Circuits, Systems, and Signal Processing (2022) 41:1223-1254
https://doi.org/10.1007/s00034-021-01837-z

Control-Bounded Analog-to-Digital Conversion

Hampus Malmberg[1] · **Georg Wilckens**[2] · **Hans-Andrea Loeliger**[1]

Received: 24 April 2021 / Revised: 25 August 2021 / Accepted: 26 August 2021 /
Published online: 13 September 2021

Abstract
A control-bounded analog-to-digital converter consists of a linear analog system that is subject to digital control, and a digital filter that estimates the analog input signal from the digital control signals. Such converters have many commonalities with delta–sigma converters, but they can use more general analog filters. The paper describes the operating principle, gives a transfer function analysis, and describes the digital filtering. In addition, the paper discusses two examples of such architectures. The first example is a cascade structure reminiscent of, but simpler than, a high-order MASH converter. The second example combines two attractive properties that have so far been considered incompatible. Its nominal conversion noise (assuming ideal components) essentially equals that of the first example. However, its analog filter is a fully connected network to which the input signal is fed in parallel, which potentially makes it more robust against nonidealities.

Keywords Analog-to-digital conversion · Continuous-time delta–sigma modulator · Chain of integrators · Kalman recursions · Wiener filter · Factor graphs

1 Introduction

Control-bounded analog-to-digital conversion was proposed in [13], as a simplification of control-aided analog-to-digital conversion proposed in [10]. Like several other

✉ Hampus Malmberg
malmberg@isi.ee.ethz.ch

Georg Wilckens
georg.wilckens@gmail.com

Hans-Andrea Loeliger
loeliger@isi.ee.ethz.ch

1 Department of Information Technology and Electrical Engineering, ETH Zurich, Zurich, Switzerland

2 Zurich, Switzerland

conversion principles (including pipelined conversion [5], beta-expansion conversion [2,6,7], and modulo conversion [15]), control-bounded conversion works by multiple stages of analog amplification with intermediate steps of adding (or subtracting) digitally controlled quantities. However, control-bounded conversion stands out by its analog part operating in continuous time (rather than with discrete-time samples), which is reminiscent of continuous-time delta–sigma ($\Delta\Sigma$) converters. Moreover, the reconstruction principle of control-bounded conversion differs from other conversion principles. In consequence, control-bounded converters can use more general analog filter structures (potentially consuming less power) than delta–sigma converters.

The description in [13] is terse and the performance analysis is rudimentary. In this paper, we describe the operating principle and the digital estimation in more detail and give a more detailed transfer function analysis. Moreover, we discuss two examples of such architectures. The first example (first presented in [13]) is a chain of integrators resembling a multi-stage noise shaping (MASH) $\Delta\Sigma$ ADC [8,16,17], but with a simpler analog part that precludes a conventional digital cancellation scheme.

Before we move on to the second example, we recall here that the challenge of real-world analog-to-digital conversion is not to minimize the nominal conversion noise with *ideal* analog circuits, but to cope efficiently (in particular, with limited power consumption) with nonideal circuits and disturbances including component mismatch, thermal noise, etc. But analog cascade structures (as in high-order MASH ADCs and in our first example) are particularly sensitive to disturbances and imperfections at the early stage(s). Therefore, these early stage(s) need to be implemented with much higher precision (and therefore with much higher power consumption) than the later stages,[1] which counteracts the idea of a uniform cascade.

With this background, we now turn to our second example, which is obtained from the first example by an orthogonal transformation of the analog state space. In consequence, the nominal conversion performance (with ideal analog circuits) remains essentially unchanged and is easily scaled to any desired level. However, the physical state space is no longer a cascade, but a fully connected (and nearly uniform) network, into which the analog input signal is fed fully in parallel. In consequence, this new architecture promises to be quite robust against component mismatch and other nonidealities.

The paper is structured as follows. The operating principle and the basic transfer function analysis of control-bounded ADCs are given in Sect. 2. A conversion noise analysis is given in Sect. 3. The first example architecture is presented and analyzed in Sect. 4. Section 5 introduces the state space representations that are used in the remaining sections. The second example architecture is described in Sect. 6. The digital estimation filter is described in Sect. 7. The actual derivation of this filter is outlined in the Appendix.

[1] This is common knowledge among designers, but textbooks seem to be taciturn about it.

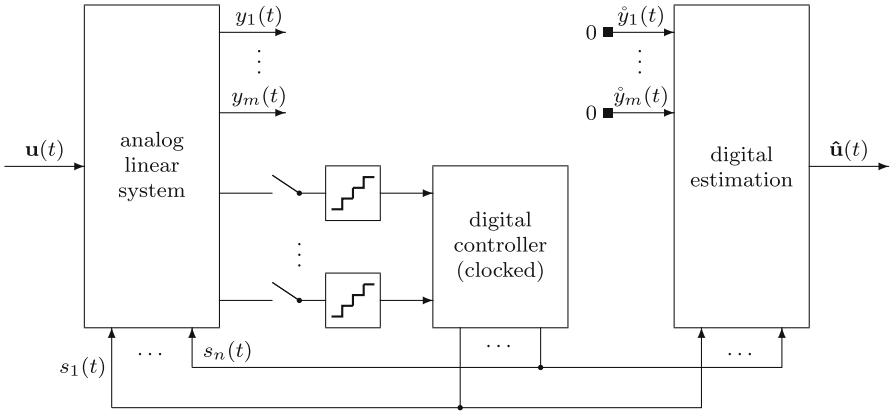

Fig. 1 Control-bounded analog-to-digital converter. The digital control signals $s_1(t), \ldots, s_n(t)$ remain constant between the ticks of the digital clock. The digital estimate $\hat{\mathbf{u}}(t)$ is mathematically defined in continuous time, but will practically be computed at discrete times $t_1, t_2, \ldots$

2 Operating Principle

2.1 Analog Part and Digital Control

Consider the system of Fig. 1. The continuous-time input signal is a scalar $u(t)$ or a vector

$$\mathbf{u}(t) \stackrel{\triangle}{=} \big(u_1(t),\ \ldots,\ u_k(t)\big)^{\mathsf{T}}. \tag{1}$$

The input signal is assumed to be bounded, i.e., $|u(t)| \le b_u$ or $|u_\ell(t)| \le b_u$ for all times t and all components $\ell = 1, \ldots, k$. This input signal is fed into a continuous-time analog linear system, which produces a continuous-time vector signal

$$\mathbf{y}(t) \stackrel{\triangle}{=} \big(y_1(t),\ \ldots,\ y_m(t)\big)^{\mathsf{T}}, \tag{2}$$

and the digital control in Fig. 1 ensures that

$$|y_\ell(t)| \le b_y \quad \text{for all } t \text{ and } \ell = 1, \ldots, m. \tag{3}$$

The digital control signals $s_1(t), \ldots, s_n(t)$ remain constant between the ticks of the digital clock. We will assume that the control is additive, i.e.,

$$\mathbf{y}(t) = \check{\mathbf{y}}(t) - \mathbf{q}(t), \tag{4}$$

where $\check{\mathbf{y}}(t)$ (given by (7)) is the fictional signal $\mathbf{y}(t)$ that would result without the digital control and where $\mathbf{q}(t)$ is fully determined by the control signals $s_1(t)$, …, $s_n(t)$. The dependence of $\mathbf{q}(t)$ on $s_1(t)$, …, $s_n(t)$ may be complicated, but we will never need (nor attempt) to determine $\mathbf{q}(t)$ explicitly.

At this point, we have already finished the discussion of the digital control in this section: its role and its effect are fully described by (3) and (4).

Note that both $\check{\mathbf{y}}(t)$ and $\mathbf{q}(t)$ are fictional signals that are not subject to any physical limits. In fact, the first key idea of control-bounded conversion is to use the approximation

$$\check{\mathbf{y}}(t) \approx \mathbf{q}(t). \tag{5}$$

Roughly speaking, the relative error of the approximation (5) can be made to vanish by letting the magnitudes of $\check{\mathbf{y}}(t)$ and $\mathbf{q}(t)$ grow to infinity while the difference (4) is kept small by (3).

We now assume that the uncontrolled analog filter is time-invariant and stable[2] with impulse response matrix

$$\mathbf{g}(t) \triangleq \begin{pmatrix} g_{1,1}(t) & \dots & g_{1,k}(t) \\ \vdots & \ddots & \vdots \\ g_{m,1}(t) & \dots & g_{m,k}(t) \end{pmatrix}, \tag{6}$$

where $g_{i,j}(t)$ is the impulse response from $u_j(t)$ to $y_i(t)$. We then have

$$\check{\mathbf{y}}(t) = (\mathbf{g} * \mathbf{u})(t) \tag{7}$$

$$\triangleq \begin{pmatrix} (g_{1,1} * u_1)(t) + \ldots + (g_{1,k} * u_k)(t) \\ \vdots \\ (g_{m,1} * u_1)(t) + \ldots + (g_{m,k} * u_k)(t) \end{pmatrix}. \tag{8}$$

We will also need the (elementwise) Fourier transform of (6), which will be denoted by $\mathbf{G}(\omega)$ and will be called analog transfer function (ATF) matrix.

2.2 Digital Estimation and Transfer Functions

Using the approximation (5), the digital estimation produces an estimate of $\mathbf{u}(t)$ from

$$\mathbf{q}(t) \approx (\mathbf{g} * \mathbf{u})(t), \tag{9}$$

which is a continuous-time deconvolution problem. The basic estimate is given by

$$\hat{\mathbf{u}}(t) \triangleq (\mathbf{h} * \mathbf{q})(t), \tag{10}$$

where $\mathbf{h}(t)$ is a matrix of stable impulse responses with (elementwise) Fourier transform given in (15).

Note that $\hat{\mathbf{u}}(t)$ is mathematically defined in continuous time, but it will in practice be computed at discrete times $t_1, t_2, \ldots$. These computations will be discussed in

[2] The extension of the following transfer function analysis to unstable analog systems is possible, but beyond the scope of this paper.

Sect. 7; it will be shown there that $\hat{\mathbf{u}}(t_1), \hat{\mathbf{u}}(t_2), \ldots$ can be computed with a digital linear filter directly from the digital control signals $s_1(t), \ldots, s_n(t)$, without actually computing $\mathbf{q}(t)$.

Using (4), the estimate (10) can be written as

$$\begin{aligned}\hat{\mathbf{u}}(t) &= (\mathbf{h} * \check{\mathbf{y}})(t) - (\mathbf{h} * \mathbf{y})(t) && (11)\\ &\approx (\mathbf{h} * \check{\mathbf{y}})(t) && (12)\\ &= (\mathbf{h} * \mathbf{g} * \mathbf{u})(t). && (13)\end{aligned}$$

Note that the step from (11) to (12) uses (5) or, equivalently, the approximation

$$\mathbf{y}(t) \approx \mathring{\mathbf{y}}(t) \triangleq \mathbf{0}, \tag{14}$$

as illustrated in Fig. 1.

The impulse response matrix $\mathbf{h}$ in (10) is determined by its (elementwise) Fourier transform

$$\mathbf{H}(\omega) \triangleq \mathbf{G}(\omega)^{\mathsf{H}} \left(\mathbf{G}(\omega)\mathbf{G}(\omega)^{\mathsf{H}} + \eta^2 \mathbf{I}_m\right)^{-1}, \tag{15}$$

where $(\cdot)^{\mathsf{H}}$ denotes Hermitian transposition, $\mathbf{I}_m$ is the m-by-m identity matrix, and $\eta > 0$ is a design parameter. The estimate (10) with $\mathbf{h}(t)$ as in (15) can be viewed as a statistical estimate or as the solution of a least-squares problem, as will be detailed in Sect. 2.3.1.

