Accurate Time-Domain Simulation of Continuous-Time Sigma–Delta Modulators

Philippe Benabes

Accurate time-domain simulation of continuous-time sigma-delta modulators

Philippe Benabes, *Member, IEEE,*

***Abstract*—In this paper we present a methodology for the simulation of continuous-time sigma-delta converters. This method, based on a fixed-step algorithm, permits not only a time-domain simulation of the modulator output but also the simulation of intermediary signals. The method is based on the discretization of the continuous-time models and the use of a discrete simulator such as Simulink, which is more efficient than an analog simulator. By using filters with a sampling frequency higher than the modulator output frequency, the model can simulate input signals with a bandwidth which is higher than half the modulator sampling frequency. The transformation is exact in terms of Noise Transfer Function and asymptotically exact in terms of Signal Transfer Function (the Transfer Function from the modulator input to each stage filter output rapidly tends to the continuous-time model transfer function when the number of steps increases).**

***Index Terms*—sigma-delta, simulation, analog-to-digital conversion, continuous-time.**

I. Introduction

Sigma-Delta ($\Sigma\Delta$) circuits are very attractive analog-to-digital converters [1] as they achieve high accuracy with few critical analog components. The speed and integrator bandwidth limitations of switched-capacitor Discrete-Time (DT) modulators can be mitigated by the use of Continuous-Time (CT) modulators [2][3]. The latter are not easy to integrate, but possess one key advantage over their discrete-time competitors: no sampling is performed within the filter itself, allowing central frequencies in the Gigahertz range [4]. An implicit anti-aliasing is performed by the analog filter [5], thus reducing the constraints at the input of the converter. On the other hand, CT circuits are more difficult to design and to simulate than DT circuits. Classical design methodologies usually start with high-level design and simulations before transistor-level simulations [6][7]. The behavior of electronic components can be modeled at high level for example to analyze the impacts of environmental and process variations such as in [8]. When the high level-simulations are performed by an analog simulator, they require a very long computational time. An analytical integration of the differential equations can be performed such as in [9], but this method is exact only if the input signal can be expressed in an analytical form. An alternative but equivalent methodology consists in using an equivalent DT model of CT modulators [2]. Such a method supposes that the input signal of the CT filters has a particular shape (for example, a sampled and held signal) and thus the CT differential equations are solved for this particular signal. The output of the filter is calculated only at the A.D.C sampling times. However, the input bandwidth is limited to half the sampling frequency. As can be seen in [10], we can use a state space representation of the filters, or alternatively we can use direct s transfer functions as seen in [11]. In this paper, an extension of the simulation method of CT modulators based on Oversampled Discrete-Time (ODT) models [12] is presented. With this method, each sampling period is divided into a fixed number of steps. The outputs of the filters are then calculated for each sub-sampling time. The advantage of DT simulations over CT algorithms is that the outputs of the filters are calculated only at each sub-sampling time. CT algorithms (especially those with a variable step) calculate more samples than required, thus increasing the simulation time.

This transformation is exact in terms of Noise Transfer Function (NTF) and asymptotically exact in terms of Signal Transfer Function (STF). The STF of the model rapidly tends to the STF of the CT model when the number of steps increases. Simulations of the response of signals with a bandwidth higher than half the sampling frequency is then possible. Furthermore, an estimation of the ideal step-size derived from the bandwidth of the input signal will be performed.

This paper is structured as follows: Section II describes both, the synthesis and the analysis method for continuous filter modulators. An application of the simulation of a high order modulator with extra loop delay is illustrated in Section III. The extension to the simulation of intermediary signals is explained in section IV. Finally, concluding remarks are given in section V.

II. Synthesis and Simulation Methods of Continuous-Time Modulators

A. Equivalency between Continuous-Time and Discrete-Time Filters

The relationship between the CT $g(s)$ and the DT filter transfer function $f(z)$ of a $\Sigma\Delta$ modulator can be expressed using the well known formula [11],[13],

$$f(z) = (1 - z^{-1})\mathrm{Z}_{t=nT_s}\left\{L^{-1}\left[\frac{g(s)e^{-dT_s s}}{s}\right]\right\} \quad (1)$$

where d is the loop delay between the A.D.C input and the D.A.C output (normalized to the sampling period), Z stands for the Z transform, L^{-1} is the inverse Laplace Transform, and T_s is the sampling period. This formula ensures the equivalency of the Noise Transfer Function (NTF) between

The author is with the Department of Signal Processing and Electronics, Supélec, 91190 Gif/Yvette, FRANCE (e-mail: philippe.benabes@supelec.fr).

the CT modulator shown in Fig. 1 and the DT equivalent topology shown in Fig. 2. This formula is valid for square feedback signals but can be extended to any signal shape, see [14] or [15].

