

IIR Approximation of FIR Filters Via Discrete-Time Hybrid-Domain Vector Fitting

Chi-Un Lei and Ngai Wong

Abstract—We present a discrete-time hybrid-domain vector fitting algorithm, called HD-VFz, for the IIR approximation of FIR filters with an arbitrary combination of time- and frequency-sampled responses. The core routine involves a two-step pole refinement process based on a linear least-squares solve and an eigenvalue problem. Through hybrid-domain data approximation and digital partial fraction basis with relative stability consideration, HD-VFz exhibits fast computation and remarkable fitting accuracy in both time and frequency domains.

Index Terms—Approximation methods, identification, IIR digital filters, rational functions, vector fitting.

I. INTRODUCTION

DIGITAL filters are widely used in digital signal processing and communication systems. Compared to a finite-impulse-response (FIR) filter, an infinite-impulse-response (IIR) filter generally features a smaller filter order though with more stringent design constraints such as numerical stability in finite precision arithmetics. There are various approaches to design IIR filters with arbitrary frequency responses such as eigenfilter, optimization, balanced truncation and approximation of FIR responses, e.g., [1]–[3]. However, most algorithms cannot achieve a hybrid-domain (*viz.* frequency- and time-domain) accuracy since a purely frequency-domain design approach may overlook certain time-domain requirements and vice versa. For example, to avoid inter-symbol interference (ISI) in digital data transmission, the designed Nyquist filter has strict time-domain constraints at the zero crossing points [1].

Recently, vector fitting (VF) [4], a continuous-time system identification technique, has been extended to its discrete-time counterpart, called discrete-time vector fitting (VFz) [3], for efficient IIR approximation of FIR filters. It is shown that VFz exhibits fast computation and a similar accuracy to the near-optimal (but much more expensive) balanced model reduction (BMR) [2] IIR filter design approach. The time-domain variant of VFz, named TD-VFz, has also been formulated and applied to linear macromodeling with excellent noise immunity [5]. Consequently, to simultaneously achieve time- and frequency-domain accuracy, we integrate VFz and TD-VFz into a discrete-time hybrid-domain counterpart, called HD-VFz, for designing

IIR filters with arbitrary time and frequency responses. A pole radius constraint is readily incorporated for numerical stability in finite-precision implementation. Design examples then confirm the remarkable efficacy of HD-VFz.

II. DISCRETE-TIME HYBRID-DOMAIN VECTOR FITTING

Our objective is to approximate an FIR digital filter response

$$f(z) = \sum_{n=0}^L h_n z^{-n}, h_n \in \mathbb{R}, h_L \neq 0 \quad (1)$$

with a causal and stable IIR filter

$$\hat{f}(z) = \frac{P(z)}{Q(z)} = \frac{\sum_{\mu=0}^N p_{\mu} z^{-\mu}}{\sum_{v=0}^N q_v z^{-v}}, p_{\mu}, q_v \in \mathbb{R}, q_0 = 1. \quad (2)$$

Therefore, all poles of $\hat{f}(z)$ (zeros of $Q(z)$) must lie in $|z| < 1$. Obviously, $\bar{f}(e^{j\Omega}) = f(e^{-j\Omega})$ and $\hat{f}(e^{j\Omega}) = \hat{f}(e^{-j\Omega})$, $\forall \Omega \in [-\pi, \pi)$, where $\bar{(\cdot)}$ denotes complex conjugate, so both $f(z)$ and $\hat{f}(z)$ correspond to real time-domain sequences. To exclude the trivial case $\hat{f}(z) \equiv f(z)$, we assume $N < L$.

