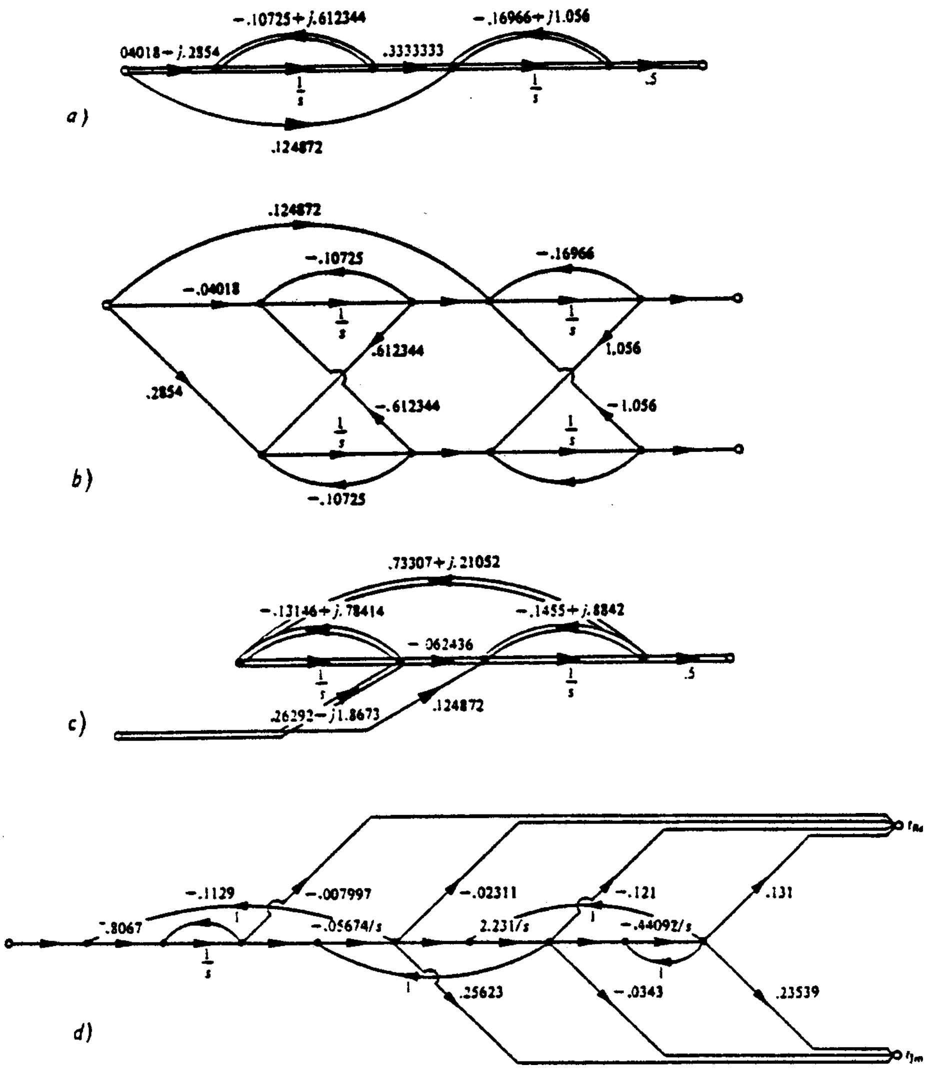


**Figure 9.15: Some Different Designs**

given in the following.

Figure 9.15 shows SFG's for these realizations, and Figure 9.16a is a plot of the worst-case integrator sensitivity $\max_i |S_{\gamma_i}^{t_s(j\omega)}|$ as a function of $\omega$ for each realization as a network of real integrators for positive frequency. This plot could be extended to negative frequency, where the stopband lies, but classical sensitivity is not the most useful measure for stopband performance because $t_s(s) \to 0$ in stopbands, so that $dt_s/t_s$ is badly behaved. Figure 9.16b shows $\max_i |\frac{dt(j\omega)}{d\tau_i} \cdot \tau_i|$ for negative frequencies, which is a more useful measure.

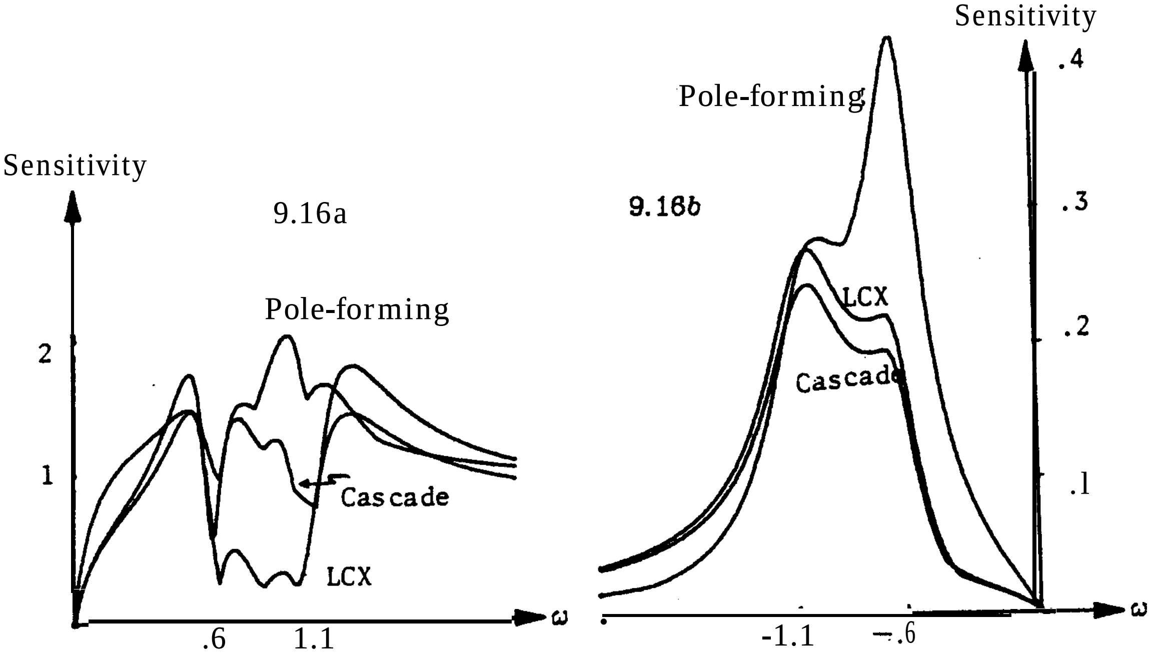


**Figure 9.16: Worst-case Sensitivities**

The better designs come quite close to a lower bound in the passband, but do relatively poorly in the stopband. This is a result of the aliasing problem previously discussed.

## 9.7 Complex Cascades

One may realize any complex transfer function as a cascade of first-order complex sections corresponding to bilinear factors of $t_s(s)$. Thus the complex SFG of figure 9.15a may be derived from the factorization

$$t_s(s)=\frac{.1872(s+j.1495)}{(s+.10725-j.6124)}\frac{.33333}{(s+.16966-j1.056)}$$

The real-valued SFG of figure 9.15b is in turn derived from this.

Complex cascade designs are conceptually even easier than real-valued cascades, because there is no need to use biquadratic sections: just bilinear ones. Of course each bilinear complex section will actually contain two real integratars, just like a conventional biquad.

## 9.8 LCX Filters

Humpherys [12] develops ladder synthesis for complex transfer functions in terms of three types of lossless elements: inductors and capacitors of purely real value and reactances whose impedances are purely imaginary and frequency-independent, which he called pure reactances, X. Because these elements allow synthesis of transfer functions with maximum power-transfer [13] between source and load one could hope to get good performance by analogy to the situation for real ladders [14].

We distinguish between two slightly different types of element, an ordinary “pure reactance” and an imaginary-valued resistor. A “pure reactance” has impedance

$$Z_s(j\omega)=\begin{cases} j & ,\omega>0 \\ -j & ,\omega<0 \end{cases}$$

while the impedance of an imaginary resistor is simpiy

$$Z=j.$$

Both types of element are lossIess but a pure reactance is a (non-causal but real-valued) element that could only be simulated with an infinite-order lumped system, while an imaginary resistor corresponds to coupling between two signal paths. We use X to denote an imaginary resistor.