In the important special case where $\mathbf{u}(t)$ is scalar (i.e., $k = 1$), the ATF matrix $\mathbf{G}(\omega)$ is a column vector and $\mathbf{H}(\omega)$ is a row vector; in this case, using the matrix inversion lemma, (15) can be written as

$$\mathbf{H}(\omega) = \frac{\mathbf{G}(\omega)^{\mathsf{H}}}{\|\mathbf{G}(\omega)\|^2 + \eta^2} \tag{16}$$

and

$$\mathbf{H}(\omega)\mathbf{G}(\omega) = \frac{\|\mathbf{G}(\omega)\|^2}{\|\mathbf{G}(\omega)\|^2 + \eta^2}. \tag{17}$$

Note that $\mathbf{H}(\omega)\mathbf{G}(\omega) \approx 1$ for frequencies ω such that $\|\mathbf{G}(\omega)\| \gg \eta$ while $\mathbf{H}(\omega)\mathbf{G}(\omega) \approx 0$ for $\|\mathbf{G}(\omega)\| \ll \eta$.

Equations (11) and (13) can then be interpreted as follows. Eq. (13) is the signal path: the signal $\mathbf{u}(t)$ is filtered with the signal transfer function (STF) matrix

$$\mathbf{H}(\omega)\mathbf{G}(\omega) = \mathbf{G}(\omega)^{\mathsf{H}} \left(\mathbf{G}(\omega)\mathbf{G}(\omega)^{\mathsf{H}} + \eta^2 \mathbf{I}_m\right)^{-1} \mathbf{G}(\omega). \tag{18}$$

The second term in (11) is the conversion error

$$\begin{aligned}\epsilon(t) &\triangleq \hat{\mathbf{u}}(t) - (\mathbf{h} * \mathbf{g} * \mathbf{u})(t) && (19)\\ &= -(\mathbf{h} * \mathbf{y})(t), && (20)\end{aligned}$$

with $\mathbf{y}(t)$ bounded as in (3). Because of (20), $\mathbf{H}(\omega)$ will be called noise transfer function (NTF) matrix. The NTF (16) is the starting points of the performance analysis in Sect. 3.

Note that the STF (17) or (18) does not entail a phase shift and is free of aliasing (hence the title of [13]): the sampling in Fig. 1 (which is used for the digital control) affects the error signal (20), but not (13).

2.3 More About the Estimation Filter

2.3.1 Alternative Characterizations

The estimate (10) and (15) is further illustrated by noting that it is the solution of the continuous-time least-squares problem

$$\hat{\mathbf{u}}(t) = \lim_{\Delta\to\infty} \operatorname*{argmin}_{\mathbf{u}(t)} \left(\int_{t-\Delta}^{t+\Delta} \|\mathbf{y}(\tau)\|^2 \, d\tau + \eta^2 \int_{t-\Delta}^{t+\Delta} \|\mathbf{u}(\tau)\|^2 \, \mathrm{d}\tau \right), \tag{21}$$

where the minimization is subject to the constraints (4) and (7). The first term in (21) quantifies (14) while the second term in (21) is a regularizer.

Moreover, the estimate (10) and (15) coincides also with the Wiener filter that computes the LMMSE (linear minimum mean squared error) estimate of $\mathbf{u}(t)$ from $\mathbf{q}(t)$ under the assumptions that $y_1(t), \ldots, y_m(t)$ are independent white-noise signals with power σ_Y^2 and $u_1(t), \ldots, u_k(t)$ are independent white-noise signals with power $\sigma_U^2 = \sigma_Y^2/\eta^2$.

The proof of these claims is beyond the scope of this paper; for the essential ideas, we refer to [4] and the Appendix.

However, the estimate (10) and (15) is not justified by these characterizations, but by its practicality.

2.3.2 Bandwidth and the Parameter η

For the following discussion of the parameter η in (15), we restrict ourselves to the scalar-input case, where the STF and the NTF are given by (17) and (16), respectively. In this case, it is easily seen from (17) that η determines the bandwidth of the estimate (10). For example, assuming that $\|\mathbf{G}(\omega)\|_\infty$ decreases with $|\omega|$, the bandwidth is roughly given by $0 \le |\omega| \le \omega_{\text{crit}}$ with ω_{crit} determined by

$$\|\mathbf{G}(\omega_{\text{crit}})\| = \eta. \tag{22}$$

However, the bandwidth of the estimate may be reduced by postfiltering as mentioned in Sect. 2.3.3.

It is also worth noting that the parameter η equals the ratio of the STF (17) and the NTF at ω_{crit}

$$\left. \frac{\mathbf{H}(\omega)\mathbf{G}(\omega)}{\|\mathbf{H}(\omega)\|} \right|_{\omega=\omega_{\text{crit}}} = \eta, \tag{23}$$

as illustrated in Fig. 5.

2.3.3 Postfiltering

The basic estimate (10) need not be the final converter output. For example, an extra (digital!) anti-aliasing filter before sampling $\hat{\mathbf{u}}(t)$ at discrete times $t_1, t_2, \ldots$ will normally be advantageous. The integration of such an extra filter in the computations of the basic estimate is straightforward, cf. Sect. 7.3.

2.4 Remarks

We conclude this section with a number of remarks. First, we note that the conversion error (19) is not due to the quantizers in Fig. 1, but due to the approximation (14) or equivalently (5). In other words, the conversion error (19) is fundamentally unrelated to the precision of the quantizer circuits in Fig. 1 (except indirectly via the effectiveness of the digital control).

Second, we note that the details of the digital control (clock frequency, thresholds, etc.) do not enter the transfer function analysis of Sect. 2.2.

Third, the digital estimate (10) is fundamentally a continuous-time quantity, and the resulting STF (18) and (17) are exact continuous-time expressions. Sampling this estimate at discrete times may be required in most applications, but it is not essential to the converter in itself. In fact, nontrivial continuous-time digital signal processing (e.g., beamforming) can be done before any sampling, as suggested in [14, Section 10.2].

Forth, the digital estimation and the transfer function analysis of Sect. 2.2 work for arbitrary stable analog transfer functions $\mathbf{g}(t)$. In fact, stability of the uncontrolled analog system has here been assumed only for the sake of the analysis: the actual digital filter in Sect. 7 is indifferent to this assumption (hence the title of [10]).

Finally, the purpose of the analog linear system in Fig. 1 is not to prepare the input signal for quantization, but to amplify the input signal, over a frequency band of interest, into the vector signal (7) such that (5) is a good approximation. This very general setting offers design opportunities for the analog system/filter beyond the limitations of conventional $\Delta\Sigma$ modulators, as will be illustrated by the examples in Sects. 4 and 6.

3 Conversion Noise Analysis

In this section, we derive an expression (32) for the nominal conversion noise (assuming ideal analog circuits and no thermal noise) in terms of the amplitude response of the analog system. While the analysis in Sect. 2 was mathematically exact, we will here resort to approximations similar to those routinely made in the analysis of $\Delta\Sigma$ ADCs. We again restrict ourselves to the case where $\mathbf{u}(t)$ is scalar (i.e., $k = 1$) and will be denoted by $u(t)$.

3.1 SNR and Statistical Noise Model

The conversion performance can be expressed as the signal-to-noise ratio (SNR)

$$\mathrm{SNR} \triangleq \frac{S}{S_\mathrm{N}} \tag{24}$$

where S and S_N are the power of $\hat{u}(t)$ and the power of the conversion error (20), respectively, both within some frequency band $\mathcal{B}$ of interest.

The numerator in (24) depends, of course, on the input signal. A trivial upper bound is $S \leq b_u^2$, and for a full-scale sinusoid, we have

$$S = b_u^2/2. \tag{25}$$

As for the in-band power S_N of the conversion error (20), we begin by writing

$$\mathrm{E}\big[\epsilon(t)^2\big] = \frac{1}{2\pi}\int_{-\infty}^{\infty} \mathbf{H}(\omega)\mathbf{S}_{\mathbf{y}\mathbf{y}^\mathsf{T}}(\omega)\mathbf{H}(\omega)^\mathsf{H}\,\mathrm{d}\omega, \tag{26}$$

where $\mathbf{y}(t)$ is modeled as a stationary stochastic process with power spectral density matrix

$$\mathbf{S}_{\mathbf{y}\mathbf{y}^\mathsf{T}}(\omega) \triangleq \int_{-\infty}^{\infty} \mathrm{E}\Big[\mathbf{y}(t+\tau)\mathbf{y}(t)^\mathsf{T}\Big] e^{-i\omega\tau}\,\mathrm{d}\tau. \tag{27}$$

(These statistical assumptions cannot be literally true, but they are a useful model.) Restricting (26) to the frequency band $\mathcal{B}$ of interest, we have

$$S_\mathrm{N} = \frac{1}{2\pi}\int_{\mathcal{B}} \mathbf{H}(\omega)\mathbf{S}_{\mathbf{y}\mathbf{y}^\mathsf{T}}(\omega)\mathbf{H}(\omega)^\mathsf{H}\,\mathrm{d}\omega. \tag{28}$$

3.2 White-Noise Analysis

If $\mathbf{S}_{\mathbf{y}\mathbf{y}^\mathsf{T}}(\omega)$ in (28) is approximated by

$$\mathbf{S}_{\mathbf{y}\mathbf{y}^\mathsf{T}}(\omega) \approx \sigma^2_{\mathbf{y}|\mathcal{B}}\mathbf{I}_m, \tag{29}$$

we further obtain

$$S_\mathrm{N} \approx \frac{\sigma^2_{\mathbf{y}|\mathcal{B}}}{2\pi}\int_{\mathcal{B}} \mathbf{H}(\omega)\mathbf{H}(\omega)^\mathsf{H}\,\mathrm{d}\omega \tag{30}$$

$$= \frac{\sigma^2_{\mathbf{y}|\mathcal{B}}}{2\pi}\int_{\mathcal{B}} \frac{\|\mathbf{G}(\omega)\|^2}{\big(\|\mathbf{G}(\omega)\|^2+\eta^2\big)^2}\,\mathrm{d}\omega \tag{31}$$

$$\approx \frac{\sigma^2_{\mathbf{y}|\mathcal{B}}}{2\pi}\int_{\mathcal{B}} \frac{1}{\|\mathbf{G}(\omega)\|^2}\,\mathrm{d}\omega, \tag{32}$$

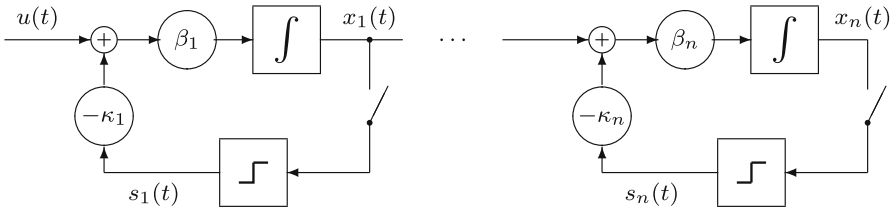

Fig. 2 Analog part and digital control of the example in Sect. 4 for $\rho_1 = \ldots = \rho_n = 0$

where the last step is justified by $\|\mathbf{G}(\omega)\| \geq \eta$ for $\omega \in \mathcal{B}$, cf. (17) and Sect. 2.3.2.

Note that the approximation (29) is restricted to $\mathcal{B}$ and is ultimately vindicated by the accuracy of (32). Using (32), the scale factor $\sigma^2_{\mathbf{y}|\mathcal{B}}$ can be determined by simulations.

It is obvious from (32) that a large SNR (24) requires a large analog amplification, i.e., $\|\mathbf{G}(\omega)\|$ must be large throughout $\mathcal{B}$.

4 A First Example: A Chain of Integrators

This example was first presented in [13], but it is here analyzed much further. Moreover, this example is the basis of the examples in Sect. 6. (An even more detailed analysis of this architecture as well as a prototype implementation is reported in [14] and [12].)

4.1 Analog Part and Digital Control

The analog part including the digital control is shown in Fig. 2. The input signal $u(t)$ is a scalar. The state variables $x_1(t), \ldots, x_n(t)$ obey the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t} x_\ell(t) = -\rho_\ell x_\ell(t) + \beta_\ell x_{\ell-1} - \kappa_\ell \beta_\ell s_\ell(t) \tag{33}$$

with $\rho_\ell \geq 0$, $\kappa_\ell \beta_\ell \geq 0$, and with $x_0(t) \triangleq u(t)$. The switches in Fig. 2 represent sample-and-hold circuits that are controlled by a digital clock with period T. The threshold elements in Fig. 2 produce the control signals $s_\ell(t) \in \{+1, -1\}$ depending on the sign of $x_\ell(kT)$ at sampling time kT immediately preceding t.

