DTU Library

Derivation of an expression for the roundoff noise determinant det (KW)^{1/2} for digital filters

Jørsboe, Helge

Published in:
IEEE Transactions on Circuits and Systems

Link to article, DOI:
10.1109/TCS.1978.1084516

Publication date:
1978

Document Version
Publisher's PDF, also known as Version of record

Citation (APA):
Jørsboe, H. (1978). Derivation of an expression for the roundoff noise determinant det (KW)^{1/2} for digital filters. *IEEE Transactions on Circuits and Systems*, *25*(8), 642-643. https://doi.org/10.1109/TCS.1978.1084516

Then the edges e_1 and e_5 belong to the path (1,2,5). Its path descriptor is therefore

$$(11001, 100010).$$

In this graph there are 3 paths from vertex 1 to vertex 5, namely (1,2,5), (1,2,4,5), and (1,4,5). The path descriptor matrix element $D(1,5)$ therefore is a list of 3 path descriptors:

((11001,100010)
(11011,100101)
(10011,001001)).

III. Algorithm

The following algorithm will generate D.

1. Initialize D by setting $D(i,j)=(d((i,j)))$, a list of one element, for each edge (i,j) in E, and $D(i,j)=$empty-list otherwise.
2. For $j=1,2,\cdots N$ repeat step A.
   A. For $i=1,2,\cdots,N$ and $k=1,2,\cdots,N$ repeat step i.
      i. For each (v,e) in $D(i,j)$ and (w,f) in $D(j,k)$, if $v\bullet w\bullet I_j=0$ then append $(v+w,e+f)$ to list $D(i,k)$.

It is seen that the step 2A is performed exactly N times. For each iteration of step 2A, step 2Ai is performed N^2 times. Therefore step 2Ai is performed exactly N^3 times.

It must be shown that this algorithm generates precisely the set of descriptors for all possible simple paths in G. Let the completion of the nth iteration of step 2, for $j=n$ be called *stage n*. At stage 0, prior to the first execution of step 2, D contains the descriptors for all simple paths with no internal vertices. (The algorithm does not generate descriptors for paths of length 0. If these are required, they may be generated separately.) It will be shown that at stage n, D contains precisely the descriptors for all simple paths whose internal vertices are in the list $(1,2,\cdots,n)$.

Assume, then, that this condition holds at stage $n-1$. Let p be any path whose internal vertices are in the list $(1,2,\cdots,n)$. If p does not contain vertex n, then its descriptor is already in D. Otherwise, p contains exactly one occurrence of vertex n, say $p=(i,\cdots,n,\cdots,j)$. Then the descriptors for paths $p'=(i,\cdots,n)$ and $p''=(n,\cdots,j)$ already exist in $D(i,n)$ and $D(n,j)$, respectively. Therefore the descriptor for p will be generated at stage n.

This shows that the descriptors for all simple paths will be generated by the algorithm exactly once each. To show that no other descriptor can be generated, notice that any pair of paths p and q with $d(p)=(v,e)$ in $D(i,j)$ and $d(q)=(w,f)$ in $D(j,k)$ such that $v\bullet w\bullet I_j=0$ has one and only one vertex in common, namely j. Therefore the concatenation pq is a simple path.

IV. Computer Implementation

An efficient computer implementation of the above mathematical algorithm has been achieved by dividing each list $D(i,j)$ at stage n into $n+1$ blocks $B_0, B_1, \cdots, B_n$ of descriptors for paths with $0,1,\cdots,n$ internal vertices, respectively. Then step 2Ai is performed only for blocks B_s in $D(i,j)$ and B_t in $D(j,k)$ such that $s+t\leqslant n-(i\leqslant n)-(j\leqslant n)-(k\leqslant n)$, where the three parenthesized expressions take on truth values 0 or 1. For all other blocks, path overlap is assured. By ordering the operations on the blocks in ascending order of $s+t$, no sorting is required, and the merging operation needed to order the list of blocks is reduced to simple concatenation.

References

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *The Design and Analysis of Computer Algorithms*. Reading, MA: Addison-Wesley, 1974, pp. 195–201.
[2] G. H. Danielson, "On finding simple paths and circuits in a graph," *IEEE Trans. Circuit Theory*, vol. CT-15, pp. 294–295, 1968.
[3] R. W. Floyd, "Algorithm 97, Shortest path," *Commun. Ass. Comput. Mach.*, vol. 5, p. 345, 1962.
[4] J. Ponstein, "Self-avoiding paths and the adjacency matrix of a graph," *J. SIAM*, vol. 14, pp. 600–609, 1966.
[5] S. Warshall, "A theorem on Boolean matrices," *J. Ass. Comput. Mach.*, vol. 9, pp. 11–12, 1962.