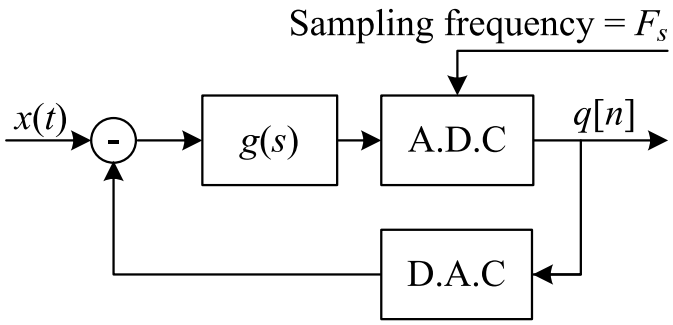

Fig. 1. CT ΣΔ modulator

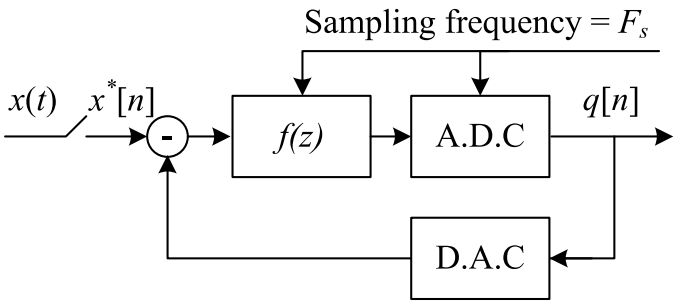

Fig. 2. Single-bit DT ΣΔ modulator

Formula (1) can be inverted and used for the synthesis of a CT modulator, i.e. in order to obtain the CT filter transfer function that ensures that the NTF of both modulators will be the same. When the delay d is non zero, (1) has no solution as a degree of freedom is missing, making it impossible to map the transfer function. An optimization method such as in [16] can be used to optimize the coefficients of the CT filters in order to maintain the stability and the performance of the loop. As explained in [17], in order to ensure the equivalency, a feedback term can also be added between D.A.C output and A.D.C input. All feedback techniques are summarized in[18].

With such a feedback term, (1) can now be written as:

$$f(z) = (1 - z^{-1})\mathrm{Z}_{t=nT_s}\left\{L^{-1}\left[\frac{g(s)e^{-dT_s s}}{s}\right]\right\} + T(z) \quad (2)$$

Using this methodology, the excess loop-delay [13] is no longer a hindrance, but becomes simply a parameter that can be adjusted [19], creating a compromise between the gain margin of the loop and the feasibility of the feedback D.A.C. The proposed method in [20] consists in decomposing $f(z)$ into rational fractions, and converting each term using pre-calculated solutions of (2). To obtain these solutions, we used Maple©software, as shown in annex A. The degree of $T(z)$ is the integer part of d plus one [21].

Table I gives solutions of (2) for several transfer functions and delays between 0 and 2. For the third-order, solutions may be found in [12] and [22]. They may also be achieved when using the Maple code given in annex A.

B. Signal transfer function

Assuming that the input signal is a band-limited signal (limited to the half of the sampling frequency), and that the quantizer can be modeled by an additive white noise, the signal transfer function of the DT topology can be expressed as:

$$STF_{DT}(\varphi) = \frac{f(e^{2j\pi\frac{\varphi}{F_S}})}{1 + f(e^{2j\pi\frac{\varphi}{F_S}})} \quad (3)$$

where F_s is the sampling frequency, and φ the signal frequency. The signal transfer function of the CT topology is [24]:

$$STF_{CT}(\varphi) = \frac{g(2j\pi\varphi)}{1 + f(e^{2j\pi\frac{\varphi}{F_S}})} \quad (4)$$

f being related to g by equation (1). It is clear that even if the DT topology can model the NTF of the modulator correctly, the STF obtained by (3) is different from (4). For example, with a first-order lowpass modulator, where $f(z) = \frac{z^{-1}}{1-z^{-1}}$, and $g(s) = \frac{F_S}{s}$, the DT signal transfer function is a simple delay:

$$STF_{DT}(z) = z^{-1} \Rightarrow STF_{DT}(\varphi) = e^{-2j\pi\frac{\varphi}{F_S}}. \quad (5)$$

This transfer function clearly differs from the CT transfer function which is:

$$STF_{CT}(\varphi) = \frac{1 - e^{2j\pi\frac{\varphi}{F_S}}}{2j\pi\frac{\varphi}{F_S}}. \quad (6)$$

As a consequence, when we deal with non constant signals, the DT model cannot be used to perform exact time-domain simulations of the behavior of a CT modulator. Furthermore, the DT model is unable to model an input signal with a frequency higher than half the sampling frequency. In order to enhance the signal-transfer function and remove the frequency limitation, we propose using an oversampled model of the DT modulator (ODT)

C. Oversampled model of a sigma delta modulator

Let us now consider the oversampled model of a ΣΔ modulator shown in Fig. 3. The sampling frequency of the A.D.C is still F_s, but the digital filter (F) runs now at kF_s. The feedback signal is held during k samples. This is a special case of multi-rate systems as seen in [23]. In order to simplify the notations, the Z variable denotes functions running at frequency kF_s, while the z variable denotes a function running at frequency F_s. A delay r is introduced at the output of the zero-order hold, in order to model the delay d between A.D.C input and D.A.C output. r will be chosen as the integer part of $k.d$ ($r = \lfloor k.d \rfloor$, where $\lfloor\rfloor$ denotes the integer part).