We will assume $|u(t)| \leq b$, and the system parameters will be chosen such that

$$|x_\ell(t)| \leq b \tag{34}$$

holds for $\ell = 1, \ldots, n$.

The control-bounded signals $y_1(t), \ldots, y_m(t)$ are selected from the state variables $x_1(t), \ldots, x_n(t)$ as will be discussed below.

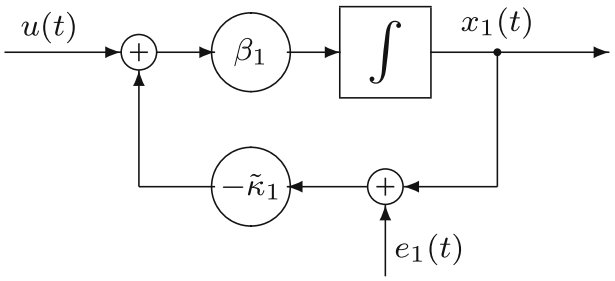

Fig. 3 Conventional view of the first stage in Fig. 2

4.2 Relation to MASH Converters

Figure 2 has some similarity with a continuous-time MASH $\Delta\Sigma$ modulator [8,16]. However, MASH converters are fundamentally built around the idea of passing only (or primarily) the quantization error of previous stages to the next stage. By contrast, Fig. 2 does not compute any quantization error signal at all, which is a significant simplification; in consequence, we conjecture that Fig. 2 can be implemented with lower power consumption than the analog part of a MASH converter.

Indeed, Fig. 2 cannot be handled by the digital cancellations schemes normally used in MASH converters. To see this, consider Fig. 3, which shows how the first stage in Fig. 2 would conventionally be modeled (perhaps with $\tilde{\kappa} \neq \kappa$), where $e_1(t)$ is the local quantization error [17]. Since $e_1(t)$ enters the system in exactly the same way as $u(t)$ (except for a scale factor), these two signals cannot be separated by any subsequent processing.

Nonetheless, the analysis in [14, Section 5.5.4] shows that Fig. 2 achieves essentially the same nominal performance as a MASH converter.

4.3 Conditions Imposed by the Digital Control

The bound (34) can be guaranteed by the conditions

$$|\kappa_\ell| \geq b \tag{35}$$

and

$$T|\beta_\ell|\big(|\kappa_\ell| + b\big) \leq b. \tag{36}$$

With the definition

$$\gamma_\ell \stackrel{\triangle}{=} T|\beta_\ell|, \tag{37}$$

(36) becomes

$$\gamma_\ell \leq \frac{b}{|\kappa_\ell| + b} \tag{38}$$

which implies $\gamma_\ell \leq 1/2$, and $\gamma_\ell = 1/2$ is admissible if and only if $|\kappa_\ell| = b$. In this case (i.e., if $|\kappa_\ell| = b$), the control frequency $1/T$ is admissible if and only if

$$1/T \geq 2|\beta_\ell|. \tag{39}$$

4.4 Transfer Functions

As mentioned, the control-bounded signals $y_1(t), \ldots, y_m(t)$ are selected from the state variables $x_1(t), \ldots, x_n(t)$. An obvious choice is $m = n$ and $y_1(t) = x_1(t), \ldots, y_n(t) = x_n(t)$. In this case, the ATF $\mathbf{G}(\omega) \triangleq \big(G_1(\omega), \ldots, G_n(\omega)\big)^{\mathsf{T}}$ of the uncontrolled analog system (as defined in Sect. 2) is given by

$$G_k(\omega) = \prod_{\ell=1}^{k} \frac{\beta_\ell}{i\omega + \rho_\ell}. \tag{40}$$

Another reasonable choice is $m = 1$ and $y_1(t) = x_n(t)$ as in [13]. In this case, the ATF is simply

$$\mathbf{G}(\omega) = \prod_{\ell=1}^{n} \frac{\beta_\ell}{i\omega + \rho_\ell}. \tag{41}$$

We now specialize to the case where $\beta_1 = \ldots = \beta_n = \beta$ and $\rho_1 = \ldots = \rho_n = \rho$, which makes the analysis more transparent. For $m = 1$ as in (41), we then have

$$\|\mathbf{G}(\omega)\|^2 = |G_n(\omega)|^2 = \left(\frac{\beta^2}{\omega^2 + \rho^2}\right)^n. \tag{42}$$

For $m = n$, we obtain

$$\|\mathbf{G}(\omega)\|^2 = \sum_{k=1}^{n} |G_k(\omega)|^2 \tag{43}$$

$$= \frac{1 - \left(\frac{\omega^2+\rho^2}{\beta^2}\right)^n}{\left(\frac{\omega^2+\rho^2}{\beta^2}\right)^n \left(1 - \frac{\omega^2+\rho^2}{\beta^2}\right)}. \tag{44}$$

Note that, for $\omega^2 + \rho^2 < \beta^2$, $|G_n(\omega)|^2$ as in (42) is the dominant term in (43). In consequence, $\mathbf{G}(\omega)$ as in (42) yields almost the same performance as (43).

For illustration, the amplitude responses $|G_1(\omega)|, \ldots, |G_n(\omega)|$ are plotted in Fig. 4 for $n = 5$, $\beta = 10$, and $\rho \in \{0,\ 0.03\beta\}$. Fig. 5 shows the resulting STF (17) and the components $H_1(\omega), \ldots, H_n(\omega)$ of the NTF (16) for $m = n$ (i.e., with $\|\mathbf{G}(\omega)\|$ as in (44)) and $\eta^2 = 104.3$.

From now on, we will normally assume $\rho = 0$ (i.e., undamped integrators).

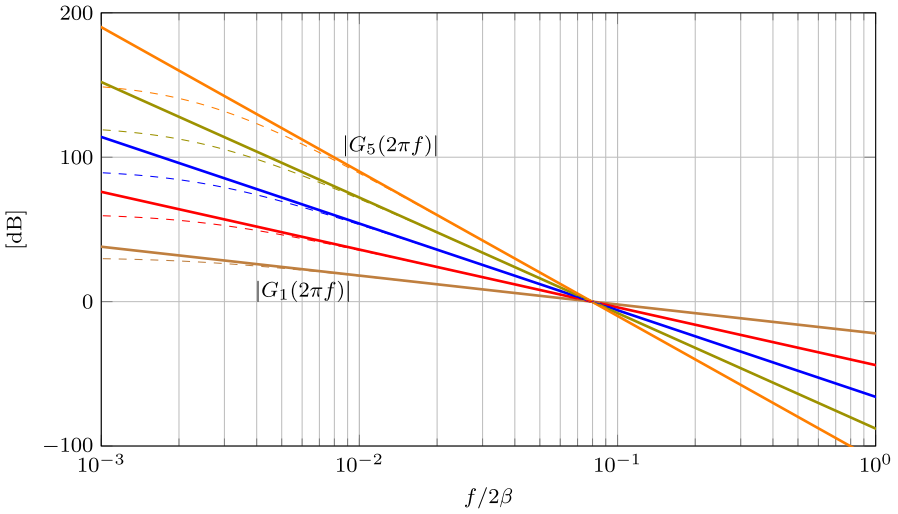

Fig. 4 Analog transfer functions (ATF) $|G_1(\omega)|, \ldots, |G_5(\omega)|$ of the example in Sect. 4.4, with $\rho = 0$ (solid) and some $\rho > 0$ (dashed). The frequency axis is normalized by the minimum control frequency (39)

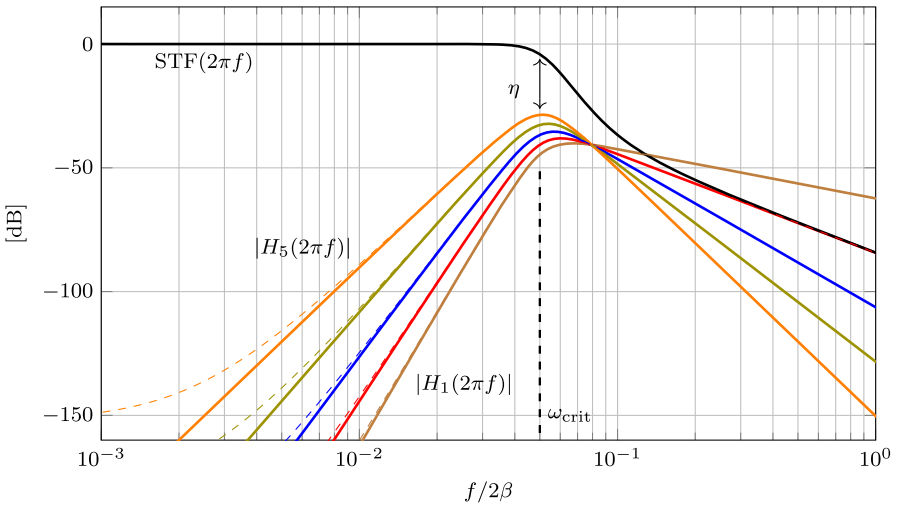

Fig. 5 Signal transfer function (STF) and noise transfer functions (NTF) of the example in Sect. 4.4, with $\rho = 0$ (solid) and some $\rho > 0$ (dashed). Also shown is the bandwidth parameter ω_{crit} from (22)

4.5 Bandwidth

Using (42) (with $\rho = 0$), the bandwidth ω_{crit} defined by (22) is easily determined to be

$$\omega_{\mathrm{crit}} = |\beta|/\eta^{\frac{1}{n}}. \tag{45}$$

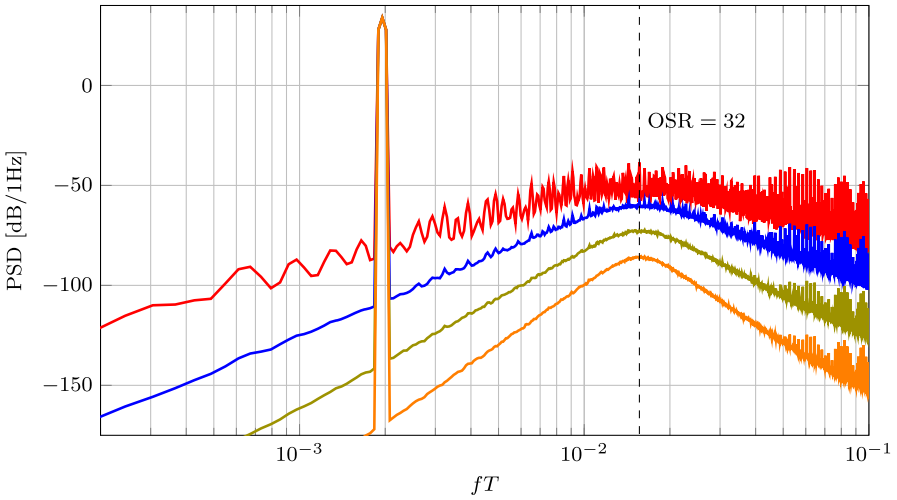

Fig. 6 Simulated power spectral density of the estimate $\hat{u}(t)$ for the example in Figs. 4 and 5 with $n = 2, \ldots, 5$ stages (from top to bottom), with OSR $= 32$, and with a full-scale sinusoidal input signal $u(t)$

For $\mathbf{G}(\omega)$ as in (44), Eq. (45) does not strictly hold, but it is a good proxy for the bandwidth also in this case.

In the following, we will use the quantity

$$\text{OSR} \triangleq \frac{1/T}{2f_{\text{crit}}} \tag{46}$$

with $f_{\text{crit}} \triangleq \omega_{\text{crit}}/(2\pi)$, which may be viewed as an analog of the oversampling ratio of $\Delta\Sigma$ converters. With (45) and with

$$\gamma \triangleq T|\beta| \tag{47}$$

as in (37), we then obtain

$$\eta = \left(\frac{\gamma}{\pi}\text{OSR}\right)^n. \tag{48}$$

Finally, we recall from Sect. 4.3 that stability can be guaranteed if and only if $\gamma \leq 1/2$.