VFz [3] utilizes a partial fraction basis to seek a rational approximation to the desired response $f(z)$, namely

$$\hat{f}(z) = \left(\sum_{n=1}^N \frac{c_n}{1 - z^{-1} a_n} \right) + d \approx f(z). \quad (3)$$

Subsequently, c_n 's and a_n 's are either real or complex conjugate pairs. Analogous to VF, suppose an initial set of poles $\{\alpha_n^{(0)}\}$, $|\alpha_n^{(0)}| < 1$, is specified for VFz, by introducing a weighting function $\sigma(z)$, we have

$$\underbrace{\left(\sum_{n=1}^N \frac{c_n}{1 - z^{-1} \alpha_n^{(i)}} \right)}_{(\sigma f)(z)} + d \approx \underbrace{\left(\left(\sum_{n=1}^N \frac{\gamma_n}{1 - z^{-1} \alpha_n^{(i)}} \right) + 1 \right)}_{\sigma(z)} f(z) \quad (4)$$

for $i = 0, 1, \dots, N_T$, where N_T denotes the number of iterations when convergence is attained or the upper bound is reached. Here (4) is linear with respect to its unknowns c_n , d , and γ_n , and constrains $(\sigma f)(z)$ and $\sigma(z)f(z)$ to share the same poles, which in turn implies that the poles of $f(z)$ in IIR form are approximated by the zeros of $\sigma(z)$. Solving for the zeros of $\sigma(z)$ therefore produces, in the least-squares (LS) sense, an approximation to the poles of $f(z)$, *viz.* $\{\alpha_n^{(i+1)}\}$, which are then fed back to (4) as the next set of known poles. We remark that here $f(z)$ (FIR in nature) actually has all its poles on the origin, but is treated by VFz (based on partial fraction synthesis) as if it is an IIR filter of unknown (possibly infinite) order, and an

Manuscript received November 04, 2008; revised February 17, 2009. Current version published April 24, 2009. This work was supported in part by the Hong Kong Research Grants Council under Project HKU 717407E, and by the University Research Committee of The University of Hong Kong. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Alfred Mertins.

The authors are with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Kowloon, Hong Kong (e-mail: culei@eee.hku.hk; nwong@eee.hku.hk).

Digital Object Identifier 10.1109/LSP.2009.2017478

attempt is made to approximate it with another IIR filter $\hat{f}(z)$ whose order is prescribed.

We subsequently transform (4) into its discrete-time time-domain counterpart. Applying an input $X(z)$ to $f(z)$ and letting $Y(z) = f(z)X(z)$ be the output, the time-domain relationship is then given by the inverse z -transform

$$\begin{aligned} y_n[k] &= \left((\alpha_n^{(i)})^k u[k] \right) * y[k] \\ x_n[k] &= \left((\alpha_n^{(i)})^k u[k] \right) * x[k] \\ y[k] &\approx dx[k] + \sum_{n=1}^N c_n x_n[k] - \sum_{n=1}^N \gamma_n y_n[k] \end{aligned} \quad (5)$$

where $*$ denotes convolution and $u[k]$ is the Heaviside unit step sequence. The combination of (4) and (5) then gives rise to the hybrid-domain fitting algorithm HD-VFz.

A. Pole Computation

Poles are first iteratively approximated. For frequency-domain approximation, starting from (4), for the N_s frequency points $z_k = e^{j\Omega_k}$, $\Omega_k \in [0, \pi)$ and $k = 0, 1, \dots, N_s - 1$, the response can be put into a system of linear equations as shown in (6), at the bottom of the page, where $p_n(k) = (1)/(1 - z_k^{-1}\alpha_n^{(i)})$ and $q_n(k) = (f(z_k))/(1 - z_k^{-1}\alpha_n^{(i)})$. For time-domain approximation, from (5) and suppose N_t samples of the input and output sequences, $x[k]$ and $y[k]$, are captured, another system of equations are set up for the same x in (6), as shown in (7), at the bottom of the page. Consequently, HD-VFz can fit both time- and frequency-sampled responses simultaneously in the LS sense via the overdetermined equation (provided $N_s + N_t > 2N + 1$)

$$Ax = b \text{ where } A = \begin{bmatrix} A_f \\ A_t \end{bmatrix} \text{ and } b = \begin{bmatrix} b_f \\ b_t \end{bmatrix}. \quad (8)$$