Next we consider the transfer function between the A.D.C output and its input. We calculate the response of the filter $Y^*[n]$ to a discrete impulse at the A.D.C output in the case of both: CT modulator, and ODT modulator.

In the case of the CT modulator:

$g(s)$	d	$f(z)$	$T(z)$
$\frac{1}{T_s s}$	$0 \le d \le 1$	$\frac{z^{-1}}{1-z^{-1}}$	dz^{-1}
$\frac{1}{T_s s}$	$1 \le d \le 2$	$\frac{z^{-1}}{1-z^{-1}}$	$z^{-1}+(d-1)z^{-2}$
$\frac{1}{T_s s-a}$	$0 \le d \le 1$	$e^{-ad}\frac{e^a-1}{a}\frac{z^{-1}}{1-e^a z^{-1}}$	$\frac{1-e^{-ad}}{a}z^{-1}$
$\frac{1}{T_s s-a}$	$1 \le d \le 2$	$e^{-ad}\frac{e^a-1}{a}\frac{z^{-1}}{1-e^a z^{-1}}$	$\frac{e^{-ad}(e^a-1)}{a}z^{-1}+\frac{1-e^{-a(d-1)}}{a}z^{-2}$
$\frac{1}{(T_s s)^2}$	$0 \le d \le 1$	$\frac{1-2d}{2}\frac{z^{-1}}{1-z^{-1}}+\left(\frac{z^{-1}}{1-z^{-1}}\right)^2$	$-\frac{d^2}{2}z^{-1}$
$\frac{1}{(T_s s)^2}$	$1 \le d \le 2$	$\frac{1-2d}{2}\frac{z^{-1}}{1-z^{-1}}+\left(\frac{z^{-1}}{1-z^{-1}}\right)^2$	$\frac{1-2d}{2}z^{-1}-\frac{(d-1)^2}{2}z^{-2}$

TABLE I
CT TO DT TRANSFORMATION TABLE

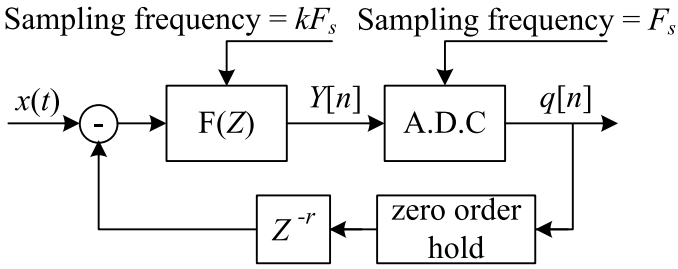

Fig. 3. Oversampled ΣΔ modulator

$g(s)$	$F(Z)$
$\frac{1}{\frac{T_s}{k}s}$	$\frac{Z^{-1}}{1-Z^{-1}}-DZ^{-1}$
$\frac{1}{\frac{T_s}{k}s-a}$	$e^{-ad}\frac{e^a-1}{a}\frac{Z^{-1}}{1-e^a Z^{-1}}-\frac{1-e^{-aD}}{a}Z^{-1}$
$\frac{1}{(\frac{T_s}{k}s)^2}$	$\frac{1-2D}{2}\frac{Z^{-1}}{1-Z^{-1}}+\left(\frac{Z^{-1}}{1-Z^{-1}}\right)^2+\frac{D^2}{2}Z^{-1}$
$\frac{1}{(\frac{T_s}{k}s-a)^2}$	$\frac{e^{-aD}(1+aD-e^a(a(1-D)-1))Z^{-1}}{a^2(1-e^a Z^{-1})}$ $+\frac{(e^a-1)e^{a(1-D)}Z^{-2}}{a^2(1-e^a Z^{-1})^2}+\frac{1-(1+aD)e^{-aD}}{a^2}Z^{-1}$

TABLE II
ODT TRANSFORMATION TABLE

$$\begin{aligned} &Y^*[n] = L^{-1}_{t=nT_s}\left[b(s)g(s)\right] \\ &\text{with} \quad b(s) = \tfrac{1-e^{-T_s s}}{s}e^{-dT_s s} \end{aligned} \tag{7}$$

In the case of the ODT modulator:

$$\begin{aligned} &Y^*[n] = \mathrm{Z}^{-1}_{N=kn}\left[B(Z)F(Z)\right] \\ &\text{with } B(Z) = \tfrac{Z^{-r}(1-Z^{-k})}{k(1-Z^{-1})} \end{aligned} \tag{8}$$

These two responses should be equal for each A.D.C sampling time at frequency F_s. A sufficient condition is that the response to a discrete impulse of $F(Z)$ is the same, for each sampling time at kF_s, as the response of $g(s)$ to an impulse with a width equal to T_s/k delayed by $d-r/k$. Indeed, an impulse with a width of T_s is the sum of k impulses with a width of T_s/k delayed by multiples of T_s/k.