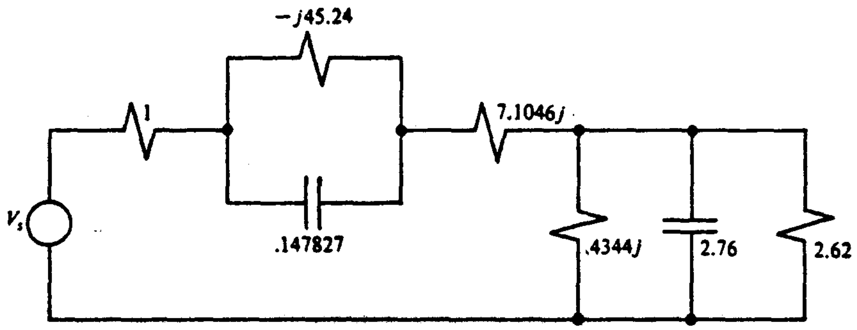


**Figure 9.17: LCX Ladder for Second-order Example**

An LCX ladder realizing the desired transfer function appears in figure 9.17. The operation of LCX ladders can be simulated using techniques similar to those employed for LC ladders (by the methods of [7] or the new ones of chapter 8). The difference from “real” filters is that in the LCX case complex SFG are ob taincd. The simulation of the LCX ladder in Fig. 9.17 was obtained by an extension ‘to complex signals of the techniques outlined in chapters 3 and 8: the transfer functions from the input to the real and imaginary parts of the voltages across the two capacitors were derived, scaled for dynamic range, and used as the four intermediate f functions. Note that, while the capacitors are real in value, they carry complex currents in the LCX ladder, and so have to be simulated with pairs of integrators. Although many other networks are possible the one in Fig. 9.17 has the output appearing across a capacitor. This means that the simulation of figure 9.15c has the output desired at the outputs of the pair of integrators which simulates this capacitor’s voltage, so that no output-summing network is needed.

## 9.9 Pole-forming LC Ladders

One can synthesize any desired transfer function using a state-variable simulation of an LC ladder with the desired natural modes by summing the outputs of the various integrators in the simulation with appropriate coefficients to get the required transmission zeros. If two transfer functions sharing the same natural modes are to be produced they may share the natural-mode forming network, and differ only in their output-summing networks. A design of this type is described in [l].

Any type of network could be used to form natural modes, for example cascade or companion forms. The purpose of basing a network on a ladder is to benefit from its good sensitivity near maximum power-transfer. In the following we show sensitivity for one design of this type, and discuss some observations on the technique in general.

## 9.10 Performance Comparison

We shall ROW compare the three realizations from the point of view of worst-case time-constant sensitivities. We ***use*** these because they are indicative of sensitivities in general and because they are not affected by details of implementation and therefore show up inherent properties of topologies.

We chose to look at a number of different implementations of **a** simple transfer function in order to demonstrate the options available. We would expect the differences among the various schemes to become ***more*** marked at higher orders.

The plots of Fig 9.16 clearly show the LCX simulation as having the best passband sensitivity, while it is slightly worse than the complex cascade design in the stopband.

The actual values of sensitivity are encouraging: passband sensitivity in the better realizations is less than 1.5, and the stopband sensitivity figure *is* about 0.25. The passband .sensitivity figure translates to a worst-case magnitude deviation of $20\log_{10}(1.015)\cong.13dB$ for a 1% time-constant (or integrator gain) error. The stopband figure is best interpreted as measuring “feed-through” that limits attenuation. The amount of this feed-through from the passband (for a 1% time-constant error with $\frac{d\,|t_s(s)|}{d\ln\tau}=.25$) is $20\log_{10}(.25\times.01)\cong-52dB$. This suggests that the required *36dB* stopband is easy to obtain

This stopband sensitivity figure is interesting because it shows that feedback within the good filters couples the two ‘channels’, thus desensitizing the transfer function to gain mismatch. Note that if one simply changes the gain constant of either the real or the imaginary-part channel by a factor a, one can expect (in the stopband) feedthrough from the passband of $d\frac{t_s(s)}{d\ln\alpha}=.5$. Changing the time-constant of either of the two integrators forming the outputs might be expected to do the same thing, since time-constants are just integrator gains: but the figure obtained is actually almost a factor of 2 better because feedback within the network tends to change the level of the other signals in the same direction.

This factor of 2 desensitivity illustrates the utility of being able to force the overall system outputs to occur at integrator outputs. This kind of flexibility is easy to get with the synthesis method of this thesis.

The pole-forming approach appears to be somewhat worse than the designs based on “complex integrators”. There are, however, many different ways to choose pole-forming networks, some of which might do better than the one we used. In particular, the advantage gained by choosing this LC-based network over, say, a cascade is probably illusory. The problem is that the transfer function of the prototype LC ladder is very far from being flat in the passband, so that over much of the critical range of frequencies the ladder is far from having maximum power-transfer to its output. It may be that choosing to simulate a prototype with transmission zeroes chosen to keep the passband near maximum transmission would produce better sensitivity. The method might, if its performance were improved, be competitive with complex cascades in applications where the number of components, in an LCX simulation is too large.

## 9.11 Comparison with Existing Art

If complex filters are to be used as broadband 90° phase shifters they will obviously have to be better than the conventional circuits using pairs of matched all-pass filters  We therefore compare the designs obtained thus far with an all-pass based design.

Phase-shifter designs may in fact be regarded just as special types of cornplex filters (cf. section 9.1.4), because they involve a pair of transfer functions $t_1$ and $t_2$ which we may conceive of as a single function $t \triangleq t_1+jt_2$ with complex coefficients. The conventional design, however, has natural modes only on the real axis, so that the class of complex transfer functions obtainable is very restricted. Since our sample function is not in this tiny sub-class, we cannot use the same complex $t_s(s)$ when investigating all-pass designs as we used earlier. We could therefore choose a function that meets the same specifications as the original as a basis for comparison: unfortunately an all-pass based design cannot even do that, because our sample filter has 36dB of attenuation down to DC while a pair of all-pass networks has low attenuation near DC. The only way to meet this specification with all-pass networks is to precede the phase-shifting pair with a conventional bandpass filter.  We therefore choose for comparison just the part of the transfer characteristic for which the all-pass portion would be responsible: obtaining an approximation to a 90° phase-shift over the passband (j0.6 to j 1.1). The resulting all-pass transfer functions are:

$$t_{Re,s}=\frac{s-l.9932}{s+1.9932}$$

$$t_{Im,s}=\frac{s-.33117}{s+.33117}$$

These may be implemented as a pair of first-order networks: note however that meeting the overall complex specification will require that the phase-shifting pair be preceded by a fourth-order “real” filter. This means that the number of .real-valued integrators needed to meet specifications with the conventional approach is 6, while any of the “complex” designs do it with 4.

Time-constant sensitivities for this approach appear in Figure 9.18, where they are compared with those for the LCX design.