4.6 Simulation Results

Figures 6 and 7 show the power spectral density (PSD) of the digital estimate $\hat{u}(t)$ for the numerical example in Figs. 4 and 5 with $\rho = 0$ and with further details as given below. In Fig. 6, the input signal $u(t)$ is a full-scale sinusoid; in Fig. 7, the input signal is $u(t) = 0$. Except for the peak in Fig. 6, both Figs. 6 and 7 thus show the PSD of the conversion error (19).

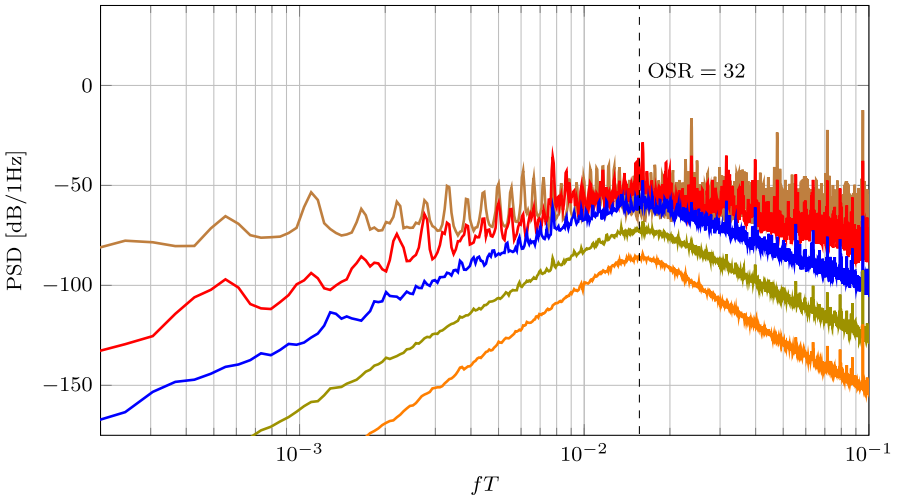

Fig. 7 Same as Fig. 6, but with input signal $u(t) = 0$ and for $n = 1, \ldots, 5$

As for the details in these simulations,[3] we have OSR $= 32$, $b = 1$, $\kappa = 1.05$, and $T = 1/21.5$, resulting in $\gamma = 10/21.5$. The frequency of the sinusoidal input signal is 0.1 Hz.

A key point of Figs. 6 and 7 is that the PSD of the conversion error appears to be well described by the white-noise analysis of Sect. 3.2.

4.7 Concluding Remarks

Throughout this section, we have just discussed the nominal performance of a converter with the structure of Fig. 2. A detailed discussion of circuit mismatch, thermal noise, etc., is beyond the scope of this paper, but given in [12,14]. Further contributions of [14] that are not reported here include a working hardware prototype and variations of the integrator chain including a chain of oscillators and a leapfrog structure.

5 State Space Representation

Both our further examples (in Sect. 6) and the digital estimation (in Sect. 7) require the analog linear system of Fig. 1 to be described in state space form. Specifically, we write

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{\Gamma}\mathbf{s}(t) \tag{49}$$

[3] Simulating the analog system requires to solve differential equations. We used the SciPy software package [18], which implements a Runge–Kutta method.

and

$$\mathbf{y}(t) = \mathbf{C}^{\mathsf{T}}\mathbf{x}(t), \tag{50}$$

where $\mathbf{x}(t)$ is the state vector, $\mathbf{s}(t) \triangleq \big(s_1(t), \ldots, s_n(t)\big)^{\mathsf{T}}$ comprises the digital control signals, and $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\boldsymbol{\Gamma}$, are matrices of suitable dimensions, The ATF matrix can then be written as

$$\mathbf{G}(\omega) = \mathbf{C}^{\mathsf{T}} \left(i\omega \mathbf{I}_n - \mathbf{A}\right)^{-1} \mathbf{B}. \tag{51}$$

For the example of Sect. 4, we have

$$\mathbf{A} = \mathbf{A}_{\mathrm{C}} \triangleq \begin{pmatrix} -\rho_1 & 0 & \ldots & \ldots & 0 \\ \beta_2 & -\rho_2 & 0 & \ddots & \vdots \\ 0 & \beta_3 & -\rho_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & \beta_n & -\rho_n \end{pmatrix}, \tag{52}$$

$\mathbf{B} = \mathbf{B}_{\mathrm{C}} \triangleq \big(\beta_1, 0, \ldots, 0\big)^{\mathsf{T}}$, and

$$\boldsymbol{\Gamma} = \boldsymbol{\Gamma}_{\mathrm{C}} \triangleq \begin{pmatrix} -\kappa_1\beta_1 & 0 & \ldots & 0 \\ 0 & -\kappa_2\beta_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & -\kappa_n\beta_n \end{pmatrix}. \tag{53}$$

If we choose $m = n$ and $y_1(t) = x_1(t)$, ..., $y_n(t) = x_n(t)$, we have $\mathbf{C}^{\mathsf{T}} = \mathbf{I}_n$; if, instead, we choose $m = 1$ and $y_1(t) = x_n(t)$, we have $\mathbf{C}^{\mathsf{T}} = (0, \ldots, 0, 1)$.

6 Hadamard Converters

The chain of integrators discussed in Sect. 4 provides excellent nominal performance (i.e., assuming ideal analog circuits). However, the real problem of analog-to-digital conversion is to efficiently cope with nonideal circuits. But every cascade structure, including that of Fig. 2, is sensitive to disturbances and imperfections at the early stage(s). In consequence, these early stage(s) need to be implemented with much higher precision (and therefore with much higher power consumption) than the later stages, which counteracts the apparent symmetry between the stages in Fig. 2.

We now show that the symmetry of the physical analog circuitry can be restored by a transformation of the state space. The resulting structure is conjectured to be more robust against disturbances and imperfections, as will be discussed in Sect. 6.4.

6.1 The Transform

For the transform, we use the orthogonal $n \times n$ matrix

$$\tilde{\mathbf{H}} \triangleq \frac{1}{\sqrt{n}}\mathbf{H}, \tag{54}$$

where $\mathbf{H}$ is a Hadamard matrix. (Other orthogonal matrices could be used, but the Hadamard matrix yields circuit friendly coefficients.)

The state space representation of the Hadamard converters is given by (49) and (50) with

$$\mathbf{A} = \tilde{\mathbf{H}}\mathbf{A}_{\mathrm{C}}\tilde{\mathbf{H}}^{\mathsf{T}}, \tag{55}$$

$$\mathbf{B} = \alpha\tilde{\mathbf{H}}\mathbf{B}_{\mathrm{C}} \tag{56}$$

$$= \frac{\alpha\beta_1}{\sqrt{n}}(1, \ldots, 1)^{\mathsf{T}}, \tag{57}$$

and

$$\mathbf{C}^{\mathsf{T}} = \alpha^{-1}\mathbf{C}_{\mathrm{C}}^{\mathsf{T}}\tilde{\mathbf{H}}^{\mathsf{T}} = \alpha^{-1}\tilde{\mathbf{H}}^{\mathsf{T}}, \tag{58}$$

where $\alpha > 0$ is a scale factor and we chose $\mathbf{C}_{\mathrm{C}} = \mathbf{I}_n$. (The digital control and the matrix $\mathbf{\Gamma}$ will be discussed below.) Note that (55)–(58) is just the chain of integrators in a different coordinate system. In particular, the ATF (51) is unchanged by this transformation. However, the circuit topology has changed: it is obvious from (57) that the input signal $u(t)$ is fed to all integrators equally and in parallel, and the matrix (55) is fully connected.

For example, for $n = 4$, the Hadamard matrix

$$\mathbf{H} = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \otimes \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{59}$$

$\beta_1 = \ldots = \beta_4 = \beta$, and $\rho_1 = \ldots = \rho_4 = 0$, we obtain

$$\mathbf{A} = \frac{\beta}{4}\begin{pmatrix} 3 & 1 & 1 & -1 \\ -1 & -3 & 1 & -1 \\ -1 & 1 & 1 & 3 \\ -1 & 1 & -3 & -1 \end{pmatrix}. \tag{60}$$

We also note from (51) that $\|\mathbf{G}(\omega)\|^2$ is unchanged if (58) is replaced by $\mathbf{C}^{\mathsf{T}} = \alpha^{-1}\mathbf{U}$, where $\mathbf{U}$ is an arbitrary orthogonal matrix. In fact, replacing (58) by $\mathbf{C}^{\mathsf{T}} = a\mathbf{U}$, with an arbitrary nonzero scale factor $a \in \mathbb{R}$, leaves the nominal conversion noise (32) unchanged (since the scale factor a enters quadratically both into $\|\mathbf{G}(\omega)\|^2$ and into $\sigma^2_{\mathbf{y}|\mathcal{B}}$). In particular, (58) can be replaced by

$$\mathbf{C}^{\mathsf{T}} = \mathbf{I}_n \tag{61}$$

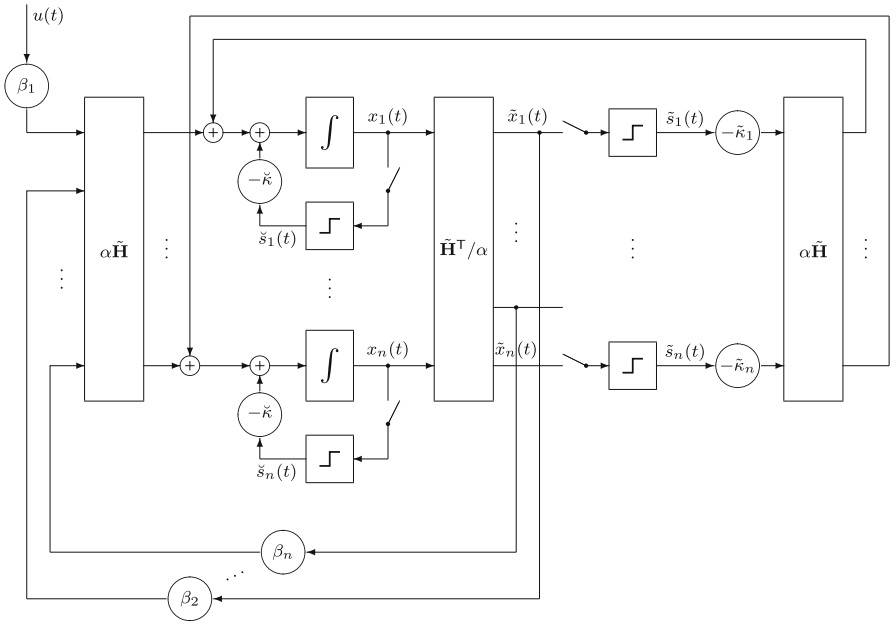

Fig. 8 Hadamard converters

with no effect on the nominal conversion noise.

6.2 Digital Control

The digital control can be effected in different ways, resulting in different Hadamard converters. In the following discussion, we refer to Fig. 8 and the symbols defined therein.

We also keep in mind that the mapping

$$\mathbb{R}^n \to \mathbb{R}^n : \xi \mapsto \tilde{\mathbf{H}}\xi \tag{62}$$

preserves the Euclidean norm of ξ, but it does not preserve bounds on the individual components of $\xi = \big(\xi_1, \ldots, \xi_n\big)^{\mathsf{T}}$. In fact, if ξ is only constrained by $|\xi_\ell| \le b$, $\ell \in \{1, \ldots, n\}$, then the best bound on the components of $\tilde{\mathbf{H}}\xi$ is $\sqrt{n}b$.