This condition can be expressed as:

$$F(Z) = (1-Z^{-1})\mathrm{Z}_{t=nT_s/k}\left\{L^{-1}\left[\frac{g(s)e^{-(d-r/k)T_s s}}{s}\right]\right\} \tag{9}$$

We denote $D = kd - r$. It can be clearly seen from its definition that $0 \le D < 1$. (9) can be rewritten as:

$$F(Z) = (1-Z^{-1})\mathrm{Z}_{t=nT_s/k}\left\{L^{-1}\left[\frac{g(s)e^{-D\frac{T_s}{k}s}}{s}\right]\right\} \tag{10}$$

If we replace D by d and T_s/k by T_s, this equation is the same as (1). Table II gives solutions of (10), i.e. shows how to obtain $F(z)$ from $g(s)$ for first-order and second-order filters.

The transformation will be denoted as ODTT, Oversampled Discrete-Time Transform, and we will write it as:

$F(Z) = ODTT(g(s), k, D)$ where k is the oversampling ratio and D the difference between the real delay and the modeled delay normalized to T_s/k. We can see that in case of $k = 1$, this transformation is the same as the transformation of (1), i.e. $f(z) = ODTT(g(s), 1, d)$.

III. APPLICATION TO THE SIMULATION OF HIGH-ORDER MODULATORS WITH LOOP DELAY

The methodology proposed in section II will be used to simulate the time-domain response of a CT ΣΔ modulator. In this section we try to obtain the output signal of the modulator. We will evaluate the error committed by this model in terms of Signal Transfer Function (STF).

A. Simulation of the output of a modulator

A CT ΣΔ modulator can always be expressed as Fig. 4, where $g_x(s)$ is the transfer function between the modulator input and the A.D.C input, and $g_q(s)$ is the transfer function between the D.A.C output and the A.D.C input.

With the definition $r = \lfloor kd \rfloor$, and $D = kd - r$, we use the ODTT transformation as:

$$F_x(Z) = ODTT(g_x(s), k, 0) \tag{11}$$

$$F_q(Z) = ODTT(g_q(s), k, D) \tag{12}$$

The ODT topology is shown in Fig. 5

If the input signal was sampled and held at frequency kF_s, then both modulators would be completely equivalent. Let us

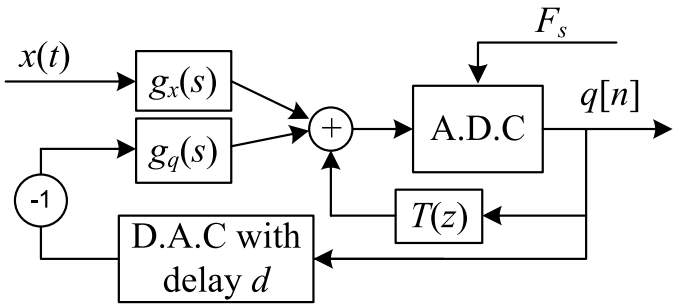

Fig. 4. CT ΣΔ modulator

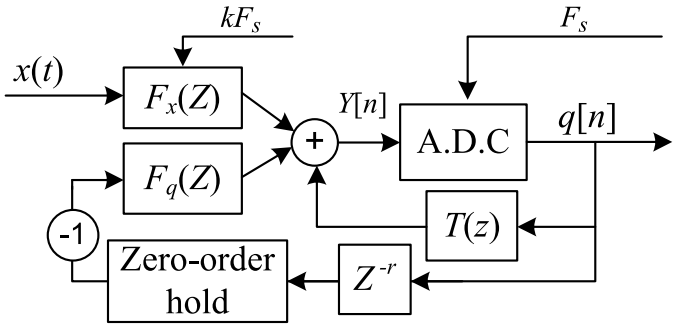

Fig. 5. ODT ΣΔ modulator

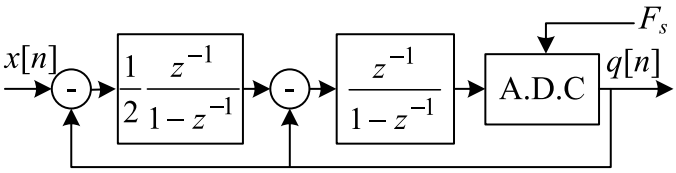

Fig. 6. 2^{nd} order DT ΣΔ modulator

take as an example a classical second-order modulator. The DT prototype is given in Fig. 6.

In this modulator,

$$f(z) = \frac{z^{-1}}{1-z^{-1}} + \frac{1}{2}\left(\frac{z^{-1}}{1-z^{-1}}\right)^2. \quad (13)$$

A CT equivalent topology can be found using table I. With a loop delay d between 0 and 1, we obtain:

$$g(s) = \frac{1}{2(T_s s)^2} + \frac{2d+3}{4T_s s}, \quad (14)$$

$$T(z) = \left(\frac{d(3+d)}{4}\right) z^{-1}. \quad (15)$$

The prototype of the CT modulator is given in Fig. 7. This modulator is equivalent to the Fig. 6 modulator in terms of NTF, but the STF will have to be evaluated separately. It should be noted that the proposed methodology does not give the signal coefficients x_1 and x_2. They will be calculated using STF considerations.