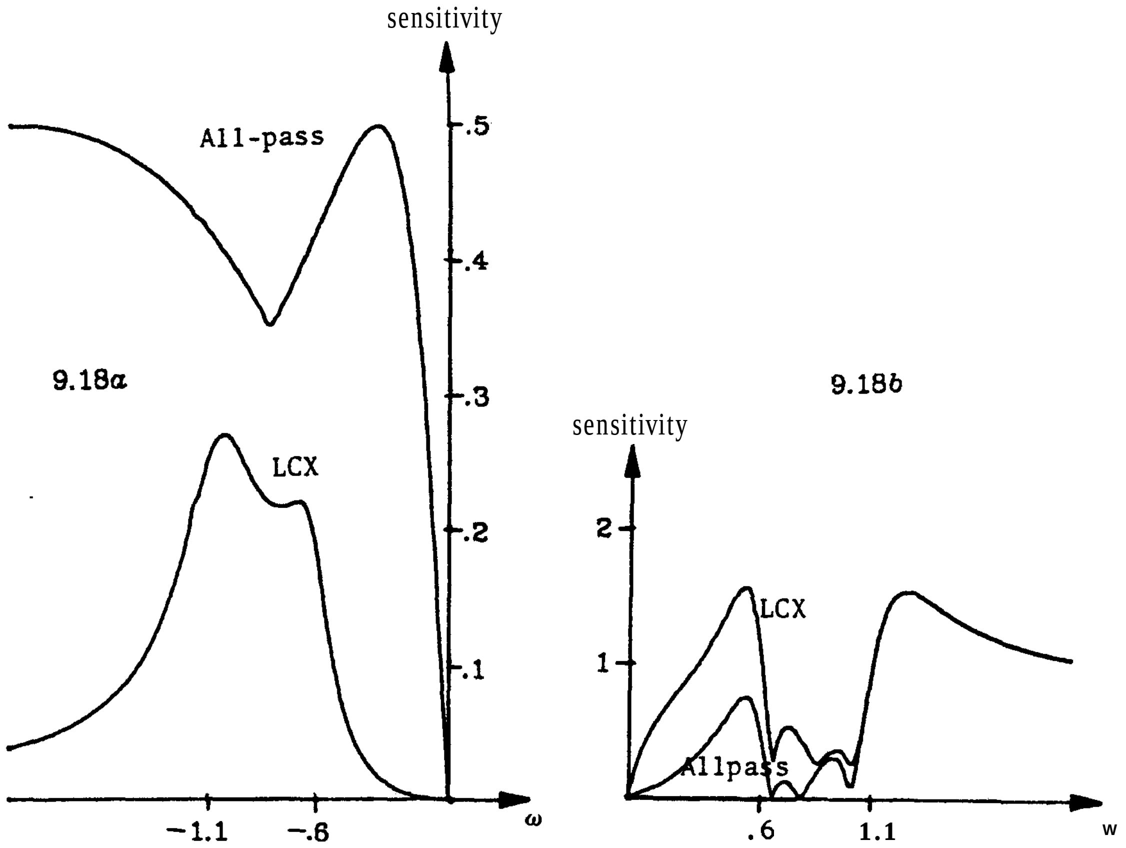


**Figure 9.18: Sensitivity Comparison; LCX vs. AP**

Nole that the all-pass design, already much worse than the LCX simulation, will also be sensitive to components of the “real filter” with which it is cascaded.

## 9. 12 On implementations of Complex Integrators

In section 9.5 above we discussed the implementation of “complex integrators”, i.e. circuits to integrate complex-valued signals, as pairs of ordinary real integrators. This section shows how the $\{f_i\}$ and $\{g_i\}$ of the ideal complex system are related to the iii] and $\{\tilde{g}_i\}$ of the “real” system obtained in this way. It also shows how a state-space description $\{\mathbf{A},\mathbf{b},\mathbf{c},d\}$ is converted.

The first-order complex synthesis example in sections 9.4 and 9.5 above will serve as an illustration: a complex synthesis could be done, resulting in

$$\mathbf{f}_{1,s}=t_s(s)=\frac{1}{s+1-j}$$

for which $\{\mathbf{g}_i\}$ would just be

$$\mathbf{g}_{1,s}=\mathbf{f}_{1,s}=\frac{1}{s+1-j}$$

Now we chose to implement the complex integrator responsible for $\mathbf{f}_1$ as a pair of real integrators producing its real and imaginary parts separately. Thus we have chosen

$$\tilde{\mathbf{f}}_{1,s}=\mathrm{Re}(\mathbf{f}_{1,s})=\frac{s+1}{s^2+2s+2}$$

$$\tilde{\mathbf{f}}_{2,s}=\mathrm{Im}(\mathbf{f}_{1,s})=\frac{1}{s^2+2s+2}$$

The effect of this on the A-matrix is to separate the complex entry $\mathbf{A}_{11}$ into a sub-matrix of four real entries:

$$\tilde{\mathbf{A}}=\begin{bmatrix}\mathrm{Re}(a_{11}) & -Im(a_{11})\\ \mathrm{Im}(a_{11}) & \mathrm{Re}(a_{11})\end{bmatrix}$$

This much just restates in terms of $\{\mathbf{f}_i\}$ and matrix entries what was given for SFGs in section 9.5.

Note that $\{\tilde{\mathbf{f}}_i\}$ should really be used in doing scaling, not $\{\mathbf{f}_i\}$, since the real integrators, not the fictitious complex ones, are the things that “clip”. The results of scaling the $\{\mathbf{f}_i\}$ will not, however, usually be much different. In particular, when a positive-pass transfer function is being implemented

$$\|\tilde{\mathbf{f}}_j\|_2 \cong \frac{\|\mathbf{f}_i\|}{2}$$

(where i is the complex integrator from which real integrator *i* was derived) because

$$\|\tilde{\mathbf{f}}_j\|_2 = \sqrt{\int_{-\infty}^{\infty} \left| \frac{\mathbf{f}_i(j\omega) \pm \overline{\mathbf{f}}_i(j\omega)}{2} \right|^2 d\omega}$$

$$\cong \sqrt{\frac{1}{4}\int_{-\infty}^{\infty} \left[ |\mathbf{f}_i(j\omega)|^2 + |\overline{\mathbf{f}}_i(j\omega)|^2 \right] d\omega}$$

(because for a positive-pass filter the regions over which $\mathbf{f}_i(j\omega)$ and $\overline{\mathbf{f}}_i(j\omega) = \overline{\mathbf{f}_i(-j\omega)}$ are significantly different from zero are disjoint)

$$= \frac{1}{2}\|\mathbf{f}_i\|_2.$$

Thus a filter whose complex signal levels have been scaled to avoid clipping is also scaled reasonably (though somewhat conservatively) for real integrators.

It remains, to find the $\{\tilde{\mathbf{g}}_i\}$, after which we will have everything we need to evaluate the performance of a real-integrator implementation of complex $\{\mathbf{f}_i\}$.

We could work with $\{\tilde{\mathbf{g}}_i\}$ computed either of two different ways: either as noise gains from the inputs of each integrator separately to each of the real and imaginary outputs – in which case we would need two sets of $\{\mathbf{g}_i\}$, one set for gains to each output – or as complex transfer functions from (real) integrator inputs to the notional “complex output”. We choose the latter because we are not interested separately in the real and imaginary parts of $t_s(s)$, but only in the (complex) overall result. Thus we have complex $\{\tilde{\mathbf{g}}_i\}$ while our $\{\tilde{\mathbf{f}}_i\}$ have real coefficients. This apparent asymmetry results from the assumption that our system has only *one* input terminal, i.e. has pure real inputs.

For the first-order example of section 9.4, the $\{\tilde{g}_i\}$ were

$$\tilde{g}_{1,s} = \frac{1}{s+1-j} = g_{1,s}$$

$$\tilde{g}_{2,s} = \frac{j}{s+1-j} = j\,g_1$$

In general, noise injected at the "real part" of a complex integrator sees the g function of the complex integrator while noise injected at the "imaginary input" sees "j times that gain to the complex output. This is because the effects are those of injecting purely real-valued and imaginary-valued noise to the original complex integrator.

## 9.13 Summary and Conclusions

We have presented some tools for dealing with the synthesis and analysis of complex analog filters. Using these techniques we have discussed three approaches to synthesis and compared the resulting realizations using a single-sideband filter example. The complex filter approach has also been compared to the traditional method of designing broadband *90'* phase-shifters.

It appears that much of the design art for "real filters" may be carried over directly to "complex" ones. In particular, one may use complex-valued signal-flow graphs to design complex filters in a straightforward way, and it appears that doubly-terminated lossless (LCX) filters designed for maximum power-transfer may be simulated to obtain very good passba'nd sensitivities. As with "real filters", cascade and ladder-based designs perform about equally well in the stopband.

The problem of matching gains, inherent in two-output complex filters, appears to be somewhat alleviated by designing such that integrator outputs form the required signals directly, because feedback can then tend to make the levels of these signals track one another.

This preliminary investigation has suggested that the complex filter concept may be used to design broadband phase-shifting networks that perform better than conventional all-pass systems.

## 9.14 References

[1] G.R. Lang and P.O. Brackett, "Complex Analog ***Filters"*** ECCTD 1981, Special Session on Active Filters, Aug. 1981

[2] W.M. Snelgrove and A.S. Sedra, ***"A Novel Synthesis Method for State-Space Active Networks*** Midwest Symp. Circuits and Systems, Toledo, Ohio, Aug. 1980 pp. 196-200

[3] B.P. Lathi, Signals ***Systems, and Controls,*** Intext Educational Publishers, NY, 1974

[4] J.L. Flanagan, ***Speech Analysis, Synthesis and Perception,*** 2nd Ed., Springer-Verlag 1972

[5] A.V. Oppenheim and R.W. Schafer, ***Digital Signal Processing,*** Prentice-Hall 1975

[6] D.E. Norgaard, ***"The Phase- Shift Method of Single- Sideband Signal Generation"*** Proc. IRE, vol. 44, pp. 1719-1735, Dec. 1956.