6.2.1 Integrator Chain Control (ICC)

This mode emulates the control of the chain of integrators (Fig. 2) using the $\{+1, -1\}$-valued control signals $\tilde{s}_1(t), \ldots, \tilde{s}_n(t)$ with $\tilde{\kappa}_\ell = \kappa_\ell \beta_\ell$, $\ell \in \{1, \ldots, n\}$; the control signals $\breve{s}_1(t), \ldots, \breve{s}_n(t)$ are not used (i.e., $\breve{\kappa}_1 = \ldots = \breve{\kappa}_n = 0$). The variables $\tilde{\mathbf{x}}(t) = \big(\tilde{x}_1(t), \ldots, \tilde{x}_n(t)\big)^{\mathsf{T}}$ in Fig. 8 are the outputs of the integrators in the chain (Fig. 2), which are related to the physical states $\mathbf{x}(t) = \big(x_1(t), \ldots, x_n(t)\big)^{\mathsf{T}}$ of the Hadamard

converter by

$$\mathbf{x}(t) = \alpha \tilde{\mathbf{H}} \tilde{\mathbf{x}}(t). \tag{63}$$

Choosing

$$\alpha = 1/\sqrt{n} \tag{64}$$

makes sure that all components of $\mathbf{x}$ are kept within the same limits as the components of $\tilde{\mathbf{x}}$.

6.2.2 Diagonal Control (DC)

In this mode, the integrator outputs $x_1(t), \ldots, x_n(t)$ in Fig. 8 are kept within an admissible range using the $\{+1, -1\}$-valued control signals $\check{s}_1(t), \ldots, \check{s}_n(t)$; the signals $\tilde{s}_1(t), \ldots, \tilde{s}_n(t)$ are not used (i.e., $\tilde{\kappa}_1 = \ldots = \tilde{\kappa}_n = 0$). In (49), this is expressed by

$$\mathbf{s}(t) = \check{\mathbf{s}}(t) \stackrel{\triangle}{=} \big(\check{s}_1(t), \ldots, \check{s}_n(t)\big)^{\mathsf{T}} \tag{65}$$

and $\mathbf{0}$ is a diagonal matrix with diagonal elements $-\check{\kappa}$.

For guaranteed stability, the analysis of Sect. 4.3 can be adapted as follows. For the sake of illustration, we here specialize to $n = 4$ and $\mathbf{A}$ as in (60). We assume $|u(t)| \le b$ and we wish to guarantee

$$|x_\ell(t)| \le b \tag{66}$$

for all $\ell \in \{1, \ldots, n\}$. Disregarding the control, it follows from (57) and (60) that the input of each integrator is upper bounded by

$$\zeta \stackrel{\triangle}{=} \frac{|\alpha\beta|b}{\sqrt{n}} + \frac{6|\beta|b}{4}. \tag{67}$$

Thus, (66) can be guaranteed by the conditions

$$|\check{\kappa}| \ge \zeta \tag{68}$$

and

$$T\big(|\check{\kappa}| + \zeta\big) \le b. \tag{69}$$

Conditions (68) and (69) can be simultaneously satisfied if and only if

$$|\beta| T \le \frac{1}{3 + |\alpha|}. \tag{70}$$

This bound is more restrictive than (39). However, extensive simulations have shown that (70) is overly pessimistic; in fact, even $|\beta|T = 1/2$ as in (39) appears to suffice (cf. Fig. 9, which will be discussed in Sect. 6.3).

6.2.3 Combined Control (CC)

The best results (to be detailed below) are obtained by using both the control signals $\check{s}_1(t), \ldots, \check{s}_n(t)$ and $\tilde{s}_1(t), \ldots, \tilde{s}_n(t)$ in Fig. 8. In this case, mathematical guarantees for $|x_\ell(t)| \leq b$ may be difficult to obtain, or too conservative to be useful.

6.3 Simulation Results

Figures 9, 10, 11, and 12 show some simulation results with these different control schemes. In all these simulations, we have $\beta_2 = \ldots = \beta_n = \beta$, $|\beta|T = 1/2$, $n = 4$, and $\alpha = 1/\sqrt{n} = 1/2$. Moreover, we have:

ICC: $\check{\kappa} = 0$, $\tilde{\kappa} = \beta$, and $\beta_1 = \beta$.
DC: $\check{\kappa} = \beta$, $\tilde{\kappa} = 0$, and $\beta_1 = 0.8\beta$.
CC: $\check{\kappa} = \tilde{\kappa} = \beta/\sqrt{2}$, and $\beta_1 = 0.8\beta$.

These choices for the parameters of DC and CC are heuristic.

Figure 9 shows the effectiveness of the different control schemes by showing histograms of $\max_{\ell\in\{1,\ldots,n\}}\{|x_\ell(t)|\}$ and of $\max_{\ell\in\{1,\ldots,n\}}\{|\tilde{x}_\ell(t)|\}$, sampled over the time t, for specific parameter settings as in Fig. 10. The corresponding histograms for Figs. 11 and 12 look quite similar.

Figures 10, 11, and 12 show the PSD of the digital estimate $\hat{u}(t)$. In Figs. 10 and 11, the input signal $u(t)$ is a full-scale sinusoid; in Fig. 12, the input signal is a small positive constant (specifically, $u(t) = 0.025$). The sharp peaks in Fig. 12 (and probably also in the other figures) are limit cycles.

From these figures, DC appears to offer little advantages while CC appears to be most attractive; in particular, CC can be used with very low OSR (46), where ICC fails due to limit cycles.

6.4 Robustness Against Nonidealities

So far, we have only considered the functionality of Hadamard converters with ideal circuits. However, our primary motivation for considering such converters is that we conjecture them to be potentially very robust against component mismatch and other nonidealities. This conjecture is suggested by the fact that Hadamard converters behave physically much like parallel structures, as is obvious from (57).

In order to demonstrate these robustness properties, we consider a possible hardware implementation as shown in Fig. 13.

This implementation uses a differential op-amp with capacitive feedback to facilitate the integrators of the Hadamard converter. The transformation $\tilde{\mathbf{H}}$ is realized using a resistor network as shown in Fig. 14.

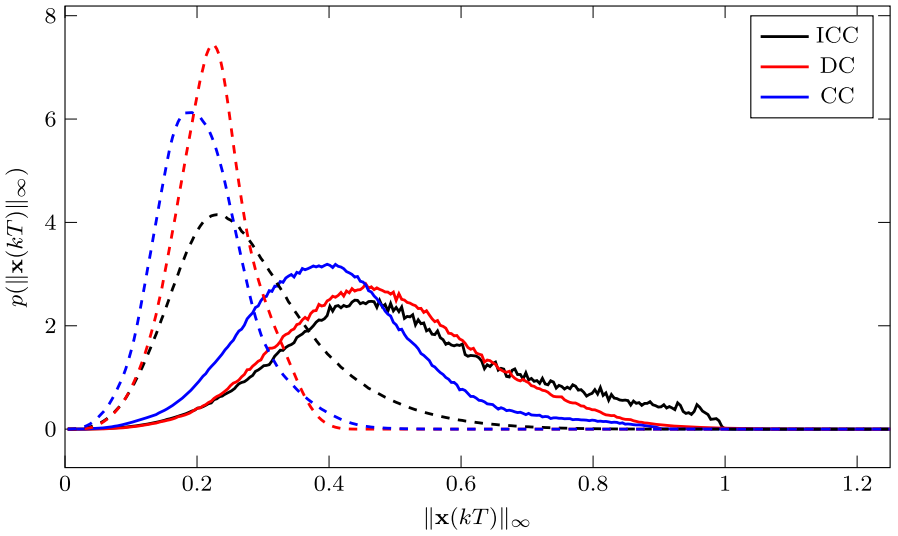

Fig. 9 Histogram of maximal component amplitudes of $\mathbf{x}(t)$ (dashed) and $\tilde{\mathbf{x}}(t)$ (solid), for integrator chain control (ICC), diagonal control (DC), and combined control (CC)

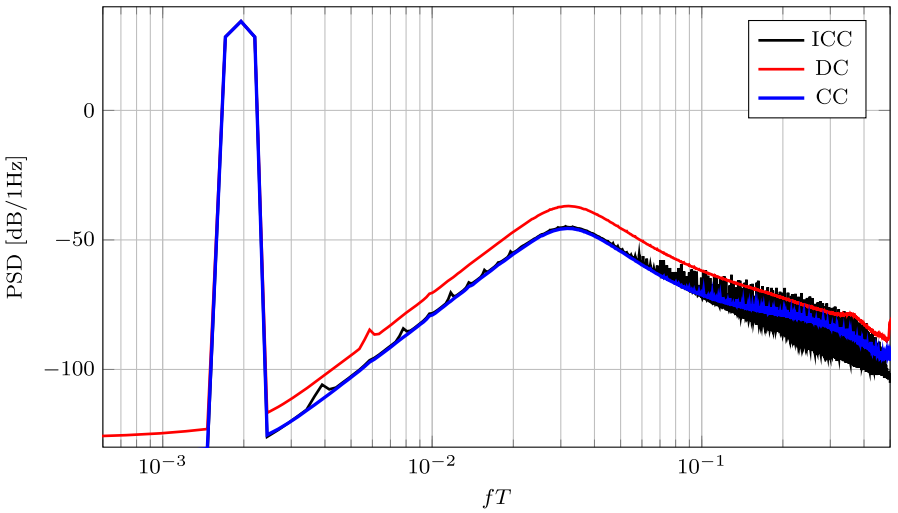

Fig. 10 Simulated power spectral density of the estimate $\hat{u}(t)$ of Hadamard converters of order $n = 4$ with a full-scale sinusoidal input signal $u(t)$ and OSR $= 16$

The global resistive values R and the capacitors values C, in the feedback path from the op-amps, are chosen such that

$$\frac{2}{RC} = \beta. \tag{71}$$

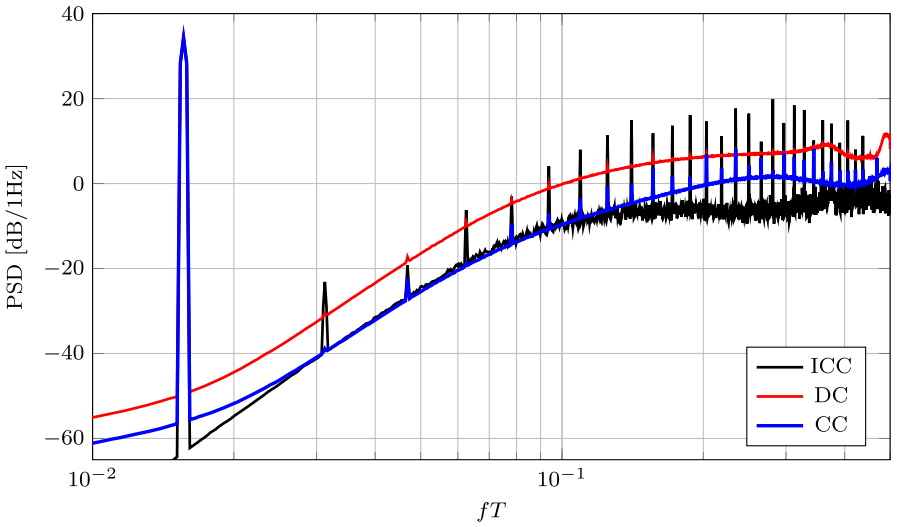

Fig. 11 Same as Fig. 10, but with OSR = 2

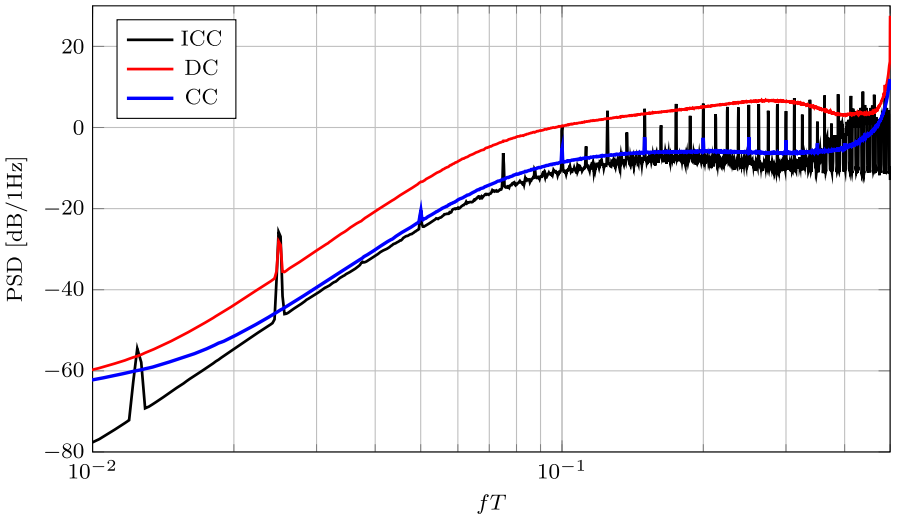

Fig. 12 Same as Fig. 11, but with (very small) constant input $u(t)$

We now consider the following mismatch scenario. The resistors are independently drawn from a uniform distribution with support of $\pm 1\%$ deviation from their respective nominal values. The same scenario is repeated for the chain-of-integrators hardware realization from [12]. The resulting PSDs of the estimate, averaged over 500 such simulations, are shown in Fig. 15. It is obvious that the Hadamard converter effectively suppresses the harmonic distortion caused by the mismatch and results in better SNR performance.