Using the last N elements of the LS solution of x , i.e., γ_1 to γ_N , $\sigma(z)$ of (4) can be constructed whose zeros, denoted by $\{\alpha_n^{(i+1)}\}$, then form the new set of poles in the next HD-VFz

iteration. Similar to the formulation in VFz [3], the zeros of $\sigma(z)$ are implicitly obtained as the eigenvalues of

$$\begin{aligned} \Psi &= \left(\begin{bmatrix} 1/\alpha_1^{(i)} & & \\ & \ddots & \\ & & 1/\alpha_N^{(i)} \end{bmatrix} + \begin{bmatrix} 1/\alpha_1^{(i)} \\ \vdots \\ 1/\alpha_N^{(i)} \end{bmatrix} [\gamma_1 \ \cdots \ \gamma_N] \right)^{-1} \\ &= \begin{bmatrix} \alpha_1^{(i)} & & \\ & \ddots & \\ & & \alpha_N^{(i)} \end{bmatrix} - \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} R^{-1} \begin{bmatrix} \alpha_1^{(i)} \gamma_1 \\ \vdots \\ \alpha_N^{(i)} \gamma_N \end{bmatrix}^T \end{aligned} \quad (9)$$

where $R = 1 + \sum_{i=1}^N \gamma_i$. When only real poles are present, Ψ is real. To ensure stability, it is required that every $|\alpha_n^{(i+1)}| < 1$. Otherwise, its reciprocal is taken, viz. $\alpha_n^{(i+1)} := 1/\alpha_n^{(i+1)}$, such that the pole is flipped back inside the unit circle. Here a real $\alpha_n^{(i+1)}$ is assumed but flipping of conjugate poles follows exactly by multiplying two conjugate reciprocals.

B. Modification for Complex Poles

The transfer function $\hat{f}(z)$ in (3) may contain complex conjugate poles and residues whose time-domain transforms also contain complex conjugate pairs, and thus conform to a real response. If these complex quantities are directly used in (6) and (7), finite-precision arithmetics would almost always result in inexact cancellation of imaginary parts and lead to erroneous responses. Subsequently, (6) and (7) should be rewritten to ensure all quantities in x are real. For instance, assume $\alpha_2^{(i)} = \overline{\alpha_1^{(i)}}$, $c_2 = \overline{c_1}$ and $\gamma_2 = \overline{\gamma_1}$, this is achieved by modifying the entries in A_f , A_t and x as shown in (10), at the bottom of the next page, where we get (11), also shown at the bottom of the next page,

$$\begin{aligned} p'_1(k) &= \left(\frac{1}{1 - z_k^{-1}\alpha_1^{(i)}} + \frac{1}{1 - z_k^{-1}\overline{\alpha_1^{(i)}}} \right) \\ p'_2(k) &= j \left(\frac{1}{1 - z_k^{-1}\alpha_1^{(i)}} - \frac{1}{1 - z_k^{-1}\overline{\alpha_1^{(i)}}} \right) \\ q'_1(k) &= \left(\frac{f(z_k)}{1 - z_k^{-1}\alpha_1^{(i)}} + \frac{f(z_k)}{1 - z_k^{-1}\overline{\alpha_1^{(i)}}} \right) \\ q'_2(k) &= j \left(\frac{f(z_k)}{1 - z_k^{-1}\alpha_1^{(i)}} - \frac{f(z_k)}{1 - z_k^{-1}\overline{\alpha_1^{(i)}}} \right) \end{aligned}$$

$$A_f x = b_f$$

$$A_f = \begin{bmatrix} p_1(0) & \cdots & p_N(0) & 1 & -q_1(0) & \cdots & -q_N(0) \\ p_1(1) & \cdots & p_N(1) & 1 & -q_1(1) & \cdots & -q_N(1) \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ p_1(N_s - 1) & \cdots & p_N(N_s - 1) & 1 & -q_1(N_s - 1) & \cdots & -q_N(N_s - 1) \end{bmatrix}$$