In order to simulate the CT modulator, we transform the CT modulator into an ODT topology. This operation takes into account the feedback signal and the input signal of the modulator. For this example, we choose $d = 0.5$. With this value,

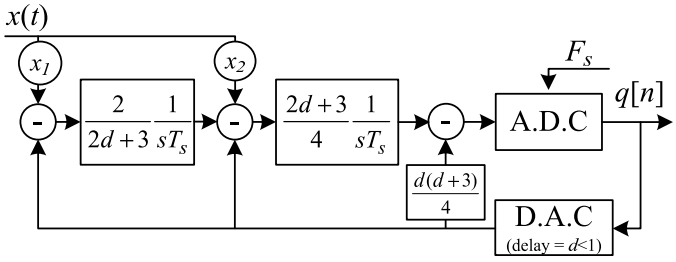

Fig. 7. 2^{nd} order CT ΣΔ modulator

$$g_q(s) = \frac{1}{2(T_s s)^2} + \frac{1}{T_s s}, \quad (16)$$

and

$$g_x(s) = x_1 \frac{1}{2(T_s s)^2} + x_2 \frac{1}{T_s s}. \quad (17)$$

In order to simplify the example, the oversampling ratio k will be chosen as an even number. As a result, $r = k/2$, and $D = 0$. Using (11), and (12), then

$$F_q(Z) = \frac{1+4k}{4k}\left(\frac{1}{k}\frac{Z^{-1}}{1-Z^{-1}}\right) + \frac{1}{2}\left(\frac{1}{k}\frac{Z^{-1}}{1-Z^{-1}}\right)^2, \quad (18)$$

$$F_x(Z) = \frac{x_1+4kx_2}{4k}\left(\frac{1}{k}\frac{Z^{-1}}{1-Z^{-1}}\right) + \frac{x_1}{2}\left(\frac{1}{k}\frac{Z^{-1}}{1-Z^{-1}}\right)^2. \quad (19)$$

The resulting ODT topology is shown in Fig. 8.

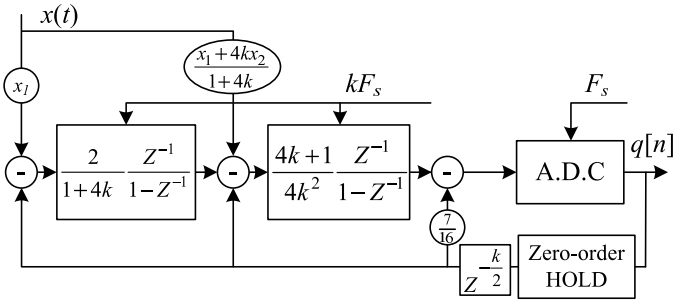

Fig. 8. Equivalent 2^{nd} order ODT ΣΔ modulator

In order to verify the theory, we have simulated the behavior of the second order DT modulator shown in Fig. 6, the equivalent CT modulator shown in Fig. 7, and the ODT model shown in Fig. 8, using Simulink software. The chosen OSR is equal to $k = 8$. Only the ten first output samples were plotted so as to make the figures more readable. In a first simulation, the input signal is a constant signal equal to 0 and the A.D.C inputs are compared in Fig. 9. $DacMax$ is the D.A.C maximum output voltage. It can be seen that these signals are equal for each sample time nT_s. Furthermore, the inputs of the comparators of the CT and the ODT modulator are equal at each sub-sampling time nT_s/k.

We have then applied a low frequency sine input signal sampled and held at frequency kF_s and verified in Fig. 10 that the comparator inputs are still equal for each sub-sampling

time for both the CT model and the ODT model. It can be verified that the DT model does not give the right response.

At last, we have applied a low frequency CT sine input signal and verified in Fig. 11 that the comparator inputs differ increasingly for the CT and ODT case. However, the fact that the STF of the CT modulator is correctly approached by the one of the ODT modulator at low frequencies will be verified in the next section.