[7] A.S. Sedra and P.O. Brackett, Filter ***Theory and Design: Active and Passive*** Matrix Publishers, Champagne, Illinois, 1978

[8] T. Tsuchiya and S. Shida On ***the Design of Broad Band '90 degree' Phase- Splitting Networks*** IEEE Trans. Circuits and Systems, vol. CAS-27, pp. 30-36, Jan. 1980.

[9] W.F. McGee, *'Numerical ***Approximation Technique for Filter Characteristics tic Functions",*** IEEE Trans. Circuit Theory, CT-14, pp. 92-94, Mar. 1967

# 10. A New Technique for Error-Modelling

A new technique for investigating the performance of filters in the presence of errors is outlined. The technique is shown to make clear fundamental limitations on sensitivity performance for filter realizations and to allow a designer to take component tolerances into account at the approximation stage of filter design. It also suggests a novel way to simultaneously approximate attenuation and group delay specifications.

## 10.1 Introduction

One conventionally analyzes the behaviour of an imperfect realization of a filter design by computing sensitivity figures $S_{\mu_i}^{|t_s|} \triangleq \frac{\partial |t_s|}{\partial \mu_i} \cdot \frac{\mu_i}{|t_s|}$ for component or parameter values $\mu_i$ affecting performance.

A common intermediate step [ 1,2] in computing $S_{\mu_i}^{|t_s|}$ is to compute sensitivities of pole positions $(\omega_{o_j}, Q_j)$ to the $\mu_i$, and write

$$S_{\mu_i}^{|t_s|} \cong \sum_j \left( S_{\omega_{o_j}}^{|t_s|} S_{\mu_i}^{\omega_{o_j}} + S_{Q_j}^{|t_s|} S_{\mu_i}^{Q_j} \right)$$

This kind of scheme is used, primarily for realizations using biquadratic sections, because $S_{\omega_o}^{|t_s|}$ and $S_{Q}^{|t_s|}$ depend only on the transfer function to be implemented, while $S_{\mu}^{\omega_o}$ and $S_{\mu}^{Q}$ depend on the particular circuit.

Our method, by contrast, models the effects of component inaccuracies by defining a frequency transformation that would have the same effect. Thus, to take a simple example, we would model the effect of a Q-enhancement phenomenon not by investigating the effect of natural modes moving right (towards the $j\omega$ axis) but by deeming the $j\omega$ axis to have moved left towards the natural modes. In this way, the actual frequency response of a filter can be evaluated using the nominal poles and zeroes along a “deformed” frequency axis – some curve, nominally $j\omega$, whose exact shape is determined by component errors.

The principal advantage of this peculiar inversion is that it enables us to state a simple variant on the conventional approximation problem for filters that allows one to design transfer functions so that their implementations will meet specifications even in the presence of components with errors (of known tolerances).

As a side-effect, the method may be used to design for minimum group delay and a simple variant of it may be used to approximate flat delay.

The following sections of this chapter develop this idea further. They show how one may regard each non-ideal integrator in a filter as defining a (slightly deformed) version of the s-plane, and how a good filter tends to average these deformations. The nature of the deformation induced by some practical circuit effects is discussed, and design criteria are developed. Some points are made concerning the new type of approximator required.

## 10.2 Deformation of the s-plane

This section develops a formal basis for the idea introduced above.

Errors in circuit components are usually modelled as changing an ideal transfer function $t_s$ to a perturbed function $\tilde{t}_s$. Instead of this we define a new frequency variable $\tilde{s}=\psi_s$ so that

$$\tilde{t}_s(s)=t_s(\tilde{s}) \tag{10-1}$$

The meaning of this is illustrated in figure 10.1, where our method is compared to the usual one.

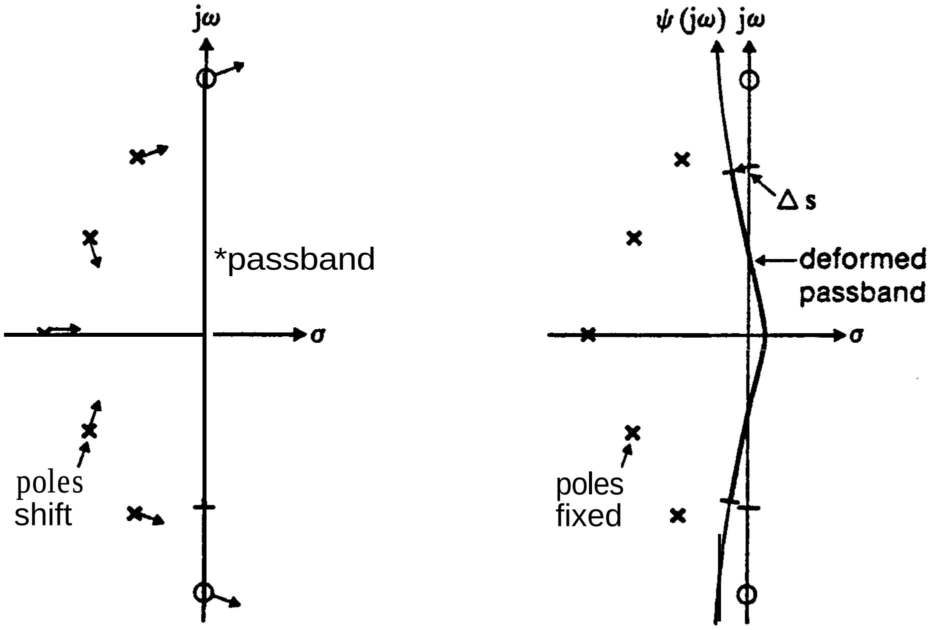


**Figure 10.1: Comparison of Methods**

Instead of investigating the effects (at ordinary physical frequencies $s=j\omega$) of changing $t_s$ to $\tilde{t}_s$ we wish to look at the *original* function $t_s$ at frequencies $\tilde{s}=\psi(j\omega)$. Equation (10-1) says that we have to choose the frequency-transforming function + so that the transfer function values we obtain are the same for both methods. The mapping $\psi$ will, of course, be a function of the circuit topology and of its components.

We will usually be concerned with the case of small errors, where $\tilde{s}\cong s$, although the technique ha▪ some applications in studying large-change performance.

Some mathematical points should be made about the transformation $\psi(s)$ needed to produce $\tilde{s}$. Firstly (concerning existence) there might not be any function that does what we want; and secondly there might not be a unique one.

We will show a way to define a suitable mapping that does what we want in the practical filter case. The mapping will exist for the kinds of $t_s$ encountered as filters, and $\tilde{s}$ approaches s for small component errors as we should expect.

For small errors, we can define the derivative of $\tilde{s}$ with respect to a component $\mu$ as

$$\frac{\partial \tilde{s}}{\partial \mu} \triangleq \frac{\frac{\partial t_s}{\partial \mu}}{\frac{\partial t_s}{\partial s}} \tag{10-2}$$

It *is* inelegant to have to define $\tilde{s}$ by referring to an implementation (i.e. using the effect of errors $\partial\mu$) and we will presently show examples for which this is unnecessary. Equation (10-2) is, however, useful because it relates analysis in terms of $\tilde{s}$ to the conventional approach, which studies $\frac{\partial t}{\partial \mu}$.

Equation (10-2) obviously blows up when the denominator $\frac{\partial t_s}{\partial s}=0$. This means, for instance, that $\tilde{s}$ will not exist when $t_s$ is a constant, in which case the derivative disappears. For practical filters, zeroes of the derivative do not appear on $s=j\omega$[†] and $\tilde{s}$ is well defined by (10-2) where it matters.

## 10.3 Modelling integrator gain errors

It is especially straightforward to apply the technique of this chapter to modelling the effects of non-idealities on integrator circuits. We are particularly interested in the effects of errors in the gains of the n integrators in a state-variable realization of $t_s$, because these tend to be the least well-controlled

[†] At reflection zeroes $\frac{d\,|t(j\omega)|}{d\,\omega}=0$, but this is only the derivative of the *magnitude*. The complex derivative (which includes phase) is non-zero.

components in high-performance filter applications. In active filter technology this means that we will model RC time-constant errors as well as the effects of finite amplifier bandwidth. The results can be directly applied to LC ladders, where reactive components correspond one-to-one with integrators in a state-variable realization (e.g. [3]). For the LC case this technique may be used to model the effects of non-zero dissipation factors (i.e. finite Q's) and small errors in component values.