$$x = [c_1 \ \cdots \ c_N \ d \ \gamma_1 \ \cdots \ \gamma_N]^T,$$

$$b_f = [f(z_0) \ f(z_1) \ \cdots \ f(z_{N_s-1})]^T \quad (6)$$

$$A_t x = b_t$$

$$A_t = \begin{bmatrix} x_1[0] & \cdots & x_N[0] & x[0] & -y_1[0] & \cdots & -y_N[0] \\ x_1[1] & \cdots & x_N[1] & x[1] & -y_1[1] & \cdots & -y_N[1] \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ x_1[N_t - 1] & \cdots & x_N[N_t - 1] & x[N_t - 1] & -y_1[N_t - 1] & \cdots & -y_N[N_t - 1] \end{bmatrix}$$

$$b_t = [y[0] \ y[1] \ \cdots \ y[N_t - 1]]^T \quad (7)$$

where $\Re(\circ)$ and $\Im(\circ)$ denote the real and imaginary parts, respectively. Modification for other complex conjugate poles and residues follows analogously. To compute the zeros of $\sigma(z)$ which now contain complex poles, we apply similarity transform to (9) to transform it to a real matrix. Each pair of conjugate poles now manifest as a 2×2 diagonal block in Ψ . For example, when $\alpha_2^{(i)} = \overline{\alpha_1^{(i)}}$, Ψ takes the form

$$\Psi = \begin{bmatrix} \Re(\alpha_1^{(i)}) & \Im(\alpha_1^{(i)}) & & \\ -\Im(\alpha_1^{(i)}) & \Re(\alpha_1^{(i)}) & & \\ & & \ddots & \\ & & & \ddots \end{bmatrix} - \begin{bmatrix} 2 \\ 0 \\ \vdots \end{bmatrix} R^{-1} \begin{bmatrix} \Re(\alpha_1^{(i)}\gamma_1) \\ \Im(\alpha_1^{(i)}\gamma_1) \\ \vdots \end{bmatrix}^T \quad (12)$$

in which R is easily seen to be real.

C. Reconstructing the Rational Function

Suppose a converged set of poles $\{\alpha_n^{(N_T)}\}$ are obtained, the final step is to reconstruct the rational function. Referring to (4) and (5), we should now have $\sigma(z) \approx 1$ or equivalently $\sigma[k] = \delta[k]$. Therefore, we have $\hat{f}(z_k) = (\sum_{n=1}^N c_n(1 - z_k^{-1}\alpha_n^{(N_T)})^{-1}) + d \approx f(z_k)$ and $y[k] \approx dx[k] + \sum_{n=1}^N c_n x_n[k]$. The residues c_n of $\hat{f}(z)$ [see (3)] are computed in the same manner as in (6)–(8), (10), and (11), except that the last N columns in A and the last N entries in x are now omitted, making c_1 to c_N and d the only unknowns.

D. Remarks

- 1) As a generalization of VFz, HD-VFz converges to a near-global-optimum in the LS sense for noise-free responses and is robust to initial pole placement [3], [5]. Moreover, using the h_n 's in (1), a Hankel singular value plot is readily derived for guiding the selection of the IIR filter order in HD-VFz [5].
- 2) The quantization effects of IIR filters in hardware implementation lead to deviated pole locations [6]. The response of the realized filter may deteriorate or even become unstable. Similar to VFz, near-unity pole radius can be directly scaled away from the unit circle in HD-VFz to provide a stability margin for filter robustness.
- 3) The accuracy in the LS solution of (8) may be affected by the initial poles and function basis. Orthonormal basis or QR decomposition with rank-revealing column weighting [7] can be used to normalize the columns of A in (8), thus reducing its condition number (i.e., closer to unity) for a more accurate solution. Theoretically, HD-VFz generates the same IIR approximant subject to any non-zero input. In