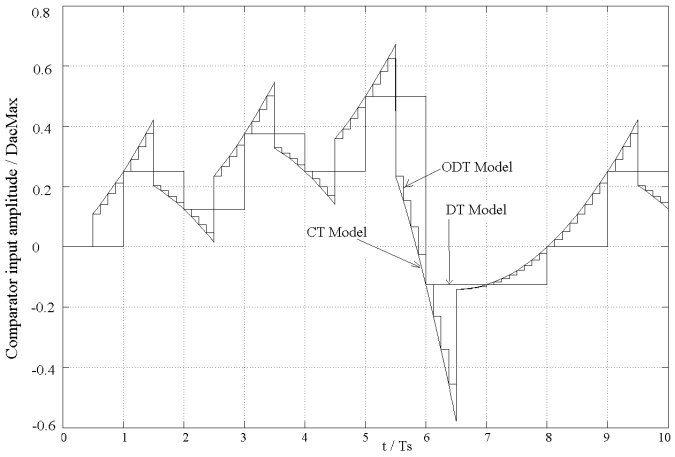

Fig. 9. Simulink simulation of A.D.C input with zero input signal

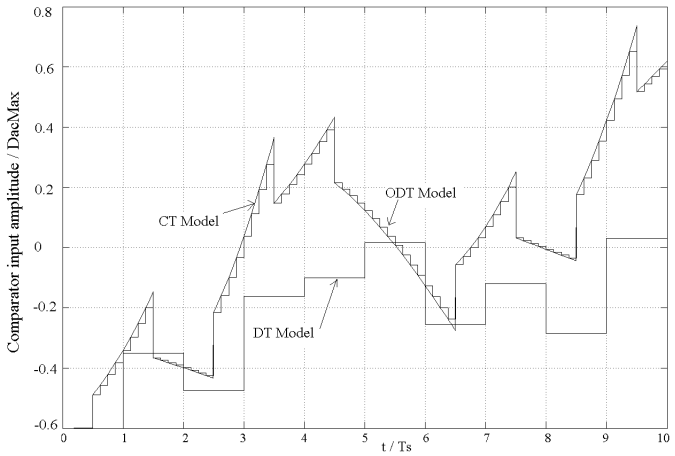

Fig. 10. Simulink simulation of A.D.C input with sampled and held input signal

B. Signal transfer function evaluation

Assuming that the quantizer can be modeled by an additive white noise, the signal transfer function of the CT topology can be expressed as:

$$STF_{CT}(\varphi) = \frac{g_x(2j\pi\varphi)}{1 + f(e^{2j\pi\frac{\varphi}{F_s}})} \tag{20}$$

The STF can be used even for frequencies higher than $F_s/2$, knowing that a signal at frequency φ is aliased into a signal at frequency $\varphi - mF_s$ at the modulator output (where $-\frac{F_s}{2} < \varphi - mF_s < \frac{F_s}{2}$).

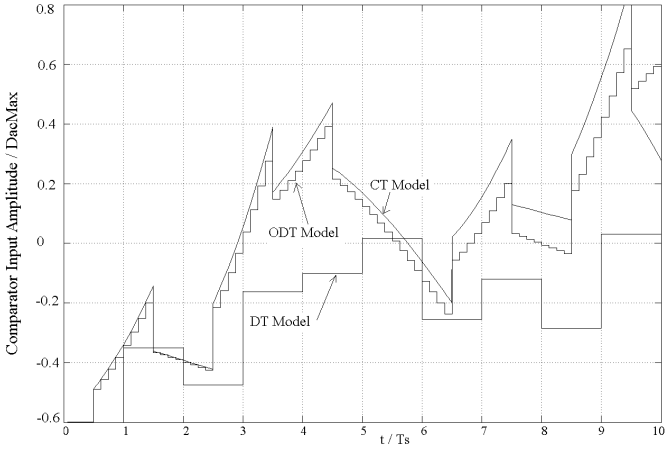

Fig. 11. Simulink simulation of A.D.C input with CT sine input signal

The signal transfer function of the ODT topology can be expressed as:

$$STF_{ODT}(\varphi) = \frac{F_x(e^{2j\pi\frac{\varphi}{kF_s}})}{1 + f(e^{2j\pi\frac{\varphi}{F_s}})}, \tag{21}$$

with $-\frac{kF_s}{2} < \varphi < \frac{kF_s}{2}$.

The error introduced by using the ODT model instead of the CT model can be evaluated using the ratio:

$$err = \frac{STF_{ODT}(\varphi)}{STF_{CT}(\varphi)} = \frac{F_x(e^{2j\pi\frac{\varphi}{kF_s}})}{g_x(2j\pi\varphi)} \tag{22}$$

we denote $\phi = \frac{\varphi}{kF_s}$. When φ varies from $-\frac{kF_s}{2}$ to $\frac{kF_s}{2}$, then ϕ varies from -0.5 to 0.5, and

$$err(\phi) = \frac{F_x(e^{2j\pi\phi})}{g_x(2j\pi\phi kF_s)}. \tag{23}$$

This error depends on the order of the filter, k, and its poles. In the case of integrators, the error does not depend on k.

For a first-order integrator,

$$err(\phi) = \frac{2j\pi\phi}{e^{2j\pi\phi} - 1}, \tag{24}$$

for a second-order integrator,

$$err(\phi) = \frac{(2j\pi\phi)^2 \left(e^{2j\pi\phi} + 1\right)}{2\left(e^{2j\pi\phi} - 1\right)^2}, \tag{25}$$

and for a third-order integrator,

$$err(\phi) = \frac{(2j\pi\phi)^3 \left(\left(e^{2j\pi\phi}\right)^2 + 4e^{2j\pi\phi} + 1\right)}{6\left(e^{2j\pi\phi} - 1\right)^3}. \tag{26}$$

These results were obtained from table II.