In order to investigate the effects of integrator gain errors we will define (by analogy to the definition for filters in general) $n$ new frequency variables $\tilde{s}_i=\psi_i(s)$ that model the frequency responses of the n "integrators". Thus, for instance, if integrator number 1 has gain $(1/1.01s)$ instead of its nominal l/s we shall say $s_1=1.01s$. As a more interesting example of a transformation $\psi(s)$, a Miller integrator with nominal gain $1/sCR$ implemented with an operational amplifier with gain $\omega_t/s$ has

$$\tilde{s}_i=s(1+\frac{1}{\omega_t CR})+\frac{s^2}{\omega_t} \qquad (10\text{-}3)$$

Figure 10.2 shows the locus in the s-plane to which this mapping moves the physical-frequency axis, i.e. plots $\tilde{s}_i=\psi_i(j\omega)$, for the normalized case CR=1 and $\omega_t=20$. Figure 10.2 also shows an error term $\Delta s\triangleq\tilde{s}_i-s$ for a particular $s=j\omega$. The second-order term in (10-3) causes the Q-enhancement effect for which these integrators are known: in our terms it shifts the "physical-frequency" line $\tilde{s}=\psi(j\omega)$ to the left of the j-axis towards natural modes.

These $\psi_i(j\omega)$ contours have exactly the same physical interpretation we discussed above for the overall filter: one may obtain the gain of "integrator" i at a frequency $j\omega$ by evaluating the ideal integrator function l/s at the corresponding point on the curve $\psi_i(j\omega)$. These $\psi_i(j\omega)$ contours are easy to estimate; we will show later how to estimate the overall $\tilde{s}$ from them.

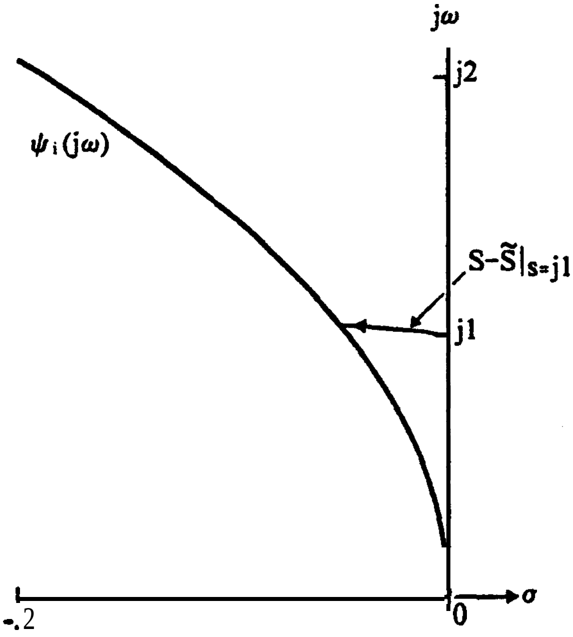


**Figure 10.2: The Effect of Finite Op-Amp Bandwidth**

As an even more extreme case, one that would be quite difficult to analyze by conventional methods, figure 10.3 plots the contour that results from implementing a Miller integrator with an op-amp with the second-order transfer function $\frac{-100}{s^2+s/10+1}$.

Note that the deformed and ideal frequency-axes are very close near $s=j1$ (because the amplifier gain is high there). This suggests that an amplifier with this "peaking" gain function would be quite useable for implementing a narrow-band filter around $s=j1$, where As is quite small.

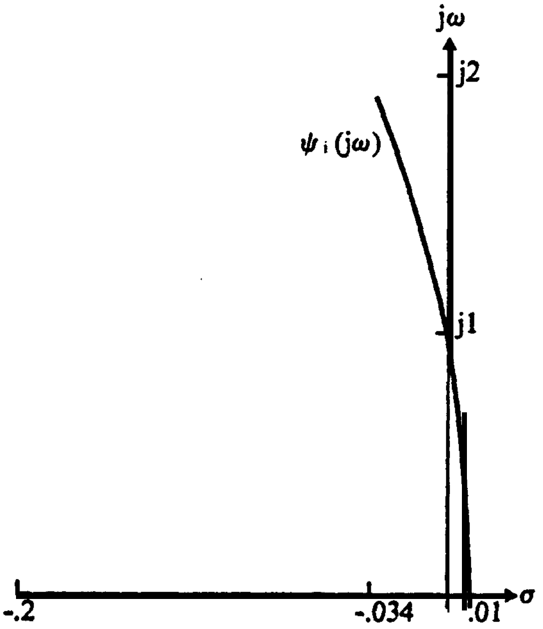


Figure 10.3: Implementing Filters with Peaky-Gain Amplifiers

## 10.4 A Second-order Analysis Example

Figure 10.4 is a signal-flow graph for a second-order lowpass filter with equal sensitivity to its two integrators (i.e. is FSLB, cf. chapter 7), a condition which we have shown (in chapter 7' and[4]gives very strong type of optimum. It is shown as having two "integrators" with gains $\frac{1}{s_1}$ and $\frac{1}{s_2}$, which would be exactly equal to $\frac{1}{s}$ in an ideal filter. Analyzing the SFG without assuming these integrators ideal we find that it has a transfer function $t=\frac{\omega_o^2}{\tilde{e}}$ where

$$\tilde{e}=\tilde{s}_1\tilde{s}_2+\frac{\omega_o}{2Q}(\tilde{s}_1+\tilde{s}_2)+\omega_o^2$$

and $\tilde{s}_1$ and $\tilde{s}_2$ may be arbitrary functions of s. For the ideal filter $\tilde{s}_1=\tilde{s}_2=s$ and so

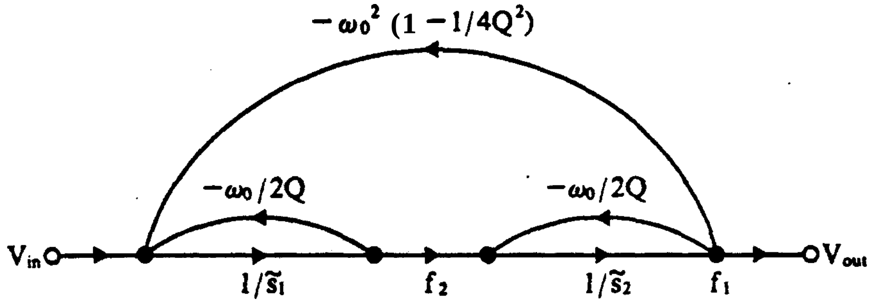


**Figure 10.4: Second-Order Lowpass, Non-ideal Integrators**

e $(s)=s^2+\frac{\omega_0}{Q}s+\omega_0^2$. Writing the equation $\tilde{e}(s)=e(\tilde{s})$

$$\tilde{s}_1\tilde{s}_2+\frac{\omega_0}{2Q}(\tilde{s}_1+\tilde{s}_2)+\omega_0^2=\tilde{s}^2+\frac{\omega_0}{Q}\tilde{s}+\omega_0^2$$

and solving for $\tilde{s}$ in terms of $\tilde{s}_1$ and $\tilde{s}_2$ gives

$$\tilde{s}=\frac{-\omega_0}{2Q}\pm\sqrt{\left(\tilde{s}_1+\frac{\omega_0}{2Q}\right)\left(\tilde{s}_2+\frac{\omega_0}{2Q}\right)} \qquad (10\text{-}4)$$

For reasons to be cxplained in the next paragraph, we prefer to choose the positive sign for the square root.

This example illustrates two things: firstly that $\tilde{s}$ is **a** kind of average of $\tilde{s}_1$ and $\tilde{s}_2$ (to be exact, $\tilde{s}+\frac{\omega_0}{2Q}$ is the geometric average of $\tilde{s}_1+\frac{\omega_0}{2Q}$ and $\tilde{s}_2+\frac{\omega_0}{2Q}$); and secondly that, while the choice of $\tilde{s}$ is not unique for this second-order $t_s$ because the choice of sign for the square-root is arbitrary, there is nonetheless a “natural” choice (the positive sign) that has $\tilde{s}=s$ when $\tilde{s}_1=\tilde{s}_2=s$.

## 10.5 Averaging Deformations

One may analyze the integrators of a filter as described in chapter 4 and section 10.3 to estimate their $\tilde{s}_i$: we next need to know how these will affect the overall $\tilde{s}$. In general the relationship is a function of the filter structure and transfer function chosen. We will show that the best structure conceivable (FSLB) has a strong “coupling” property, so that changing any one integrator is equivalent to changing them all (by a smaller amount).

