NONLINEAR FIR FILTER CLONES

R.K. PEARSON, M. GABBOUJ, AND J. ASTOLA
Tampere International Center for Signal Processing
Tampere University of Technology
P.O. Box 553, FIN-33101 Tampere, Finland
Phone: +358 3 365 38 47
E-mail: pearson@cs.tut.fi

Abstract

This paper considers the problem of designing nonlinear FIR filters from a *bottom-up* approach, in which simple components with known characteristics are combined into more complex filter structures. The specific approach considered here is based on the concept of a *clone* from universal algebra, generalizing the FIR-median hybrid (FMH) filter construction. Results presented here illustrate the flexibility of this procedure, the vast array of interesting special cases that emerge, including some new ones, and some of the types of qualitative behavior that are preserved under the clone interconnection strategy.

1 Introduction

An important idea in many engineering disciplines is that of modular design, in which more complex systems are built by interconnecting several simpler components. The general motivation for this approach lies in the fact that the simpler components on which these modular designs are based are often much easier to design and characterize than the overall interconnected system is. Hence, the design benefits can be substantial if we can develop systematic procedures for combining simple, well-understood components into larger systems in ways that preserve some of the desirable characteristics of these simple components while giving us a composite system with greater flexibility than the individual components have by themselves. A specific signal processing example is the development of separable median filters for two-dimensional signals (Astola and Kuosmanen, 1997). There, the overall filter output is the median of the outputs from individual median filters based on *subwindows* of a large two-dimensional window. These subwindows can be based on horizontal, vertical, diagonal, or cross-shaped subsets and they can be chosen to deal with different classes of image artifacts (e.g., horizontal or vertical streaks, etc.). Further, this approach is closely related to the "divide and conquer" strategy common in algorithm design (Sedgewick, 1998, p. 210), which often yields substantial computational savings.

This paper considers a specific implementation of the modular design strategy just described, specialized to the problem of nonlinear digital filter design. Because they are the most popular nonlinear digital filter class in practical applications, the focus is further restricted to the nonlinear FIR filter class defined in Sec. 2. The specific interconnection procedure considered here is based on the concept of a *clone* from universal algebra, introduced in Sec. 3.

2 NFIR filter design

This paper is concerned with the design of *nonlinear FIR (NFIR) filters,* defined by:

$$y_k = \Phi(x_{k-m}, \ldots, x_{k-n}), \tag{1}$$

where $n \geq m$ are integers and $\Phi : R^{n-m+1} \rightarrow R$ is any arbitrary function. This filter class contains the vast majority of nonlinear digital filters currently in use, including almost all of the filters described in the books by Astola and Kuosmanen (1997) and Pitas and Venetsanopoulos (1990). One reason for the popularity of this filter class is that, so long as the function $\Phi(\cdot)$ is continuous, the resulting filter is bounded-input, bounded-output (BIBO) stable (Pearson, 1999, p. 149). In contrast, stability conditions for recursive structures, in which y_{k-j} appears as an argument of the function $\Phi(\cdot)$ for some $j > 0$, are much more complicated, and instabilities can be extremely subtle (Pearson, 1999, p. 186).

Even within the class of nonlinear FIR structures, the flexibility is enormous, making design decisions difficult: how do we choose $\Phi(\cdot)$? As a specific example, the class of *C-filters* or *combination filters* is defined by (Astola and Kuosmanen, 1997, p. 70):

$$y_k = \sum_{i=m}^{n} c(r_j, j) x_{k-j}, \tag{2}$$

where r_j denotes the rank of x_{k-j} in the data window and $c(r_j, j)$ is a filter weighting coefficient that depends on both the temporal order j and the rank-order r_j of the input sample x_{k-j}. Despite its extremely special form, this class of filters includes many important special cases (e.g., both linear FIR filters and order-statistic filters like the median filter). Conversely, a C-filter based on a moving data window of length $N = n - m + 1$ requires the specification of an $N \times N$ matrix of coefficients and poor choices can lead to undesirable filter behavior.

