A Note on Variable Recursive Digital Filters

KENNETH STEIGLITZ

Schüssler and Winkelnemper [1] note that when $z$ is replaced by the low-pass-to-low-pass bilinear frequency transformation in the transfer function of a recursive digital filter, the resulting direct form structure has delay-free loops and is therefore not realizable without modification. Johnson [2], [3] gives two methods for computing the new coefficients in a realization which is in direct form, except for a factor of the form $(1 + dz^{-1})^k$, where $k$ is the difference between the degrees of the original denominator and numerator. In [2] he shows that the new coefficients of both the denominator and numerator can be produced at the taps of a network similar to the frequency-warping network in [4], [5]. In [3] an FIR network is used to recompute the coefficients. (Mullis and Roberts [6] discuss the recomputation of coefficients in a state variable realization.)

In this correspondence we describe another method for realizing the transformed transfer function, one which preserves the all-pass substructure inherent in the bilinear transformation, and which results in a very fast coefficient recomputation.

We consider the transformation

$$\hat{z}^{-1} \rightarrow \frac{d + \hat{z}^{-1}}{1 + d\hat{z}^{-1}} = A(z)$$

and define

Manuscript received September 8, 1978; revised September 12, 1979. This work was supported by National Science Foundation Grant GK-42048 and the U.S. Army Research Office, Durham, under Grant DAHCO4-75-G0192.

The author is with the Department of Electrical Engineering and Computer Science, Princeton University, Princeton, NJ 08544.
\[ B(z) = A(z) - d \frac{(1 - d^2) z^{-1}}{1 + d z^{-1}}. \]

Given the transfer function
\[
\frac{N(z)}{D(z)} = \frac{\sum_{j=0}^{M} a_j z^j}{1 + \sum_{i=1}^{L} b_i z^i},
\]
the transformed transfer function is
\[
\frac{\sum_{j=0}^{M} a_j A(z)^j}{1 + \sum_{i=1}^{L} b_i A(z)^j}.
\]
The numerator can be implemented as it stands, as described in [1]. We therefore concentrate on the denominator: the difficulty is caused by the constant term in the sum. The following algebraic manipulation decreases by 1 the degree of the offending sum:
\[
1 + \frac{c_1 d \sum_{i=2}^{L} b_i A^{i-1} + c_1 B \sum_{i=1}^{L} b_i A^{i-1}}{1 + \sum_{i=1}^{L} b_i A^{i}} = \frac{c_1}{1 + \sum_{i=1}^{L} b_i A^{i-1}}.
\]
where
\[ c_1 = 1/(1 + dB). \]
If this is repeated \( L \) times, we get
\[
1 + \frac{c_1 d \sum_{i=2}^{L} b_i A^{i-1} + c_1 B \sum_{i=1}^{L} b_i A^{i-1}}{1 + \sum_{i=1}^{L} b_i A^{i}} = \frac{c_L}{1 + \sum_{i=1}^{L} b_i A^{i-1}}.
\]
Thus, we obtain the identity
\[
1 + \sum_{i=1}^{L} b_i A^{i-1} = \frac{1 + B \sum_{i=1}^{L} b_i A^{i-1}}{c_L}.
\]
Substituting \( B = A - d \) and equating coefficients of like powers of \( A \), we get the recursion relations
\[
b_L' = b_L,
b_i' = b_i + db_{i+1} \quad i = L - 1, \ldots, 1
\]
and
\[ c_L = \frac{1}{1 + dB}. \]
The transfer function \( B \) has no feedthrough term, so this form of the denominator is directly realizable as shown in Fig. 1.

When the numerator and denominator of the original filter are of the same degree, Johnson's form has the advantage of having the same filtering complexity as the original filter, although both numerator and denominator coefficients must be recomputed, whereas the present method requires re-computation of only the denominator coefficients. In the case of an all-pole transfer function, such as might arise in linear predictive coding, both methods result in a structure requiring more arithmetic to implement than the original.

A real advantage of the present method is the fast coefficient re-computation: the new \( b \)'s can be found with only \( L \) multiplications, \( L \) additions, and one division, in contrast with the techniques described by Johnson, which appear to require \( O(L^2) \) steps. The method also works without change for any causal \( A \); \( B \) is defined by subtracting the constant term from \( A \).

REFERENCES