The most important point about the relationship between $\tilde{s}$ and the n $\tilde{s}_i$ is this: that in a good filter structure $\tilde{s}$ tends to be an average of the $\tilde{s}_i$. This averaging is depicted in Figure 10.5, where some exaggerated $\psi_i(j\omega)$ contours are shown together with the resulting $\tilde{s}=\psi(j\omega)$ for a good filter structure. One may observe that this $\tilde{s}$ contour is some sort of an average of the $\tilde{s}_i$ contours.

Note that $\tilde{s}$ doesn’t wander further away from ideality than do the $\tilde{s}_i$; in fact the averaging effect tends to make $\tilde{s}$ better than the $\tilde{s}_i$ because their errors partially cancel. That averaging gives rise to performance improvement is familiar from statistics: by taking an average of several estimates of $s$, all somewhat corrupted by “measurement error”, we may expect to get closer to the “true value” than the individual $\tilde{s}_i$ do.

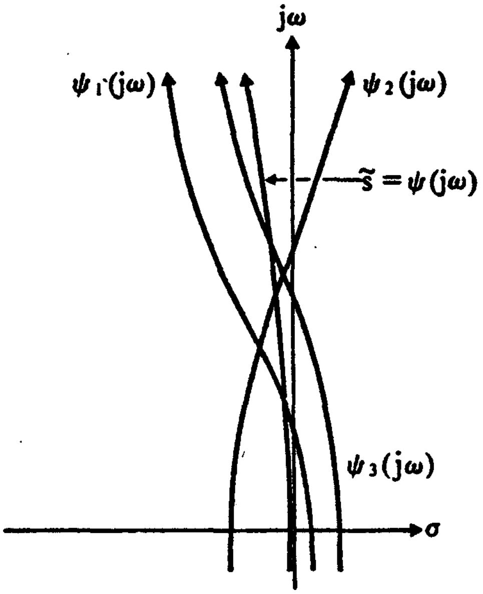


Figure 10.5: Averaging Gain Errors

We already saw, for a second-order case, how averaging came about. The type of closed-form relationship obtained there is not usually available for higher-order filters. As a substitute, we can look at a small-change formulation that applies at arbitrary order.

The formula

$$\Delta s \cong \sum_i \frac{\partial \tilde{s}}{\partial \tilde{s}_i} \Delta s_i \tag{10-5}$$

relates a small change As for an entire filter to the $\Delta s_i$ of its component integrators, which we can readily estimate. We are therefore interested in the derivatives $\frac{\partial \tilde{s}}{\partial \tilde{s}_i}$.

To see how these derivatives perform for the simple second-order case, we may differentiate (10-4) with respect to $\tilde{s}_1$ to find that

$$\frac{\partial \tilde{s}}{\partial \tilde{s}_1} = \frac{1}{2} \cdot \frac{\sqrt{\tilde{s}_2 + \frac{\omega_0}{2Q}}}{\sqrt{\tilde{s}_1 + \frac{\omega_0}{2Q}}}$$

A similar expression gives $\frac{\partial s}{\partial \tilde{s}_2}$, and when $\tilde{s}_1 \cong \tilde{s}_2$ both expressions reduce to

$$\frac{\partial \tilde{s}}{\partial \tilde{s}_i} \cong \frac{1}{2.} \tag{10-6}$$

In the general case an “averaging” like

$$\tilde{s} \cong \frac{1}{n} \sum \tilde{s}_i$$

would give $\frac{\partial \tilde{s}}{\partial \tilde{s}_i} = \frac{1}{n}$. Note that we were able, in our second-order example, to attain this perfect coupling. While one cannot generally do this well, good filter structures often approach this performance. In particular (cf. chapters 7 and

8), over their passbands, lossless filters designed for maximum power-transfer between input and output [5,6,7] come fairly close to this limit. Cascade designs are not as good, since at frequencies near their singularities their performance is dominated by single biquadratic sections, so that they average $\tilde{s}$ over only two $\tilde{s}_i$: this may be understood as meaning that they "weight" the averaging described by equation (10-5) in a Iess-than-optimum **way.**

If we plan to use a type of realisation that "averages" well, we can make an estimate of $\Delta\tilde{s}$ from our knowledge of the expected behaviour of the $\tilde{s}_i$. We can either take a "worst-case" approach, saying that $\Delta\tilde{s}$ will be no worse than the $\Delta\tilde{s}_i$, or a less conservative statistical approach that suggests that the variance of $\Delta\tilde{s}$ will be almost $\sqrt{1/n}$ times lower than that of the A$. Either approach gives us an estimate of the area in the $\tilde{s}$-plane to which we can expect $\psi$ to map an area (e-g. a passband) in the s-plane. We will use this estimate in section 10.7.

## 10.6 Application I: Sensitivity Bounds

This section introduces a simple application of the method of this chapter. We show that an "averaging" represents the best possible relationship between $\tilde{s}$ for a filter and the $\tilde{s}_i$ of its component integrators. This section gives an alternate way to look at the FSLB condition of chapter 7. An important identity concerning the $\tilde{s}_i$ is this:

$$\sum_i \frac{\partial\tilde{s}}{\partial\tilde{s}_i} = 1 \qquad (10\text{-}7)$$

that the sum of all integrator sensitivities is 1 regardless of implementation. This comes about because the effect of a uniform gain change in all integrators must be a simple frequency shift, since terms in s in an expression for the transfer function of a system of integrators and frequency-independent gain elements come only from integrator gains $\tilde{s}_i$. Thus the effect of putting $\tilde{s}_i=s+\delta \; \forall \, i$ is to give $\tilde{t}(s)=t\,(s+\delta)$ and $\tilde{s}=s+\delta$. Putting $\Delta s_i=\delta$ in (10-5) then gives

$$\Delta s=\delta=\sum_i \frac{\partial \tilde{s}}{\partial \tilde{s}_i}\Delta s_i$$

$$=\delta\sum_i \frac{\partial \tilde{s}}{\partial \tilde{s}_i}$$

$$=> \sum_i \frac{\partial \tilde{s}}{\partial \tilde{s}_i}=1$$

This corresponds to an identity in passive network theory concerning the sensitivity of RLC networks to their inductors and capacitors [8]:

$$\sum_i \mathrm{S}_{L_i}^{t_s}+\sum_i \mathrm{S}_{C_i}^{t_s}=\frac{s}{t_s}\frac{dt_s}{ds} \qquad (10\text{-}8)$$

In the RLC case the inductor and capacitor values set integrator gains -- because they multiply s in their immitances. Equation (10-5) and the definition of classical sensitivity,

$$\mathrm{S}_{\mu}^{y}\triangleq\frac{\mu}{y}\frac{dy}{d\mu},$$

may be used to relate (10-7) to (10-8).

The interesting thing about (10-7) is that it suggests an optimality condition for sensitivity. All topologies for filters respond in exactly the same way when all of their integrators are simultaneously disturbed by the same amount; systems differ from each other in how they react when integrators change different amounts. For this reason figures of merit like

$$\left[\sum \left|\frac{\partial \tilde{s}}{\partial \tilde{s}_i}\right|^p\right]^{1/p} \qquad 1 \leq p \leq \infty \tag{10-9}$$

may be useful in measuring the aggregate effect of uncorreIated random errors in $\tilde{s}_i$. These aggregate measures may be related in the better-known case of classical sensitivity analysis to worst-case $(p=\infty)$ and statistical [9] figures of merit.

The problem of optimizing measures of form (10-9) under a constraint like (10-7') has a simple solution: that all sensitivities should be equal, i.e.

$$\frac{\partial \tilde{s}}{\partial \tilde{s}_i} = \frac{1}{n} \tag{10-10}$$

This condition implies a very strong type of optimality, because many different and important measures of integrator sensitivity are simultaneously attained. UnfortunateIy it is not always possible to obtain (10-10) although we did so in the second-order example above and in fact can for most second-order functions [4].

It also appears that simulations of doubly-terminated lossless ladders designed for maximum power-transfer quite often come fairly close to (10-10) in the passband.

## 10.7 Application II: Approximation

We showed in section 10.5 how to estimate the sizes of the regions in the s-plane over which the passbands and stopbands might wander because of errors modelled by $\tilde{s}(s)$.

Figure 10.6 illustrates these regions for a low-pass filter together with a contour plot of a particular transfer function.