A pre-distortion filter may be introduced in order to compensate this STF distortion. This filter will not flatten the error from $-\frac{kF_s}{2}$ to $\frac{kF_s}{2}$, but it will work efficiently from $-\frac{kF_s}{4}$ to $\frac{kF_s}{4}$. The filter must also be a symmetrical FIR filter in order to maintain a linear phase correction (simple delay). Fig. 12 compares the STF modulus before and after correction for a

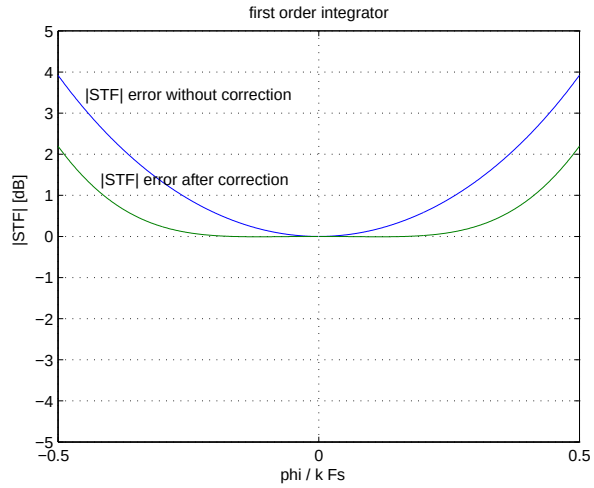

Fig. 12. $\|STF\|$ distortion with and without correction for a first-order integrator

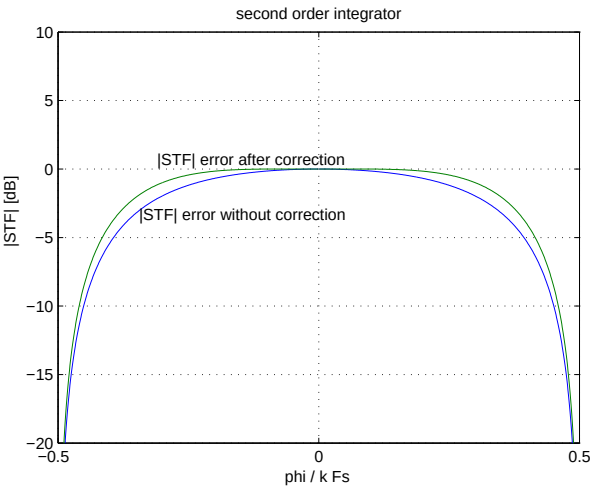

Fig. 13. $\|STF\|$ distortion with and without correction for a second-order integrator

first-order integrator and Fig. 13 for a second-order integrator (φ is denoted as phi on the figures).

The most simple usable FIR (2^{nd} order) can be expressed as, for a n^{th} order integrator by

$$C_n(z) = \lambda_n + (2 - \lambda_n)Z^{-1} + \lambda_n Z^{-2} \qquad (27)$$

with

$\lambda_1 \approx 0.045$, $\lambda_2 \approx -0.045$, $\lambda_3 \approx -0.045$.

It can be verified that the coefficient k has not appeared in the previous expressions.

C. Example of a second-order modulator

As an application, we can consider the modulator shown in Fig. 7. An equivalent bandpass example can be found in [22]. The model shown in Fig. 8 must be changed in order to introduce the STF correction filters, resulting in Fig. 14 topology.

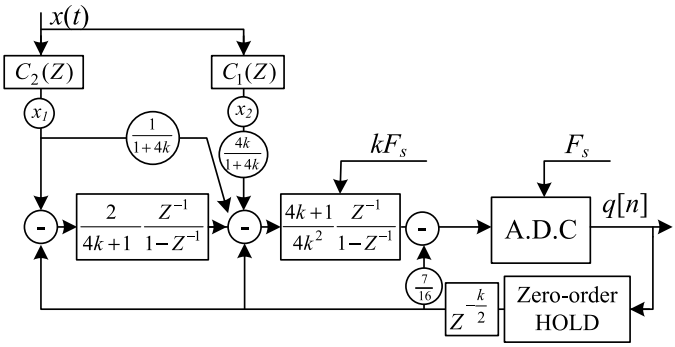

Fig. 14. ODT model of a second-order modulator including signal pre-distortion filter

The STF of this modulator has been evaluated and compared with the one obtained for the ODT model in Fig. 15. The curves are given for the case $x_1 = 1$ and $x_2 = 0$.

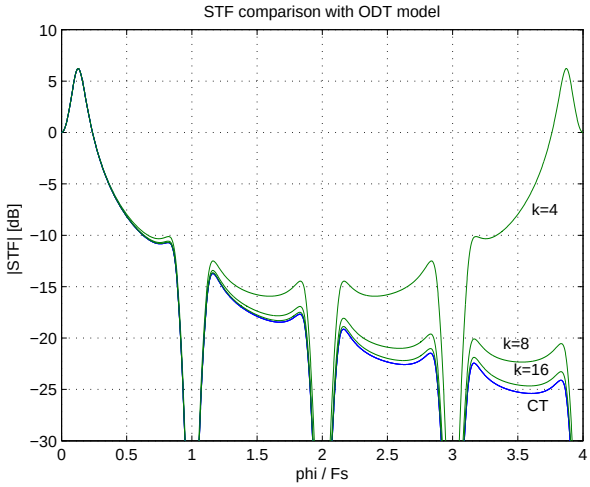

Fig. 15. Comparison of STF obtained with ODT and CT model

In order to evaluate the efficiency of our methodology, this STF could be compared with the one that would be obtained by making a bilinear transform of the filters of the CT modulator. It was shown in [12] that this topology is not strictly equivalent in terms of NTF. In the case of bandpass modulators, there would be a shift in the central frequency of the modulator [22]. In order to obtain a good NTF approximation, an oversampling ratio of at least 64 or 128 would be required. Using the ODT model, the NTF is correct for any value of k.