As a partial solution to the nonlinear FIR filter design problem, this paper considers a special case of the *bottom-up design strategy:* given a collection of well-characterized, simple nonlinear FIR components and a constructive interconnection strategy, design a more complex composite NFIR filter that exhibits some of the desirable characterstics of the components but with greater overall flexibility than the components exhibit individually. The specific interconnection strategy considered here is based on the algebraic concept of a *clone*, defined in Sec. 3. Motivation for this particular choice lies in the fact that it is the interconnection strategy on which the FIR-median hybrid (FMH) filters are based (Heinonen and Neuvo, 1987). There, the clone-based interconnection of linear FIR filters with the nonlinear median filter leads to a class of nonlinear FIR filters with behavior that is intermediate in important practical aspects between that of the linear and nonlinear components on which it is built. For example, FMH filters exhibit impulsive noise rejection characteristics comparable to that of the median filter, while introducing less jitter in preserved edges and better smoothing characteristics for low-level noise.

3 Clones

The concept of a *clone* comes from universal algebra (Szendrei, 1986, p. 11) and is defined as follows. Suppose A is an arbitrary set, $r \geq 1$ and $q \geq 1$ are integers, $\phi_i : A^q \rightarrow A$ are arbitrary mappings for $i = 1, 2, \ldots, r$ and $G : A^r \rightarrow A$ is another arbitrary mapping. Here, $A^q = A \times A \times \cdots \times A$ is the cartesian product of the set A taken with itself q times and A will generally be taken either as the real line R or some subset of R. Also, the dimension of the domains of these functions (i.e., number of arguments q or r) is defined as the *arity* of the function. A *clone of A* is a set of such mappings that is closed under arbitarary *clone superpositions* of the form:

$$\Phi(\mathbf{x}) = G(\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), \ldots, \phi_r(\mathbf{x})), \tag{3}$$

for all $\mathbf{x} \in A^q$. In addition, a clone must contain all *projections* of the form:

$$P_i(\mathbf{x}) = x_i, \tag{4}$$

where x_i is the i^{th} component of the vector $\mathbf{x}$.

As a concrete example of a clone, consider the class of NFIR filters defined by Eq. (1). First, note that the projection operators may be viewed as identity systems (mapping x_k into x_k), delay operators (mapping x_k into x_{k-j} for $j > 0$) or advance operators (mapping x_k into x_{k+j} for $j > 0$). Next, consider arbitrary NFIR filters based on the functions $\Phi_0 : R^r \rightarrow R$ and $\Phi_i : R^m \rightarrow R$. The clone superposition of these filters is given by

$$\begin{aligned} y_k &= \Phi_0(\Phi_1(x_{k-m_1}, \ldots, x_{k-n_1}), \\ &\qquad \Phi_2(x_{k-m_2}, \ldots, x_{k-n_2}), \\ &\qquad \ldots, \Phi_r(x_{k-m_r}, \ldots, x_{k-n_r})) \\ &\equiv \Psi(x_{k-m^*}, \ldots, x_{k-n^*}), \end{aligned} \tag{5}$$

where $m^* = \min\{m_1, \ldots, m_r\}$ and $n^* = \max\{n_1, \ldots, n_r\}$. Since this result is an NFIR filter, it follows that this filter class is closed under clone superposition and includes the projections, so it defines the *clone of NFIR filters*, denoted $\mathcal{N}$.