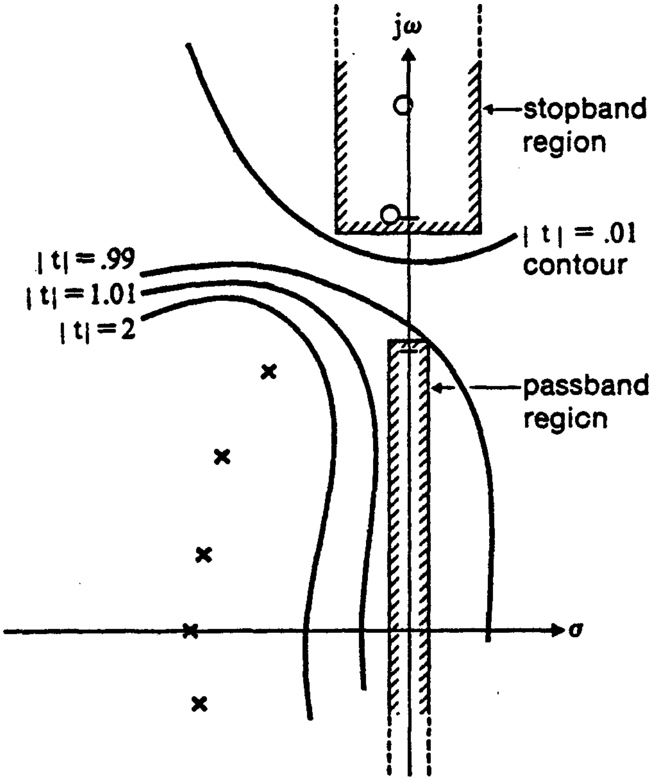


**Figure 10.6: Specification Regions and a Transfer Function**

The contour plot shows lines of constant magnitude for a transfer function, and in particular shows some.lines for magnitudes that one might specify as limits to passband and stopband attenuation. This diagram shows the problem to be solved if a transfer function is to be designed to be tolerant of component errors: a rational function $t_s$ must be found that meets a passband specification (e.g. $.99 \leqq | t_s | \leqq 1.01$) inside the "passband region" and that meets a stopband specification (e.g. $|t_s| < .01$) everywhere in the "stopband region". This function will have to enclose the passband and stopband regions entirely within the magnitude contours ($| t_s | = .99$ etc.) that define the limits of acceptability, because $\tilde{s}$ in the practical filter might wander anywhere inside those regions.

Specifications are thus expressed by "boxes" in the s-plane whose heights and widths are determined by tolerances. on $\tilde{s}(s)$ and whose depths are determined by attenuation specifications.

Natural modes of $t_s$ cannot be allowed into the specification “boxes” because at a natural mode $|t_s| = \infty$ and specifications are certainly not met*. Zeroes of $t_s$ must also be kept out of the “passband” regions.

Once poles of $t_s$ are banned from the specification boxes, we have the useful fact that $t_s$ is analytic within these regions; and because zeroes may not lie in passbands $\frac{1}{t_s}$ is analytic in the passband region. The maximum modulus theorem [10] may then be invoked to show that maximum and minimum passband region values always appear on the boundary of the passband region and similarly that maximum ‘stopband magnitudes (i.e. minimum stopband attenuation levels) appear on the stopband boundary. Since these maxima and minima are the values normally of interest, this means that an approximation program only need force specifications to be met on the boundaries of specification bands and they will automatically be met everywhere inside.

Another interesting side-effect of analyticity in the passband region may be seen by investigating $\ln(t_s) \triangleq \alpha + j\varphi$. An approximation solving the pr’oblem of constraining gain variation $-\Delta\alpha-$ over a region in s of finite width Au (where s $\triangleq \sigma + j\omega$) obviously controls $\frac{d\alpha}{d\sigma}$ (and of course this becomes more precise as $\Delta\alpha$ and $\Delta\sigma$ tend to zero). Now the Cauchy-Riemann equations [10] for the function $\ln(t_s)$, which is analytic in thepassband region, give

$$-\frac{d\alpha}{d\sigma} = -\frac{d\varphi}{d\omega}$$

but now the right-hand term is simply $\tau$, the group delay of $t_s$! Thus by designing a filter to have gain variation less than$\Delta\alpha$ over a region of width$\Delta\sigma$ we simultaneously constrain the group delay to about $\frac{\Delta\alpha}{\Delta\sigma}$

* In fact they should be kept to the left of any possible $\tilde{s}(j\omega)$ even in transition bands, since otherwise filters could be unstabIe.

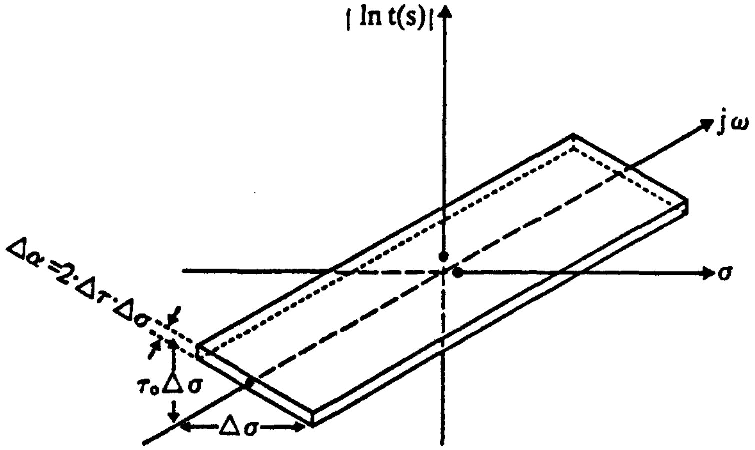


**Figure 10.7: Tilted Passband Specification**

The problem of designing filters to simultaneously meet attenuation and group delay specifications is an old one (e.g. [11]). Group delay may in practice either be specified to stay below a given maximum, in which case the preceding approach may be used, or to approximate an arbitrary constant to within a stated tolerance, in which case a minor variation is required. By allowing the passband box to "tilt" so that $\alpha=-\tau_0\sigma\pm\Delta\alpha$ one may approximate a constant delay $\tau_0$ with variation $\frac{\Delta\alpha}{\Delta\sigma}$. A three-dimensional view of a tilted passband specification box is presented in figure 10.7. The box is $\Delta\alpha=2\Delta\tau\Delta\sigma$ "deep" to allow slope variation of $\pm\Delta\tau$, and has an overall slope $\tau_0$. In this case, unfortunately, some of the sensitivity advantage of this kind of approximation is lost because a may vary by an additional amount $\tau_0\Delta\sigma$ across the specification box. A general approximator might best be designed to allow an adjustable maximum amount of tilt $\tau_0$ together with a specified $\Delta\alpha$ so that attenuation, group delay and sensitivity specifications could all be dealt with simultaneously.

## 10.8 Summary

We have presented a new theoretical tool for investigating the performance of systems implementing linear transfer functions in the presence of errors and non-idealities in circuit elements. We have applied this method to obtain new insight into the behaviour of integrators, a second-order structure, and a sensitivity bound. We have also used it to derive a novel formulation of the filter approximation problem that accounts for the important problems of finite tolerance and group delay at the very first step of filter design.

## 10.9 References

[1] G.S. Moschytz, A Note on ***pole Frequency and Q Sensitivity*** IEEE J. Solid State Circuits, Vol SC-6, pp.267-269, Aug 1971

[2] A.S. Sedra and P.O. Brackett, ***Filter Theory and Design Active and Passive*** Matrix Publishers, Champagne, Illinois, 1978

[3] P.O. Brackett and A.S. Sedra, Direct SFG ***Simulation of LC Ladder Networks with Applications to Active Filter Design,*** IEEE Trans. Circuits and Systems, vol. CAS-23, pp. 61-67, Feb. 1976

[4] Snelgrove, W.M. and Sedra, A.S. ***State- Variable biquads with optimum integrator sensitivities,*** IEE Proc., Vol. 128, Pt. G, No. 4, August 1981 pp. 173-175

[5] H.J. Orchard, ***Inductorless Filters*** EIectron. Lett. vol. 2, pp.224-225, June 1966

[6] Moschytz, G.S., ***Linear Integrated Networks,*** Van Nostrand Reinhold 1975

[7] G.C. Temes ***First- Order Estimation and Pre- Correction of Parasitic Loss Effects in Ladder Filters*** IRE Trans. Circuit Theory, vol. CT-9 pp. 385-400 Dec. 1962

[8] M.L. Blostein Some Bounds on ***Sensitivity in RLC Networks*** Proc. First Allerton Conf. on Circuit & System Theory, Urbana, Ill., pp. 488-501, 1963

[9] Laker, K.R. and Ghausi, M.S,, ***Statistical Multiparameter Sensitivity – A Valuable Tool for CAD"*** IEEE Int. Symp. Circuits and Systems, Newton, Mass., April 197'5, pp. 333-338

[10] Daniels, R.W., ***Approximation Methods for Electronic Filter Design, with Applications to Passive, Active, and Digital Networks*** McGraw-Hill 1974

# 11. Summary and Conclusions

We have introduced and developed a new approach to the design of anaIog circuits "intermediate-function synthesis". This approach *derives* circuit topologies from the behaviour that is required of them, whereas it has been conventional to start with a circuit structure and try to analyze it for its performance. Intermediate-function synthesis has been applied to the analysis, synthesis, and design of filters. It has produced good results.