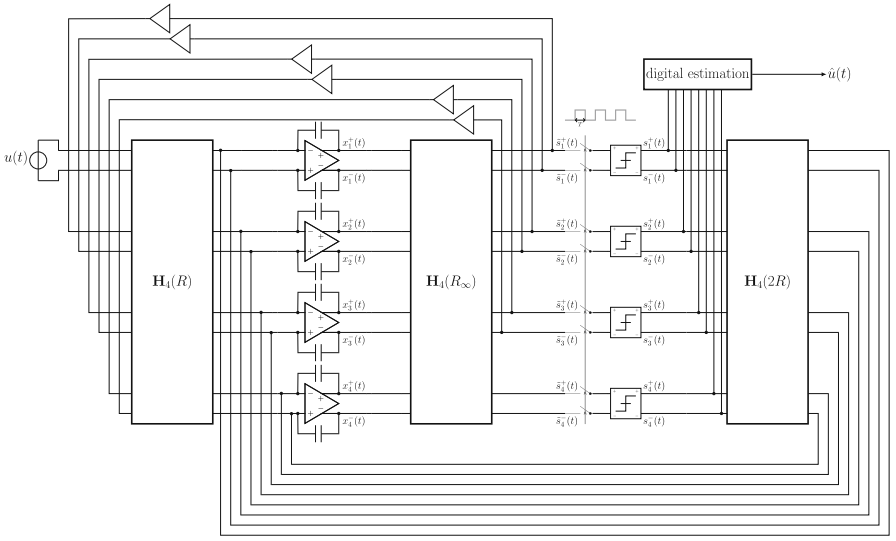

Fig. 13 Circuit implementation of the control-bounded Hadamard converter for $n = 4$. An implementation for the resistor networks $\mathbf{H}_4(R)$ is shown in Fig. 14. Note that the integrators are implemented as differential amplifiers with capacitive feedback resulting in eight state vector voltages $x_1^+(t), x_1^-(t), \ldots, x_4^+(t), x_4^-(t)$. However, as the corresponding signals are represented as differential voltages, i.e., $\mathbf{x}_\ell(t) = x_\ell^+(t) - x_\ell^-(t)$, the state space order $m = n = 4$. The feedback capacitors are all of the same capacitive value C which is chosen, together with R, such that they agree with (71). Finally, R_∞ represents a resistor value that is substantially larger than R

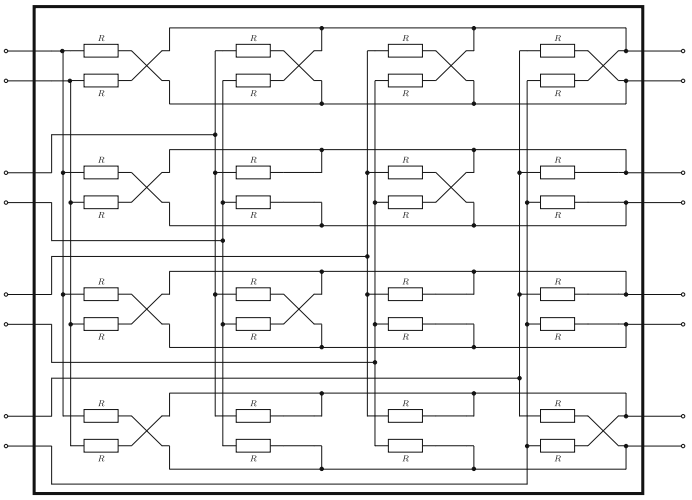

Fig. 14 A $\mathbf{H}_4(R)$ Hadamard resistor network where the k-th differential output is connected to the ℓ-th differential input via the k-th row ℓ-th column resistor pair in the figure

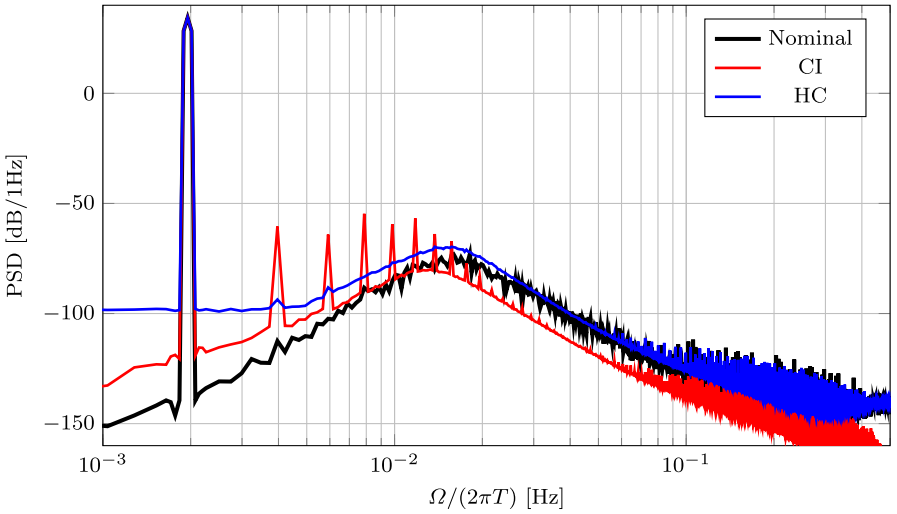

Fig. 15 Simulated averaged power spectral density of the estimate $\hat{u}(t)$ for the mismatch scenario in Sect. 6.4. The figure shows a chain-of-integrators converter (CI) as in [12], and a Hadamard converter (HC) with ICC control architecture as in Fig. 13. Also shown is the nominal (no-mismatch) performance

Of course, such simulations do not *prove* the conjectured robustness of the Hadamard converter in an actual implementation, but they support the conjecture.

7 Computing $\hat{\mathbf{u}}(t)$

The job of the digital estimation in Fig. 1 is to compute samples of the continuous-time estimate $\hat{\mathbf{u}}(t)$ defined by (10) and (15). At first sight, this computation looks daunting, involving not only the continuous-time convolution (10), but also the computation of $\mathbf{q}(t)$ from the control signals $s_1(t), \ldots, s_n(t)$.

It turns out, however, that samples of $\hat{\mathbf{u}}(t)$ can be computed quite easily and efficiently by the recursions given in Sect. 7.1. A brief derivation of these recursions is given in the Appendix; in outline, it involves the following steps.

The starting point is that the filter (15) is formally identical with the optimal filter (the Wiener filter) [1,9] for a certain statistical estimation problem (cf. Sect. 2.3.1). This same statistical estimation problem can also be solved by a variation of Kalman smoothing [9], which leads to recursions based on a state space model of the analog system. The precise form of the required Kalman smoother is not standard, as it combines input signal estimation as in [3] with a limit to continuous-time observations.

Throughout, we will use the state space representation of the analog system as in Sect. 5.

7.1 Basic Filter Algorithm

Assume that we wish to compute the basic estimate $\hat{\mathbf{u}}(t)$ given by (10) for $t = t_1, t_2, \ldots$. We will here restrict ourselves to regular sampling[4] with $t_k = kT_u$ such that T (the period of the clock in Figs. 1 and 2) is an integer multiple of T_u; in other words, we interpolate regularly between the ticks of the clock in Fig. 1. Moreover, we focus on the steady-state case $k \gg 1$ where border effects can be neglected. The algorithm consists of a forward recursion and a backward recursion.

Forward recursion: for $k = 0, 1, 2, \ldots$, compute the vectors $\overrightarrow{\mathbf{m}}_k$ (of the same dimension as $\mathbf{x}(t)$) by

$$\overrightarrow{\mathbf{m}}_{k+1} \triangleq \mathbf{A_f}\overrightarrow{\mathbf{m}}_k + \mathbf{B_f}\mathbf{s}(t_k) \tag{72}$$

starting from $\overrightarrow{\mathbf{m}}_0 \triangleq \mathbf{0}$.

The required matrices $\mathbf{A_f}$ and $\mathbf{B_f}$ will be given in Sect. 7.4.

Backward recursion: Compute the vectors $\overleftarrow{\mathbf{m}}_k$ (of the same dimension as $\mathbf{x}(t)$) by

$$\overleftarrow{\mathbf{m}}_k \triangleq \mathbf{A_b}\overleftarrow{\mathbf{m}}_{k+1} + \mathbf{B_b}\mathbf{s}(t_k) \tag{73}$$

starting from $\overleftarrow{\mathbf{m}}_N = \mathbf{0}$ for some $N > k$, as well as

$$\hat{\mathbf{u}}(t_k) = \mathbf{W}^{\mathsf{T}}\big(\overleftarrow{\mathbf{m}}_k - \overrightarrow{\mathbf{m}}_k\big)\,. \tag{74}$$

The required matrices $\mathbf{A_b}$ and $\mathbf{B_b}$ and the matrix $\mathbf{W}$ will be given in Sect. 7.4.

To be precise, (74) agrees with (10) only for $k \gg 0$ and $k \ll N$. In practice, however, $N - k$ need not be very large for (74) to be accurate, i.e., only a moderate delay (i.e., latency) is required.

7.2 FIR Filter Version

The computation of (74) can be formulated as a finite impulse response (FIR) filter. For $T_u = T$ (i.e., samples of $\hat{\mathbf{u}}(t)$ are produced at the clock rate), we thus obtain

$$\hat{\mathbf{u}}(t_k) \approx \sum_{\ell=-L_1}^{L_2} \tilde{\mathbf{h}}_\ell\, \mathbf{s}(t_{k-\ell}) \tag{75}$$

with coefficient matrices

$$\tilde{\mathbf{h}}_\ell \triangleq \begin{cases} \mathbf{W}^{\mathsf{T}}\mathbf{A}_{\mathbf{b}}^{\ell}\mathbf{B_b} & \text{if } \ell \le 0 \\ -\mathbf{W}^{\mathsf{T}}\mathbf{A}_{\mathbf{f}}^{-\ell+1}\mathbf{B_f} & \text{else,} \end{cases} \tag{76}$$

[4] In this section, we use k to index time steps, which is unrelated to the dimensionality of $\mathbf{u}(t)$ as in (1).

where $L_1 > 0$ and $L_2 > 0$ need to be chosen large enough such that the truncation of (74) to the finite sum (75) does not significantly affect the overall performance.

If the control signals $\mathbf{s}(t) = \big(s_1(t),\ \ldots,\ s_n(t)\big)^{\mathsf{T}}$ are $\{+1, -1\}$ valued, the computation of (75) requires $n(L_1 + L_2 + 1)$ additions (and no multiplications) per time index k for a scalar input signal (or for each scalar component of a vector input signal).

Multiple alternative ways to organize the computation of (74) are discussed in [14].

7.3 Decimation Filtering

In many applications, the clock rate samples (75) will be subsampled to a lower rate. In this case, including an anti-aliasing filter before subsampling (like in a $\Delta\Sigma$ converter) is mandatory. If the control signals $\mathbf{s}(t)$ are $\{+1, -1\}$ valued, combining the anti-aliasing filtering with the filtering (75) retains the multiplierless FIR filter structure.

7.4 Offline Computations

We now turn to the matrices $\mathbf{A_f}$, $\mathbf{B_f}$, $\mathbf{A_b}$, $\mathbf{B_b}$ and the matrix $\mathbf{W}$ in (72)–(74), which can be precomputed.