Note that the NFIR filter clone $\mathcal{N}$ is defined *structurally:* all filters in this clone may be written in the form of Eq. (1) for the appropriate choice of the function $\Phi(\cdot)$. It is also possible to define clones *behaviorally,* specifying filters by their input/output behavior and showing that this behavior is preserved under clone superposition. An important example is the clone of *positive-homogeneous* or *scale-invariant* filters, defined as follows. A filter $\mathcal{F}$ is positive-homogeneous if it satisfies the following condition (Pearson, 2002):

$$\mathcal{F}\{\lambda x_k\} = \lambda \mathcal{F}\{x_k\}, \tag{6}$$

for all $\lambda > 0$. It is easy to see that if the component filters $\Phi_i(\cdot)$ in Eq. (5) all satisfy Eq. (6), then so does the clone superposition defined by $\Psi(\cdot)$. A second behaviorally-defined filter clone is the clone of *location-invariant* nonlinear filters (Pearson, 2001), defined as those filters $\mathcal{F}$ satisfying the condition:

$$\mathcal{F}\{x_k + c\} = \mathcal{F}\{x_k\} + c. \tag{7}$$

Again, it is not difficult to see that if the component filters $\Phi_i(\cdot)$ in Eq. (5) all satisfy Eq. (7), then so does the clone superposition defined by $\Psi(\cdot)$. Further, note that filters satisfying both scale-invariance and location-invariance have the useful property that their responses are invariant to changes in measurement units; for example, nonlinear filters applied to temperature sequences expressed in degrees Farenheit or Celsius yield compatable results. Finally, note that a practical disadvantage of behaviorally-defined clones is that behavioral definitions are not constructive: conditions (6) and (7) tell us how filters in the class behave, but not how to build them. One advantage of the clone construction is that, if we can identify simple members of these filter classes, it provides us a way of constructing new members of these behavioral classes.

4 Properties of clones

The clone superposition defined in Eq. (3) includes a number of important structures as special cases; the following discussions describe two.

4.1 Cascade interconnection

Any clone containing NFIR filters $\mathcal{F}$ and $\mathcal{G}$ also contains their cascade interconnection $\mathcal{F} \circ \mathcal{G}$. To see this point, consider the following NFIR filter cascade:

$$\begin{aligned} z_k = \mathcal{F}\{y_k\} &= \Psi(y_{k-m}, \ldots, y_{k-n}) \\ y_k = \mathcal{G}\{x_k\} &= \Phi(x_{k-p}, \ldots, x_{k-q}). \end{aligned} \tag{8}$$

Combining these equations gives a result that may be written in the following form:

$$\begin{aligned} \mathcal{F} \circ \mathcal{G}\{x_k\} &= \Psi(\Phi(x_{k-m-p}, \ldots, x_{k-m-q}), \\ &\qquad \ldots, \Phi(x_{k-n-p}, \ldots, x_{k-n-q})) \\ &= \Psi(\phi_m(x_{k-p}, \ldots, x_{k-q}), \\ &\qquad \ldots, \phi_n(x_{k-p}, \ldots, x_{k-q})), \end{aligned} \tag{9}$$

where $\phi_i(x_{k-p}, \ldots, x_{k-q}) = \Phi(x_{k-i-p}, \ldots, x_{k-i-q})$ represents the clone superposition of $\Phi(\cdot)$ with $q - p + 1$ copies of the projection P_i.

4.2 Weighted structures

Define a function $\Phi(\cdot)$ to be *of arbitrary arity* if $\Phi : A^q \to A$ is well-defined for all q. Typical examples include sums and unweighted averages, products and geometric means, ℓ_p norms, and arbitrary order statistics, including minima, maxima, and medians. If $\Phi(\cdot)$ is a function of arbitrary arity, the following *weighted filters* are well-defined:

$$\Phi_{\mathbf{w}}(\mathbf{x}_k) = \Phi(w_m \diamond x_{k-m}, \ldots, w_n \diamond x_{k-n}), \tag{10}$$

where $w_j \diamond x_j$ denotes w_j-fold replication of the variable x_j, where w_j is a positive integer. Since weighted filters may be represented as the clone superposition of the unweighted filter with a finite set of projections (w_j projections for each component x_j), it follows that any clone containing the unweighted filters defined by $\Phi(\cdot)$ also necessarily contains all weighted versions of this filter.