## 11.1 Summary

A novel approach to filter design was presented in chapter 2, and following chapters developed the central idea of vector-space oriented design towards a practical design technique.

A link between abstract state-variable systems and the dominant practical problems of filter circuits was forged in chapters 3 and 4. Chapter 8 provided a detailed design example on a medium-order problem.

In order to be useful, a filter design technique must be suited to design automation. Intermediate function synthesis is ideal for computer-aided design because it is general enough to allow a designer to use a single language to describe a wide variety of circuits and circuit problems while simultaneously admitting of concrete enough interpretation that good models of important phenomena (like sensitivity and dynamic range performance) tend to be straightforward. Chapter 6 developed a useable set of algorithms that have been incorporated into a preliminary version of a design program.

A powerful design technique should readily adapt itself to variations in the problem to be solved. We used the "IF synthesis" technique to investigate "complex filters" in chapter 9, and obtained good results. In the course of research on this problem new structures fundamentally better than current art were developed.

A new approach to a problem should produce new insights. Chapter I developed a new way of looking at, and designing for, good sensitivity performance; chapter 5 developed some “geometric” ideas about the filter performance problem; and chapter 10 suggested a new way to do transfer-function approximation that tackles the difficult problems of designing *ab initio* for good sensitivity performance and of meeting simultaneous specifications on magnitude and delay.

## 11.2 Summary of Results

The main result of this work is the design approach itself: nevertheless significant incidental discoveries have been made in the course of its development.

Practical results include; new structures for “phasing networks”, i.e. complex filters, that are significantly less sensitive to their components than those in use (chapter 9); a new strategy for ladder simulation that results in canonic systems that retain good passband sensitivity (chapters 3 and 8); a powerful CAD tool (“dot”, chapter 6); a fundamentally new filter structure with better performance than a ladder simulation (the near-FSLB realization of chapter 8, which was based on ideas deveIoped in chapter ‘7); and an unusual-looking third-order ladder simulation using only one op amp (chapter 3).

Theoretical points were also made, especially in the work on FSLB filters in chapter 7 and in the investigation of “redundancy” in chapter 5.

## 11.3 Suggestions for Further Work

Because we have presented a design tool for filters, we clearly think that there is more work to be done in the areas of filter theory and generally of linear system synthesis. Much of the work that has been presented strongly suggests directions for future research in the area.

Extension of the synthesis technique to digital filters would be fairly easy, as suggested in chapter 3. A designer would then be able to use similar synthesis techniques for both digital and active filters.

The ease with which one may use $\{\mathbf{f}_i\}$ and $\{\mathbf{g}_i\}$ to help in linear functional tuning (chapter 4) suggests that there might be value in extending the study of filter performance in terms of the synthesis vectors to the large-change case. This type of extension might be useful both in correcting errors too large for a linear approximation to be valid and in analog fault diagnosis.

The material of chapter 5 suggests a new approach to the "tolerancing" problem which might be fruitful. A designer should be able to check, for a given design problem, whether the implicit redundancy of representation of important states offered by a highly coupled structure is worthwhile or whether the same problem can be solved more cheaply by tolerance assignment.

The beginnings of a geometric filter theory are suggested by our results on orthonormal systems, and some simple measurements of angles have been found useful in diagnosing and correcting problems in designs (chapter 8). It would be interesting to see whether this could be extended in the manner of [1] without losing touch with the practical problems of circuits. The link to information and communications theory suggested by the study of redundancy could also be productive.

Signal-space geometry could be related to the geometry of eigenvectors suggested in [2].

Chapter 6 produced fairly well-conditioned algorithms and its "choice of basis" seems reasonable, but there is probably still room for numerical work. It might be more useful, though, to work on the "man-machine interface" and try to turn "dot" into a generally useable program. FORTRAN is a wholly inadequate language for this task: if ADA compilers are ever made to work that language's richness in handling programmer-defined datatypes would be useful.

Chapter 6 used fairly primitive linear algebra to find answers to the specific problem of designing algorithms, yet nonetheless produced the result that $\sum_i f_i g_i$ is invariant (which in turn led to the work of chapter 7). The application of more powerful mathematics in this area (especially to the relation between $\{f_i\}$ and $\{g_i\}$) could produce even more.

It would be useful to add an iterative optimizer to "dot" [3].

Chapter 7 presented, among other things, some new fiIter structures which produce FSLB-sensitivity filters for some special transfer functions. These structures were capable of producing only a restricted set of functions because the "blocks" of which they were composed had to be identical. It would therefore be interesting to find out whether more generality can be obtained just by weakening the constraint to one that blocks should be "similar". A precise definition for "similarity" and a method for decomposing a transfer function into "similar" blocks inserted into FSLB topologies could yield a new and powerful type of filter synthesis. This would be especially useful if it did well in stopbands, where (as was shown in chapter 8) existing methods are poor. It would also be useful in designing amplitude-shaping networks, which are not handled well by ladders because their "passbands" are not flat.

Chapter 8 outlined the design of a practical filter by the methods of this thesis. Experimental work on a slightly simpler **(6th** order) function yielded encouraging results, so a detailed experimental characterization of the designs is now in order. The general procedures suggested in that chapter could be applied to some different transfer functions to explore their generality.

Work is continuing in the area of "complex filter synthesis" (chapter 9). Some preliminary results on medium-order examples are very encouraging, suggesting that only a few components need to be tightly matched. Analytic

**signals, Analytic signals, and complex Signals in general, seem to** have been **under-used in analog signal processing** because of two things: the lack of a general and straightforward design technique (including approximation and synthesis); and the poor sensitivity performance of current realizations. Chapter 9 has addressed these problems, so that now applications work is called for to use complex analog signals. Some experimental work is under way.

Several technologies can support complex signals. Chapter 9 was written for the active-RC case, but switched-C filters have good ratio-matching, low cost at high production volume, and can use switches for modulation. Switched-C complex filters therefore seem to be interesting building blocks for economical analog signal processing [4].

The work on complex signals could usefully be related to the older work on “N-path’ filtering [5] and to “sequency-domain” filters [6].

There are infinitely may representations of complex numbers as sets of real numbers, and decomposition into real and imaginary parts is not necessarily the best one to choose in every application. The material of chapter 9 could be reviewed with an eye to choosing optimum representations of complex signals.

Chapter 10 suggests a new approximation strategy for filters that shows promise: the next thing to do is to write an approximator and see how much improvement is possible on current transfer functions.

There seems to be plenty to do.

## 11.4 References

[1] W.M. Wonham, Linear *Multivariable Control:* ***A Geometric Approach,*** Springer-Verlag 1974

[2] C.W. Barnes, "*Roundoff* Noise and *Overfow* ***in*** *Normal* ***Digital*** *Filters*" IEEE Trans. Circuits and Systems, CAS-26, No. 3, March 1979

[3] R.F. Mackay, "*Generation* ***of*** *Low-****Sensitivity*** *State-* ***Space Active*** *Filters*", Ph.D. Thesis, University of Toronto, Toronto, Canada 1980

[4] K. Martin, "*Switched- Capacitor Networks*", Ph.D. Thesis, University of Toronto, Toronto, Canada, 1980

[5] L.E. Franks, "*N- path Filters*" Chapter 11 of *Modern Filter* ***Theory and Design ed. G.C.*** *Temes* ***and*** *S.K. Mitra* Wiley, 1973

[6] E. Daoud, W.B. Mikhael and M.N.S. Swamy, "New ***Active- RC Networks*** for ***the*** *Generation* ***and Detection of Single- Sideband Signals",*** IEEE Trans. Circuits and Systems, CAS-27, no. 12, pp. 1140-1154, Dec. 1980

# Index