Derivation of an Expression for the Roundoff Noise Determinant det $(KW)^{1/2}$ for Digital Filters

HELGE JØRSBOE

***Abstract*—The minimal roundoff noise in fixed point digital filters is determined by a certain determinant, generally denoted by det $(KW)^{1/2}$. This determinant may be expressed by the poles and zeros of the filter transfer function $H(z)$. This paper presents a simple and direct derivation of this expression, based upon an expansion of $H(z)$ in partial fractions of the first order.**

The minimal roundoff noise in fixed point digital filters has been shown [1] to depend upon a certain determinant det $(KW)^{1/2}$ where the matrices K and W are related to a state variable description of the filter in the following way: element k_{ij} of K is defined as the scalar product of the response sequences of state variables i and j when a unit impulse is fed to the filter input; element w_{ij} of W is defined as the scalar product of the following two sequences at the filter output: one is the output sequence when a unit impulse is fed to state variable i, and the other is the output sequence when a unit impulse is fed to state variable j.

Now, det $(KW)^{1/2}$ is determined by the transfer function of the filter $H(z)$, by (29) in [1], reproduced here as (3). A derivation is given in [2]. It is the purpose of this paper to present a very simple and direct derivation of this relation.

Then, let the filter transfer function considered be

$$H(z)=\frac{q(z)}{p(z)}=\frac{q(z)}{(z-\lambda_1)(z-\lambda_2)\cdots(z-\lambda_n)}$$

$$=\frac{\dfrac{q(\lambda_1)}{(\lambda_1-\lambda_2)(\lambda_1-\lambda_3)\cdots(\lambda_1-\lambda_n)}}{z-\lambda_1}+\frac{\dfrac{q(\lambda_2)}{(\lambda_2-\lambda_1)(\lambda_2-\lambda_3)\cdots(\lambda_2-\lambda_n)}}{z-\lambda_2}+\cdots+\frac{\dfrac{q(\lambda_n)}{(\lambda_n-\lambda_1)(\lambda_n-\lambda_2)\cdots(\lambda_n-\lambda_{n-1})}}{z-\lambda_n}$$

$$\triangleq\frac{c_1q(\lambda_1)z^{-1}}{1-\lambda_1z^{-1}}+\frac{c_2q(\lambda_2)z^{-1}}{1-\lambda_2z^{-1}}+\cdots+\frac{c_nq(\lambda_n)z^{-1}}{1-\lambda_nz^{-1}}$$

Manuscript received September 12, 1977; revised February 14, 1978.
The author is with the Institute of Circuit Theory and Telecommunication, Technical University of Denmark, Lyngby, Denmark.

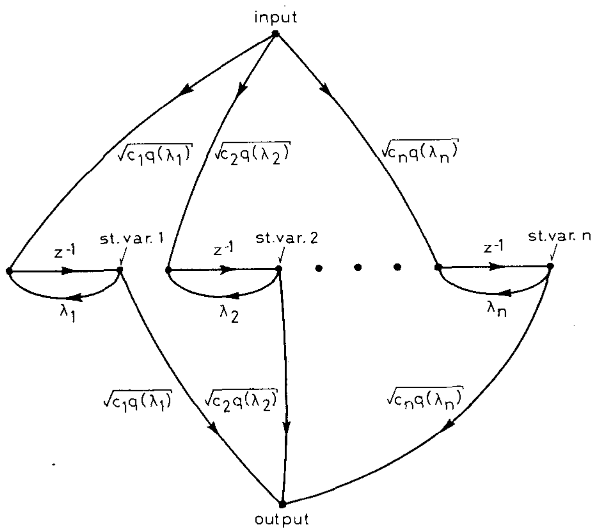

Fig. 1.

where we have also expanded the rational function $H(z)$ in a sum of partial fractions. (Note in passing that the numerator polynomial is that corresponding to the form shown of the denominator polynomial and not the more usual $z^{-n}q(z)$.) We will assume that the roots λ_i are distinct. Also, since $H(\infty)=h(0)$ is not interesting, we put it equal to zero. For convenience, we have introduced

$$c_i \triangleq [(\lambda_i-\lambda_1)\cdots(\lambda_i-\lambda_{i-1})(\lambda_i-\lambda_{i+1})\cdots(\lambda_i-\lambda_n)]^{-1}.$$

A signal flowgraph corresponding to this expansion may consist of n first-order filters in parallel, as illustrated in Fig. 1. The roots may be complex, in conjugate complex pairs (real coefficients in $H(z)$), and the filters may be rather impractical; however, at the output, of course, only real sequences appear.