5 The L-filter clone

Although it does not constitute a clone itself, the class of L-filters *generates* an extremely interesting clone, defined as the set of filters obtained from clone superpositions of L-filters and projections. Here, we define the L-filter class rather strictly as filters of the form:

$$y_k = \sum_{j=-K}^{K} \alpha_j x_{(j)}, \quad 0 \leq \alpha_j \leq 1, \quad \sum_{j=-K}^{K} \alpha_j = 1, \tag{11}$$

where $x_{(j)}$ denotes the j^{th} order statistic in the rank-ordered symmetric data window. Under these restrictions, the only linear L-filter is the unweighted average, obtained by taking $\alpha_j = 1/(2K+1)$ for all j. Taking $\alpha_j = 0$ for all $j \neq 0$ and $\alpha_0 = 1$ leads to the standard median filter, and taking $\alpha_j = 1$ for some other fixed j and zero otherwise leads to other order-statistic filters. Other members of the class of L-filters include trimmed mean filters (Bednar and Watt, 1984) and the moving-window extension of *Gastwirth's estimator* (Andrews et al., 1972, p. 8), for which the weights are $\alpha_j = 0.3$ when j is the nearest integer to $\pm K/3$ and 0.4 for $j = 0$ (this estimator was one of the few non-adaptive estimators that appeared frequently as a "reasonable choice" in the summary of the Princeton Robustness Study (Andrews et al., 1972)).

It follows from the results presented in Sec. 4 that the L-filter clone includes all weighted median filters with positive integer weights (Yin et al., 1996), all linear FIR filters with rational weights summing to 1, all FMH filters and weighted FMH filters (Neejarvi et al., 1993) based on these linear FIR filters, cascade structures like the simple data sieve of Bangham (1993), LUM smoothers (Hardie and Boncelet, 1993), all multi-stage median filters (Astola and Kuosmanen, 1997, p. 88), the Wilcoxson filter based on the Hodges-Lehmann estimator (Crinon, 1985), and the simpler Hodges-Lehmann D-filter based on the Bickel-Hodges estimator (Kundu and Wu, 1989). Further, it is easy to show that all members of the clone generated by the L-filters are positive-homogeneous (Pearson, 2002), location-invariant (Pearson, 2001), and range-preserving: if $a \leq x_k \leq b$ for all k, then $a \leq \mathcal{F}\{x_k\} \leq b$ for all k.

6 A novel example

Mallows (1980) describes a useful nonlinear filter proposed by J.W. Tukey, called the "53 filter." This filter corresponds to a clone superposition of median filters, with the second-stage filter $\Phi_0(\cdot)$ taken as a 3-point median filter and each of the three first-stage filters taken as 5-point median filters:

$$\begin{aligned} y_k &= \text{median}\,\{\text{median}\{x_{k-3}, \ldots, x_{k+1}\}, \\ &\quad\quad \text{median}\{x_{k-2}, \ldots, x_{k+2}\}, \\ &\quad\quad\quad \text{median}\{x_{k-1}, \ldots, x_{k+3}\}\}. \end{aligned} \tag{12}$$

Since the median filter components appearing in this clone superposition belong to the L-filter class, it follows that Tukey's 53 filter belongs to the L-filter clone and exhibits all of the desirable behavior characteristic of this filter clone. Further, the recognition that this filter belongs to the L-filter clone immediately suggests some simple extensions that lead to new members of the L-filter clone.

For example, note that the 53 filter defined in Eq. (12) imposes different weights on the 7 data values, x_{k-3} through x_{k+3} appearing in its data window. Specifically, the terms x_{k-3} and x_{k+3} each appear once, the terms x_{k-2} and x_{k+2} each appear twice, and the terms x_{k-1}, x_k, and x_{k+1} all appear three times. A logical extension of this filter would be to replace these standard median filters with weighted median filters, allowing us to change these inherent relative weightings. It is known that there only exist a finite number of distinct integer-weighted median filters, and that this number grows rapidly with the increasing filter window width (Yin et al., 1996). For example, there are no nontrivial integer-weighted extensions of the three-point median filter appearing in Tukey's 53 filter, but there are 76 possible five-element integer-weighted median filters, listed in Table 1, giving $76^3 = 438,976$ possible weighted versions of Tukey's 53 filter, including the original.