We first need the symmetric square matrices $\overrightarrow{\mathbf{V}}$ and $\overleftarrow{\mathbf{V}}$ (of the same dimension as $\mathbf{A}$) as follows. The matrix $\overrightarrow{\mathbf{V}}$ is the limit

$$\overrightarrow{\mathbf{V}} \stackrel{\triangle}{=} \lim_{\tau \to 0} \lim_{\ell \to \infty} \overrightarrow{\mathbf{V}}_\ell \tag{77}$$

of the iteration

$$\overrightarrow{\mathbf{V}}_{\ell+1} \stackrel{\triangle}{=} \overrightarrow{\mathbf{V}}_\ell + \tau\left(\mathbf{A}\overrightarrow{\mathbf{V}}_\ell + (\mathbf{A}\overrightarrow{\mathbf{V}}_\ell)^{\mathsf{T}} + \mathbf{B}\mathbf{B}^{\mathsf{T}} - \frac{1}{\eta^2}\overrightarrow{\mathbf{V}}_\ell \mathbf{C}\mathbf{C}^{\mathsf{T}}\overrightarrow{\mathbf{V}}_\ell\right); \tag{78}$$

equivalently, $\overrightarrow{\mathbf{V}}$ is the solution of the continuous-time algebraic Riccati equation

$$\mathbf{A}\overrightarrow{\mathbf{V}} + (\mathbf{A}\overrightarrow{\mathbf{V}})^{\mathsf{T}} + \mathbf{B}\mathbf{B}^{\mathsf{T}} - \frac{1}{\eta^2}\overrightarrow{\mathbf{V}}\mathbf{C}\mathbf{C}^{\mathsf{T}}\overrightarrow{\mathbf{V}} = \mathbf{0}. \tag{79}$$

The matrix $\overleftarrow{\mathbf{V}}$ is defined almost identically, but with a sign change in $\mathbf{A}$, i.e., $\overleftarrow{\mathbf{V}}$ is the solution of the continuous-time algebraic Riccati equation

$$\mathbf{A}\overleftarrow{\mathbf{V}} + (\mathbf{A}\overleftarrow{\mathbf{V}})^{\mathsf{T}} - \mathbf{B}\mathbf{B}^{\mathsf{T}} + \frac{1}{\eta^2}\overleftarrow{\mathbf{V}}\mathbf{C}\mathbf{C}^{\mathsf{T}}\overleftarrow{\mathbf{V}} = \mathbf{0}. \tag{80}$$

The matrix $\mathbf{W}$ in (74) is then obtained by solving the linear equation

$$(\overrightarrow{\mathbf{V}} + \overleftarrow{\mathbf{V}})\mathbf{W} = \mathbf{B} \tag{81}$$

for $\mathbf{W}$.

The matrix $\mathbf{A_f}$ in (72) is given by

$$\mathbf{A_f} \triangleq e^{(\mathbf{A}-\overrightarrow{\mathbf{V}}\mathbf{C}\mathbf{C}^\mathsf{T}/\eta^2)T_u} \tag{82}$$

and the matrix $\mathbf{A_b}$ in (73) is

$$\mathbf{A_b} \triangleq e^{-(\mathbf{A}+\overleftarrow{\mathbf{V}}\mathbf{C}\mathbf{C}^\mathsf{T}/\eta^2)T_u}. \tag{83}$$

Finally, the matrix $\mathbf{B_f}$ in (72) is

$$\mathbf{B_f} \triangleq \int_0^{T_u} e^{(\mathbf{A}-\overrightarrow{\mathbf{V}}\mathbf{C}\mathbf{C}^\mathsf{T}/\eta^2)(T_u-t)}\mathbf{\Gamma}\,\mathrm{d}t \tag{84}$$

and the matrix $\mathbf{B_b}$ in (73) is

$$\mathbf{B_b} \triangleq -\int_0^{T_u} e^{-(\mathbf{A}+\overleftarrow{\mathbf{V}}\mathbf{C}\mathbf{C}^\mathsf{T}/\eta^2)(T_u-t)}\mathbf{\Gamma}\,\mathrm{d}t. \tag{85}$$

Note that the only free parameter of the digital filter is η^2 as in (15).

Care must be taken that the quantities of this section are computed with sufficient numerical precision, and the matrices $\overrightarrow{\mathbf{V}}$ and $\overleftarrow{\mathbf{V}}$ should be exactly symmetric.

For the example of Sect. 4 (and Fig. 2) with $n = 2$ and $\rho = 0$, the quantities in (81) turn out to be

$$\overrightarrow{\mathbf{V}} = \begin{pmatrix} \beta\sqrt{2\eta} & \beta\eta \\ \beta\eta & \beta\eta\sqrt{2\eta} \end{pmatrix}, \tag{86}$$

$$\overleftarrow{\mathbf{V}} = \begin{pmatrix} \beta\sqrt{2\eta} & -\beta\eta \\ -\beta\eta & \beta\eta\sqrt{2\eta} \end{pmatrix}, \tag{87}$$

and $\mathbf{W} = \frac{1}{2\sqrt{2\eta}}(1,0)^\mathsf{T}$, which may be a useful test case for numerical computations.

8 Conclusion

Control-bounded conversion is a new type of analog-to-digital conversion where a digital estimate of the continuous-time analog input signal(s) is obtained from a principled solution of a natural inverse problem. We have developed the fundamentals of such converters, including a transfer function analysis and the implementation of the digital estimate as a practical linear filter. The flexibility of the digital control and estimation allows to use more general analog filter structures than conventional converters.

We gave two examples of such architectures. The first example is a chain of integrators (first proposed in [13]), which is reminiscent of a continuous-time MASH ADC, but with a simpler analog part that cannot be satisfactorily handled by a conventional digital cancellation scheme. The second example is obtained from the first example

by a transformation of the state space, resulting in essentially the same nominal performance, but with a fully connected physical structure that is conjectured to be more robust against component mismatch and other nonidealities.

Funding Open access funding provided by Swiss Federal Institute of Technology Zurich.

Availability of data and materials Data sharing is not applicable to this article as no datasets were generated or analyzed during the current work.

Declarations

Conflict of interest Not applicable.

9 Appendix

In this appendix, we give a condensed derivation of the algorithm of Sect. 7. (A detailed development of all the required background is beyond the scope of this paper.)

As mentioned in Sect. 2.3.1, the filter (15) can be viewed as a multivariate extension of the continuous-time Wiener filter [1] that estimates a multivariate zero-mean white Gaussian noise "signal" $\mathbf{U}(t)$ from the signal

$$\tilde{\mathbf{Y}}(t) \stackrel{\triangle}{=} (\mathbf{g} * \mathbf{U})(t) + \mathbf{Z}(t), \tag{88}$$

where $\mathbf{Z}(t)$ is m-dimensional zero-mean white Gaussian noise that is independent of $\mathbf{U}(t)$. In this statistical model, the average

$$\tilde{\mathbf{U}}(t, \Delta) \stackrel{\triangle}{=} \frac{1}{\Delta} \int_{t-\Delta}^{t} \mathbf{U}(\tau)\, \mathrm{d}\tau \tag{89}$$

(for $\Delta > 0$) is a K-dimensional[5] zero-mean Gaussian random variable with covariance matrix $\frac{\sigma_U^2}{\Delta}\mathbf{I}_K$. The covariance matrix $\frac{\sigma_Z^2}{\Delta}\mathbf{I}_m$ of $\mathbf{Z}(t)$ is defined analogously.

By "estimating $\mathbf{U}(t)$," we really mean to estimate the random variable(s) (89) for any fixed t, and then taking the limit $\Delta \to 0$ [4]. In this setting, the MAP estimate, the MMSE estimate, and the LMMSE estimate agree and equal the mean of the posterior distribution of $\tilde{\mathbf{U}}(t, \Delta)$ conditioned on the observation of $\tilde{\mathbf{Y}}(t)$. The Wiener filter

[5] In this appendix, we use K, rather than k as in (1), to denote the number of input signals.

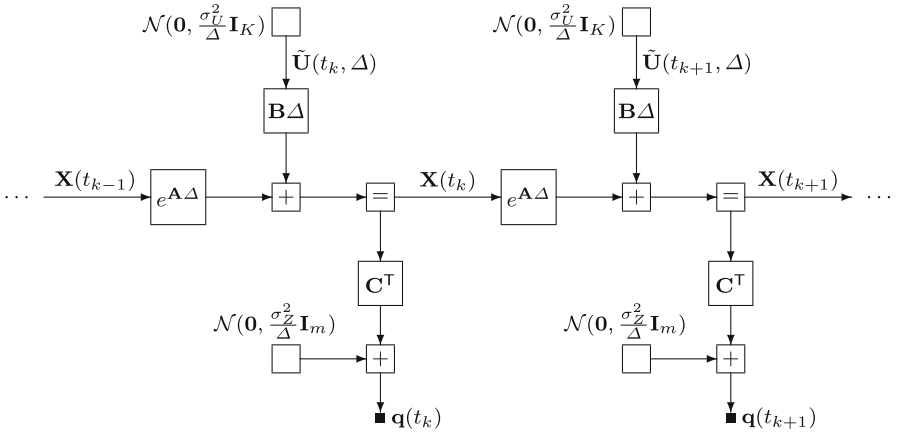

Fig. 16 Two sections of the factor graph of the (uncontrolled) state space model. The total factor graph consists of many such sections (perhaps with initial and final conditions, which we can ignore in this paper). A box labeled "$\mathcal{N}(\mathbf{m}, \mathbf{6})$" represents a multivariate Gaussian density with mean vector $\mathbf{m}$ and covariance matrix $\mathbf{6}$, $\mathbf{0}$ refers to an all zero vector of appropriate dimensions, and a small filled box represents a known quantity; all other boxes represent linear equations. This factor graph representation is exact only in the limit $\Delta = t_k - t_{k-1} \to 0$

computes this estimate (for $\Delta \to 0$) as

$$\hat{\mathbf{U}}(t) = (\mathbf{h} * \tilde{\mathbf{Y}})(t) \tag{90}$$

where the Fourier transform of $\mathbf{h}(t)$ is (15) with

$$\eta^2 = \sigma_Z^2/\sigma_U^2. \tag{91}$$

Applying this Wiener filter to the signal $\mathbf{q}(t)$ as in (10) means that we solve the statistical estimation problem for the observation $\tilde{\mathbf{Y}}(t) = \mathbf{q}(t)$.

The same statistical estimation problem can also be solved by a variation of Kalman smoothing (or by an equivalent variation of recursive least squares, cf. (21)). In contrast to the Wiener filter, the Kalman approach is based on the state space equations (49) and (50), which leads to recursive estimation algorithms. We will use a discrete-time approximation of the state space model with discrete times[6] $t_1, t_2, \ldots$ and fixed $t_k - t_{k-1} = \Delta > 0$; our continuous-time results will then be obtained by taking the limit $\Delta \to 0$.

From now on, we will use factor graphs as in [11], which allow to compose recursive estimation algorithms from lookup tables of "local" computations. A factor graph of (the discrete-time approximation of) our statistical model in state space form is shown in Fig. 16. Note that Fig. 16 represents the uncontrolled analog system with the observations $\tilde{\mathbf{Y}}(t_k) = \mathbf{q}(t_k)$.

[6] The discrete times $t_1, t_2, \ldots$ in this appendix (with $t_k - t_{k-1} = \Delta \to 0$) are unrelated to the discrete time steps in Sect. 7.