Each numerator constant has been split in two equal factors, for convenience; in this way we get $k_{ij}=w_{ij}$, so that matrices K and W become identical. We then only need calculate det K, because

$$\det(KW)^{1/2}=\det(KK)^{1/2}=\det K.$$

So, at zero time we feed a unit impulse to the input and watch the response sequences of the various state variables (indicated on Fig. 1). These responses, however, are very simple: state variable i responds with the sequence

$$\sqrt{c_i q(\lambda_i)}\ \{0,1,\lambda_i,\lambda_i^2,\lambda_i^3,\cdots\}.$$

(The appearance of an initial zero in all these sequences has no influence on the k_{ij}.) Then k_{ij}, the scalar product of response i and response j, becomes

$$k_{ij}=\sqrt{c_i q(\lambda_i)}\ \sqrt{c_j q(\lambda_j)}\ \frac{1}{1-\lambda_i\lambda_j}.$$

Now consider the terms in det K. Each term will contain a factor $\Pi_i\sqrt{c_i q(\lambda_i)}\ \Pi_j\sqrt{c_j q(\lambda_j)}$, where i runs through $1,2,\cdots,n$ and j runs through a permutation of these same numbers $1,2,\cdots,n$. Then all the terms will contain the factor

$$\prod_{i=1}^{n}\sqrt{c_i q(\lambda_i)}\ \prod_{j=1}^{n}\sqrt{c_j q(\lambda_j)} = \prod_{i=1}^{n} c_i q(\lambda_i) = \prod_{i\neq j=1}^{n}(\lambda_i-\lambda_j)^{-1}\prod_{i=1}^{n} q(\lambda_i) \quad (1)$$

and we are left with the determinant

$$\begin{vmatrix} \frac{1}{1-\lambda_1^2} & \frac{1}{1-\lambda_1\lambda_2} & \cdots & \frac{1}{1-\lambda_1\lambda_n} \\ \frac{1}{1-\lambda_1\lambda_2} & \frac{1}{1-\lambda_2^2} & \cdots & \frac{1}{1-\lambda_2\lambda_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{1}{1-\lambda_1\lambda_n} & \frac{1}{1-\lambda_2\lambda_n} & \cdots & \frac{1}{1-\lambda_n^2} \end{vmatrix}. \quad (2)$$

This determinant, however, is familiar to mathematicians under the name of Cauchy's determinant [3], and the following relation is well known:

$$\begin{vmatrix} \frac{1}{x_1+y_1} & \frac{1}{x_1+y_2} & \cdots & \frac{1}{x_1+y_n} \\ \frac{1}{x_2+y_1} & \frac{1}{x_2+y_2} & \cdots & \frac{1}{x_2+y_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{1}{x_n+y_1} & \frac{1}{x_n+y_2} & \cdots & \frac{1}{x_n+y_n} \end{vmatrix} = \frac{\prod_{1\leq i<j\leq n}(x_j-x_i)(y_j-y_i)}{\prod_{1\leq i,j\leq n}(x_i+y_j)}.$$

If we put $x_i=\lambda_i^{-1}$ and $y_i=-\lambda_i$ and manipulate somewhat, we find the value

$$\frac{\prod_{i\neq j=1}^{n}(\lambda_i-\lambda_j)}{\prod_{i,j=1}^{n}(1-\lambda_i\lambda_j)}$$

for the determinant (2). As we should multiply by the factor (1), we get the result

$$\det(KW)^{1/2}=\frac{\prod_{i=1}^{n} q(\lambda_i)}{\prod_{i,j=1}^{n}(1-\lambda_i\lambda_j)}. \quad (3)$$

To conclude, the significance of this relation just proved is that it allows the calculation of det $(KW)^{1/2}$, and thereby of the minimal noise obtainable, in a straightforward way directly from $H(z)$ (with poles $\lambda_1,\cdots,\lambda_n$ and numerator polynomial $q(z)$).

ACKNOWLEDGMENT

The author would like to thank Dr. Jørn Justesen for several useful discussions.

REFERENCES

[1] C. T. Mullis and R. A. Roberts, "Synthesis of minimum roundoff noise fixed point digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-23, pp. 551–562, 1976.
[2] ——, private communication.
[3] D. E. Knuth, *The Art of Computer Programming: Vol. 1, Fundamental Algorithms*. Reading, MA: Addison-Wesley, 1973, p. 36.