7 Summary

The bottom-up design approach described in this paper provides a systematic method for building more complex nonlinear digital filters from simpler ones that are easier to design and analyze. One possible implementation of this idea is the use of cascade in-

Weight Vectors, $\mathbf{w}$	Possible Permutations
$[1, 1, 1, 1, 1]$	1
$[3, 1, 1, 1, 1]$	${}_1C_5 = 5$
$[2, 2, 1, 1, 1]$	${}_2C_5 = 10$
$[3, 2, 2, 1, 1]$	${}_1C_5 \cdot_2 C_4 = 60$
Total:	76

Table 1: Possible 5-element weighted median filters

terconnection structures, which can be analyzed using algebraic category theory (Pearson, 1999, ch. 7). The specific implementation of this idea described here is based on the concept of a clone from universal algebra, leading to a more flexible, practical interconnection strategy that is easily shown to preserve behavioral characteristics like BIBO stability, scale-invariance, location-invariance, and range preservation. As a special case, applying this interconnection strategy to the class of L-filters leads to a wide variety of known nonlinear FIR filters, along with some interesting novel ones like the weighted versions of Tukey's 53 smoother described in Sec. 6. Future publications will describe specific applications of this design strategy.

References

Andrews, D., Bickel, P., Hampel, F., Huber, P., Rogers, W., and Tukey, J. (1972). *Robust Estimates of Location.* Princeton University Press.

Astola, J. and Kuosmanen, P. (1997). *Fundamentals of Nonlinear Digital Filtering.* CRC Press.

Bangham, J. (1993). Properties of a series of nested median filters, namely the data sieve. *IEEE Trans. Signal Proc.*, 41:31–42.

Bednar, J. and Watt, T. (1984). Alpha-trimmed means and their relationship to median filters. *IEEE Trans. Acoustics, Speech, Signal Proc.*, 32:145–153.

Crinon, R. (1985). The wilcoxon filter: a robust filtering scheme. In *Proc. IEEE Intl. Conf. Acoustics, Speech, Signal Proc.*, pages 668–671, Tampa, Florida.

Hardie, R. and Boncelet, C. (1993). Lum filters: a class of rank order based filters for smoothing and sharpening. *IEEE Trans. Signal Proc.*, 41:1061–1076.

Heinonen, P. and Neuvo, Y. (1987). Fir-median hybrid filters. *IEEE Trans. Acoustics Speech Signal Proc.*, 35:832–838.

Kundu, A. and Wu, W.-R. (1989). Double-window hodges-lehman (d) filter and hybrid d-median filter for robust image smoothing. *IEEE Trans. Acoustics, Speech, Signal Proc.*, 37:1293–1298.

Mallows, C. (1980). Some theory of nonlinear smoothers. *Ann. Statist.*, 8:695–715.

Neejarvi, J., Varri, A., Fotopoulos, S., and Y.Neuvo (1993). Weighted fmh filters. *Signal Processing*, 47:181–190.

Pearson, R. (1999). *Discrete-Time Dynamic Models.* Oxford.

Pearson, R. (2001). A recursive variant of mallows' filter class. *IEEE Trans. Signal Proc.*, 49:2745–2752.

Pearson, R. (2002). Scale-invariant nonlinear digital filters. *IEEE Trans. Signal Proc.*, 50:1986–1993.

Pitas, I. and Venetsanopoulos, A. (1990). *Nonlinear Digital Filters.* Kluwer.

Sedgewick, R. (1998). *Algorithms in C++, Parts 1–4.* Addison-Wesley, 3rd edition.

Szendrei, A. (1986). *Clones in Universal Algebra.* Les Presses de l'Universite de Montreal.

Yin, L., Yang, R., Gabbouj, M., and Neuvo, Y. (1996). Weighted median filters: A tutorial. *IEEE Trans. Circuits Syst.—II: Analog Digital Signal Proc.*, 43:157–192.