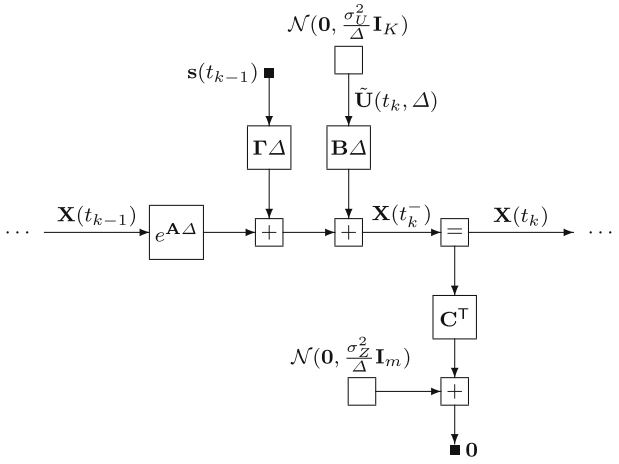

Fig. 17 One section of the factor graph of the state space model with plugged-in digital control signals $\mathbf{s}(t)$. The representation is exact only in the limit $\Delta = t_k - t_{k-1} \to 0$, where $e^{\mathbf{A}\Delta} \to \mathbf{I}_n + \mathbf{A}\Delta$

Now we plug in the (known and piecewise constant) control signals $\mathbf{s}(t) = (s_1(t), \ldots, s_n(t))$ into the state space model. We thus obtain the factor graph of Fig. 17, where all the observed signals are now zero, cf. (14). This second factor graph is easy to work with, and to take the $\Delta \to 0$ to continuous time at the end.

Using the notation of [11], we now consider the quantities $\overrightarrow{\mathbf{m}}_{\mathbf{X}(t)}$ and $\overrightarrow{\mathbf{V}}_{\mathbf{X}(t)}$ as well as $\overleftarrow{\mathbf{m}}_{\mathbf{X}(t)}$ and $\overleftarrow{\mathbf{V}}_{\mathbf{X}(t)}$. The former denote the mean vector and the covariance matrix, respectively, of the forward sum-product message, which equals the Gaussian probability density of the time-t state $\mathbf{X}(t)$ given past observations (up to a scale factor); the latter denote the mean vector and the covariance matrix, respectively, of the backward sum-product message, which equals the likelihood of the (given) future observations conditioned on $\mathbf{X}(t)$ (up to a scale factor).

From Fig. 17, we determine these quantities using Tables II–IV of [11] as follows. From (III.1) and (II.7) of [11], we have

$$\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k^-)} = e^{\mathbf{A}\Delta}\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_{k-1})}(e^{\mathbf{A}\Delta})^\mathsf{T} + \sigma_U^2\Delta\mathbf{B}\mathbf{B}^\mathsf{T}, \tag{92}$$

and from (IV.2) and (IV.3) of [11], we have

$$\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k)} = \overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k^-)} - \overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k^-)}\mathbf{C}\left(\frac{\sigma_Z^2}{\Delta}\mathbf{I}_o + \mathbf{C}^\mathsf{T}\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k^-)}\mathbf{C}\right)^{-1}\mathbf{C}^\mathsf{T}\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k^-)} \tag{93}$$

For $\Delta \approx 0$, we have

$$e^{\mathbf{A}\Delta} \approx \mathbf{I}_n + \Delta\mathbf{A}; \tag{94}$$

thus (92) becomes

$$\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k^-)} \approx \overrightarrow{\mathbf{V}}_{\mathbf{X}(t_{k-1})} + \Delta\left(\mathbf{A}\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_{k-1})} + (\mathbf{A}\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_{k-1})})^{\mathsf{T}} + \sigma_U^2\mathbf{B}\mathbf{B}^{\mathsf{T}}\right) \tag{95}$$

and (93) becomes

$$\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k)} \approx \overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k^-)} - \frac{\Delta}{\sigma_Z^2}\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k^-)}\mathbf{C}\mathbf{C}^{\mathsf{T}}\overrightarrow{\mathbf{V}}_{\mathbf{X}(t_k^-)}. \tag{96}$$

Combining (95) and (96) yields (77)–(79) as the steady-state condition for

$$\overrightarrow{\mathbf{V}} \triangleq \overrightarrow{\mathbf{V}}_{\mathbf{X}(t)}/\sigma_U^2 \tag{97}$$

in the limit $\Delta \to 0$.

The derivation of (80) is essentially identical except that the matrix $e^{\mathbf{A}\Delta}$ is replaced by its inverse, which amounts to a sign change in $\mathbf{A}$.

As for $\overrightarrow{\mathbf{m}}_{\mathbf{X}(t)}$, we have

$$\overrightarrow{\mathbf{m}}_{\mathbf{X}(t_k^-)} = e^{\mathbf{A}\Delta}\overrightarrow{\mathbf{m}}_{\mathbf{X}(t_k)} + \mathbf{\Gamma}\mathbf{s}(t_{k-1})\Delta \tag{98}$$

from (III.2) and (II.9) of [11], and

$$\overrightarrow{\mathbf{m}}_{\mathbf{X}(t_k)} = \overrightarrow{\mathbf{m}}_{\mathbf{X}(t_k^-)} - \overrightarrow{\mathbf{V}}_{X(t_k^-)}\mathbf{C}\left(\frac{\sigma_Z^2}{\Delta}\mathbf{I}_o + \mathbf{C}^{\mathsf{T}}\overrightarrow{\mathbf{V}}_{X(t_k^-)}\mathbf{C}\right)^{-1}\mathbf{C}^{\mathsf{T}}\overrightarrow{\mathbf{m}}_{\mathbf{X}(t_k^-)} \tag{99}$$

from (IV.1) and (IV.3) of [11]. For $\Delta \approx 0$, we obtain with (94)

$$\overrightarrow{\mathbf{m}}_{\mathbf{X}(t_k)} = \overrightarrow{\mathbf{m}}_{\mathbf{X}(t_{k-1})} + \Delta\Big(\mathbf{A}\overrightarrow{\mathbf{m}}_{\mathbf{X}(t_{k-1})} + \mathbf{\Gamma}\mathbf{s}(t_{k-1}) - \frac{1}{\eta^2}\overrightarrow{\mathbf{V}}\,\mathbf{C}\mathbf{C}^{\mathsf{T}}\overrightarrow{\mathbf{m}}_{\mathbf{X}(t_{k-1})}\Big), \tag{100}$$

where we have used the normalized stationary covariance matrix (97). Note that (100) is exact in the limit $\Delta \to 0$ and amounts to the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\overrightarrow{\mathbf{m}}_{\mathbf{X}(t)} = \left(\mathbf{A} - \frac{1}{\eta^2}\overrightarrow{\mathbf{V}}\,\mathbf{C}\mathbf{C}^{\mathsf{T}}\right)\overrightarrow{\mathbf{m}}_{\mathbf{X}(t)} + \mathbf{\Gamma}\mathbf{s}(t). \tag{101}$$

The solution of this differential equation (for $t > 0$) is

$$\overrightarrow{\mathbf{m}}_{\mathbf{X}(t)} = e^{\tilde{\mathbf{A}}t}\overrightarrow{\mathbf{m}}_{\mathbf{X}(0)} + e^{\tilde{\mathbf{A}}t}\int_0^t e^{-\tilde{\mathbf{A}}\tau}\mathbf{\Gamma}\mathbf{s}(\tau)\,\mathrm{d}\tau \tag{102}$$

with $\tilde{\mathbf{A}} \triangleq \mathbf{A} - \overrightarrow{\mathbf{V}}\,\mathbf{C}\mathbf{C}^{\mathsf{T}}/\eta^2$. This solution applies to any interval between t_k and t_{k+1} in Sect. 7.1 and yields (72) with (82) and (84).

The derivation for $\overleftarrow{\mathbf{m}}_{\mathbf{X}(t)}$ is essentially identical except for a sign change in both $\mathbf{A}$ and $\mathbf{\Gamma}$, where the latter is due to (II.10) of [11].

Finally, we use the result from [3] that the MAP/MMSE/LMMSE estimate of $U(t)$ (i.e., the posterior mean of (89) for $\Delta \to 0$) is given by

$$\hat{\mathbf{u}}(t) = \sigma_U^2 \mathbf{B}^{\mathsf{T}} \tilde{\mathbf{W}}(t) \left(\overleftarrow{\mathbf{m}}_{X(t)} - \overrightarrow{\mathbf{m}}_{X(t)} \right) \tag{103}$$

with

$$\tilde{\mathbf{W}}(t) \triangleq \left(\overrightarrow{\mathbf{V}}_{\mathbf{X}(t)} + \overleftarrow{\mathbf{V}}_{\mathbf{X}(t)} \right)^{-1}, \tag{104}$$

which yields (74) and (81). Note that (103) and (104) may also be obtained directly from Fig. 17 using (II.12), (III.8), and (III.9) of [11] and then taking the limit $\Delta \to 0$.

References

1. B.D.O. Anderson, J.B. Moore, *Optimal Filtering* (Prentice Hall, Hoboken, 1979)
2. J. Biveroni, H.-A. Loeliger, On sequential analog-to-digital conversion with low-precision components, in *2008 Information Theory & Applications Workshop*, UCSD, La Jolla, CA, January 27–February 1 (2008)
3. L. Bolliger, H.-A. Loeliger, C. Vogel, LMMSE estimation and interpolation of continuous-time signals from discrete-time samples using factor graphs, arXiv:1301.4793
4. L. Bruderer, H.-A. Loeliger, Estimation of sensor input signals that are neither bandlimited nor sparse, in *2014 Information Theory & Applications Workshop (ITA)*, San Diego, CA, February 9–14 (2014)
5. T.C. Carusone, D.A. Johns, K.W. Martin, *Analog Integrated Circuit Design*, 2nd edn. (Wiley, Hoboken, 2012)
6. I. Daubechies, R. A. DeVore, C. S. Güntürk, V. A. Vaishampayan, Beta expansions: a new approach to digitally corrected A/D conversion, in *Proceedings of 2002 IEEE International Symposium on Circuits and Systems (ISCAS),* May 26–29, (2002), pp. II-784–II-787
7. I. Daubechies, R.A. DeVore, C.S. Güntürk, V.A. Vaishampayan, A/D conversion with imperfect quantizers. IEEE Trans. Inf. Theory **52**(3), 874–885 (2006)
8. J.M. de la Rosa, Sigma-delta modulators: tutorial overview, design guide, and state-of-the-art survey. IEEE Trans. Circuits Syst. I **58**(1), 1–21 (2011)
9. T. Kailath, A.H. Sayed, B. Hassibi, *Linear Estimation* (Prentice Hall, Hoboken, 2000)
10. H.-A. Loeliger, L. Bolliger, G. Wilckens, J. Biveroni, Analog-to-digital conversion using unstable filters, in *2011 Information Theory & Applications Workshop (ITA),* UCSD, La Jolla, CA, USA, February 6–11 (2011)
11. H.-A. Loeliger, J. Dauwels, J. Hu, S. Korl, L. Ping, F.R. Kschischang, The factor graph approach to model-based signal processing. Proc. IEEE **95**(6), 1295–1322 (2007)
12. H.-A. Loeliger, H. Malmberg, G. Wilckens, Control-bounded analog-to-digital conversion: transfer function analysis, proof of concept, and digital filter implementation, arXiv:2001.05929
13. H.-A. Loeliger, G. Wilckens, Control-based analog-to-digital conversion without sampling and quantization, in *2015 Information Theory & Applications Workshop (ITA)*, UCSD, La Jolla, CA, USA, February 1–6 (2015)
14. H. Malmberg, *Control-Bounded Converters*, Ph.D. thesis at ETH Zurich no. 27025 (2020). https://doi.org/10.3929/ethz-b-000469192
15. O. Ordentlich, G. Tabak, P.K. Hanumolu, A.C. Singer, G.W. Wornell, A modulo-based architecture for analog-to-digital conversion. IEEE J. Sel. Top. Signal Process. **12**(5), 825–840 (2018)
16. M. Ortmanns, F. Gerfers, *Continuous-Time Sigma-Delta A/D Conversion: Fundamentals, Performance Limits and Robust Implementations* (Springer, Berlin, 2006)
17. S. Pavan, R. Schreier, G.C. Temes, *Understanding Delta-Sigma Data Converters*, 2nd edn. (Wiley, Piscataway, 2017)
18. P. Virtanen et al., SciPy 1.0: Fundamental algorithms for scientific computing in Python. Nat. Methods **17**, 261–272 (2020). https://doi.org/10.1038/s41592-019-0686-2