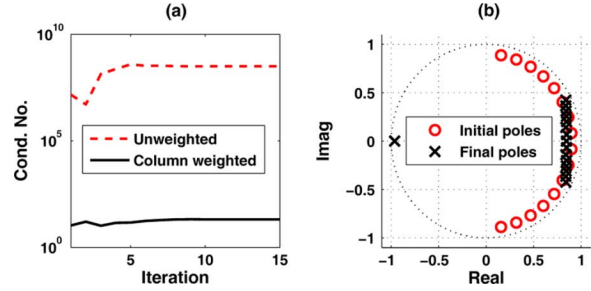


Fig. 1. (a) Condition numbers of A in (8) at each iteration in the Nyquist filter example with and without column weighting. (b) Converged poles of the filter designed by HD-VFz with column weighting.

practice, larger input and output amplitudes are preferred for a better conditioning of (8). For example, apart from an impulse input, a step input may sometimes result in a better IIR approximant.

III. NUMERICAL EXAMPLES

The proposed HD-VFz algorithm is compared against BMR [2] and the original VFz [3] (performance of VFz versus other IIR filter design algorithms can be found in [3]). All algorithms are coded in Matlab m-script files and run on a 3.4 GHz 1G-RAM PC. We begin with a 90th-order Nyquist filter designed with the Matlab filter design toolbox command `firnyquist`, using the specification $L = 90, K = 46, W = 10$ and a roll-off factor 0.65, such that its time response satisfies

$$h(K \pm kW) = \begin{cases} 1/W, & k = 0 \\ 0, & k \neq 0 \end{cases} \quad (13)$$

i.e., the impulse response crosses the time axis every W samples. The prototype response is fitted by HD-VFz, VFz and BMR into 16th-order IIR filters. We use 110 frequency samples and 91 impulse response samples (viz., h_0 to h_{90}) for HD-VFz, 210 linearly spaced frequency samples for VF, and again the 91 impulse samples in BMR. QR decomposition with column weighting [7] is employed in HD-VFz and VFz to improve the condition number of A in (8) for better approximation accuracy, as seen in Fig. 1(a). The poles in both algorithms converge in 15 iterations, as in Fig. 1(b). (In numerous experiments not reported here, HD-VFz and VF always converge within 20 iterations.) The CPU times of HD-VFz, VFz and BMR are 0.625 s, 0.484 s, and 1.026 s, respectively. The longer time of HD-VFz than VFz is mainly due to the convolution in (5). The frequency and time responses of the designed IIR filters are shown in Fig. 2 and Tables I and II. It is obvious from the plots that HD-VFz outperforms other algorithms and has the least group delay ripple

$$A_f = \begin{bmatrix} p'_1(0) & p'_2(0) & \cdots & 1 & -q'_1(0) & -q'_2(0) & \cdots \\ p'_1(1) & p'_2(1) & \cdots & 1 & -q'_1(1) & -q'_2(1) & \cdots \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \\ p'_1(N_s - 1) & p'_2(N_s - 1) & \cdots & 1 & -q'_1(N_s - 1) & -q'_2(N_s - 1) & \cdots \end{bmatrix} \quad (10)$$

$$x = [\Re(c_1) \quad \Im(c_1) \quad \cdots \quad d \quad \Re(\gamma_1) \quad \Im(\gamma_1) \quad \cdots]^T$$

$$A_t = \begin{bmatrix} 2\Re(x_1[0]) & -2\Im(x_1[0]) & \cdots & x[0] & -2\Re(y_1[0]) & 2\Im(y_1[0]) & \cdots \\ 2\Re(x_1[1]) & -2\Im(x_1[1]) & \cdots & x[1] & -2\Re(y_1[1]) & 2\Im(y_1[1]) & \cdots \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \\ 2\Re(x_1[N_t - 1]) & -2\Im(x_1[N_t - 1]) & \cdots & x[N_t - 1] & -2\Re(y_1[N_t - 1]) & 2\Im(y_1[N_t - 1]) & \cdots \end{bmatrix} \quad (11)$$

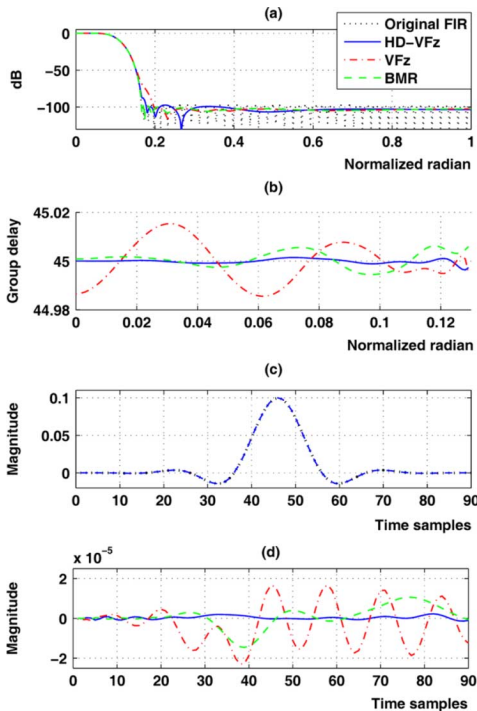


Fig. 2. Responses of various IIR approximants of the Nyquist filter: frequency-domain (a) magnitude responses and (b) passband group delays; time-domain (c) impulse responses and (d) errors versus the original response.

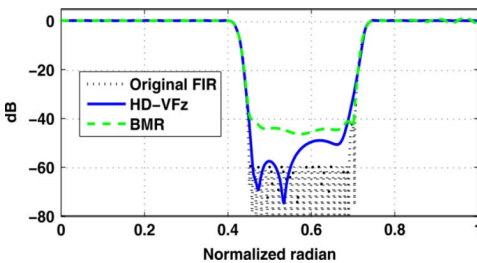


Fig. 3. Various IIR approximations of the bandstop filter.

TABLE I
APPROXIMATION ERRORS OF THE NYQUIST FILTER: PASSBAND $(0, 0.1\pi)$ (A) MAGNITUDE AND (B) GROUP DELAY; AND (C) STOPBAND $(0.2\pi, \pi)$ ATTENUATION. THE BEST TWO CANDIDATES ARE UNDERLINED

	BMR [2]	VFz [4]	HD-VFz (impulse)	HD-VFz (step)	HD-VFz (unweighted)
l_2 (A)	2.00e-3	1.60e-3	2.38e-4	1.03e-4	2.96e-4
l_2 (B)	0.0289	0.0949	0.0070	0.0219	0.0260
l_2 (C)	1.94e-4	1.86e-4	2.08e-4	2.09e-4	2.08e-4
	BMR [2]	VFz [4]	HD-VFz (impulse)	HD-VFz (step)	HD-VFz (unweighted)
l_∞ (A)	4.40e-4	2.94e-4	4.75e-5	2.39e-5	6.09e-5
l_∞ (B)	0.0056	0.0153	0.0014	0.0014	0.0041
l_∞ (C)	1.00e-5	9.06e-6	1.38e-5	1.38e-5	1.38e-5

TABLE II
IMPULSE RESPONSES OF THE NYQUIST FILTER AT THE ZERO CROSSING POINTS. THE BEST TWO CANDIDATES ARE UNDERLINED

Time samples	6th	16th	26th	36th
BMR [2]	3.1e-7	1.2e-7	2.9e-6	-1.1e-5
VFz [4]	8.5e-7	-2.5e-6	-1.4e-5	-1.5e-5
HD-VFz (impulse)	-2.6e-7	1.3e-7	2.5e-7	1.8e-6
HD-VFz (step)	-3.6e-7	1.5e-7	-3.4e-7	-1.4e-7
HD-VFz (unweighted)	-2.6e-7	1.5e-7	3.4e-7	1.6e-6
Time samples	56th	66th	76th	86th
BMR [2]	-4.4e-7	3.1e-6	1.1e-5	2.9e-6
VFz [4]	1.1e-5	-1.1e-5	-1.6e-5	4.6e-6
HD-VFz (impulse)	3.7e-9	-5.6e-7	4.5e-8	-1.6e-7
HD-VFz (step)	-2.0e-7	-5.5e-7	-1.3e-6	-1.1e-7
HD-VFz (unweighted)	-3.0e-8	-2.0e-7	1.2e-6	1.8e-7

TABLE III
APPROXIMATION ERRORS OF THE BANDSTOP FILTER: PASSBANDS $(0, 0.4\pi)$ AND $(0.75\pi, \pi)$ (A) MAGNITUDE AND (B) GROUP DELAY; AND (C) STOPBAND $(0.45\pi, 0.69\pi)$ ATTENUATION

	l_∞ (A)	l_∞ (B)	l_∞ (C)	CPU Time/sec
BMR [2]	0.1470	5.6146	0.0118	2.3625
HD-VFz	0.0143	0.5473	0.0102	1.2031

(closest to linear phase) in the passband and smallest error on the time axis. The tables, wherein the best two candidates are underlined, further confirm the generally better performance of the IIR filters designed by HD-VFz. In other words, the longer computation in HD-VFz than VFz is justified by the much more accurate design and better preservation of dual-domain characteristics, in contrast to VF and BMR that operate purely in the frequency domain and time domain, respectively. Next, an equiripple 130th-order linear-phase FIR bandstop filter, designed by the Parks-McClellan algorithm, is approximated by TD-VFz (131 time-domain step responses, 220 frequency samples and four iterations) and BMR (131 impulse responses), both resulting in 55th-order IIR approximants. The magnitude responses and approximation errors are shown in Fig. 3 and Table III, which again demonstrate the superiority of HD-VFz.

Finally, to study the performance of HD-VFz under the modifications in Section II.D remark #3, the first example is tested with: (i) a step input (to obtain the time response) while still using column weighting in solving (8), and (ii) impulse input but without column weighting. In (i), the CPU time is similar to the above, while that for (ii) is only 0.365 s by skipping the weighting operation. The results are also included in Tables I and II. It is seen that QR decomposition with column weighting, though requires longer computation, generally leads to better numerical conditioning and higher filter quality. Also, using step input/output may sometimes lead to more accurate IIR approximants but the use of impulse response would generally suffice. In all cases, the accuracy of the HD-VFz approach over that of VFz or BMR is obvious.

IV. CONCLUSION

VFz and TD-VFz have been generalized to a hybrid-domain counterpart, called HD-VFz, for IIR filter approximation with dual-domain accuracy. Compared to VFz or TD-VFz, HD-VFz exhibits comparable complexity while producing more accurate IIR approximants.

REFERENCES

- [1] S. C. Pei, C. C. Hsu, and P. H. Wang, "Design of a class of IIR eigenfilters with time- and frequency-domain constraints," *IEEE Trans. Circuits Syst. II*, vol. 49, no. 2, pp. 145–151, Feb. 2002.
- [2] B. Beliczynski, I. Kale, and G. D. Cain, "Approximation of FIR by IIR digital filters: An algorithm based on balanced model reduction," *IEEE Trans. Signal Process.*, vol. 40, no. 3, pp. 532–542, Mar. 1992.
- [3] N. Wong and C.-U. Lei, "IIR approximation of FIR filters via discrete-time vector fitting," *IEEE Trans. Signal Process.*, vol. 56, no. 3, pp. 1296–1302, Mar. 2008.
- [4] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Trans. Power Del.*, vol. 14, no. 3, pp. 1052–1061, Jul. 1999.
- [5] C. U. Lei and N. Wong, "Efficient linear macromodeling via discrete-time time-domain vector fitting," in *Proc. Int. Conf. VLSI Design*, Jan. 2008, pp. 469–474.
- [6] A. V. Oppenheim, R. W. Schaffer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1999.
- [7] B. Gustavsen, "Comments on 'A comparative study of vector fitting and orthonormal vector fitting techniques for EMC applications,'" in *Proc. IEEE Int. Symp. Electromagnetic Compatibility*, Sep. 2007, pp. 131–134.