Table III compares the simulation times of a second-order lowpass modulator for 100000 output samples in the simulink environment. The first model is the one of Fig. 6 with two CT integrators. The algorithm used to solve the equations is ODE45. The DT model is the model of Fig. 7. The third and fourth models (bilinear with k=8 and k=128) are obtained by replacing the CT integrators of Fig. 7 by integrators obtained by a bilinear transform. The last model (ODT) is the one of Fig. 8. The fastest method is the one that uses the DT model ($k = 1$) but it does not provide accurate results in terms of STF. The bilinear method provides good results for $k = 128$ but implies long simulation times. Our methodology provides

good results in terms of NTF and STF with a reasonable simulation time compared to the simulation time required by a continuous algorithm. This kind of results can be verified in [10]. The advantage of our methodology is to be able to simulate the response to signals with large bandwidth by adjusting k.

IV. Intermediary signals simulation

A. A new ODT model taking into account intermediary signals

The goal of the previous section was to simulate the output of the modulator as accurately as possible. This methodology can be extended to the simulation of intermediary signals, especially the outputs of each filter. Indeed, the previous model provides a correct output value of the modulator in the case of a constant or sampled input signal, but the output of intermediary filters does not fit the real output of the CT filters. The methodology proposed in section III is now extended to intermediary signals. If the input signal was a sample and held signal at frequency kF_s, then the methodology would provide the exact intermediary values at every sub-sampling time. For a CT input signal, the model gives an approximation of the outputs of all filters. Note that in this paper, we consider only single-loop topologies, but the methodology could be extended to any kind of topology.

Fig. 16 shows the topology of a n^{th} order CT modulator:

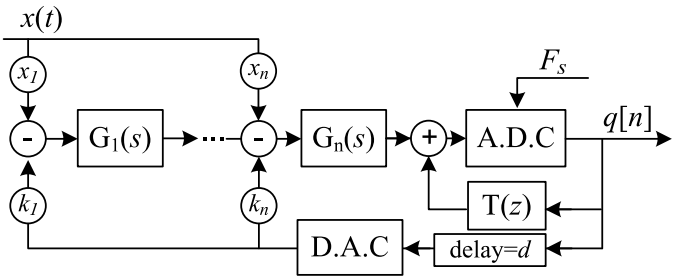

Fig. 16. General CT topology

We denote the transfer functions between D.A.C output and the m^{th} filter output as g_q^m. From Fig. 16, it can be seen that

$$\begin{aligned} &g_q^1(s) = k_1 G_1(s) \\ &g_q^2(s) = G_2(s)(k_2 + k_1 G_1(s)) \\ &... \\ &g_q^n(s) = G_n(s)(k_n + G_{n-1}(s)(k_{n-1} + G_{n-2}(s)(...))) \end{aligned} \tag{28}$$

We also denote the transfer function between the modulator input and the m^{th} filter output as g_x^m. It can also be seen that

$$\begin{aligned} &g_x^1(s) = x_1 G_1(s) \\ &g_x^2(s) = G_2(s)(x_2 + x_1 G_1(s)) \\ &... \\ &g_x^n(s) = G_n(s)(x_n + G_{n-1}(s)(x_{n-1} + G_{n-2}(s)(...))) \end{aligned} \tag{29}$$

Each transfer function can be transformed into its ODT equivalent by

$$\begin{aligned} F_x^m &= ODTT(g_x^m, k, 0), m = 1..n \\ F_q^m &= ODTT(g_q^m, k, D), m = 1..n \end{aligned} \tag{30}$$

The output of F_q^m would be equal to the output of filter G_m if it was connected only to the modulator feedback through k_m. The output of F_x^m would be an approximation of the output of filter G_m if it was connected only to the modulator input through x_m.

The resulting topology is given in Fig. 17. This topology provides the output of each filter separately. It is, of course, functionally correct. However, the quantity of performed calculations is not optimal as each CT filter output requires a separate DT filter.

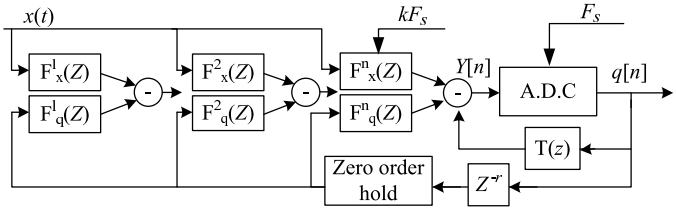

Fig. 17. ODT topology providing intermediary outputs

The topology of Fig. 17 has been transformed in order to be expressed as a classical multi-feedback topology. As the poles are conserved by the ODT transformation, F_x^m and F_q^m have the same poles. Furthermore, these poles are included in the poles of F_x^{m+1} and F_q^{m+1}. As a consequence F_x^m and F_q^m have the same denominator, and the latter divides the denominator of F_x^{m+1} and F_q^{m+1}.

F_x^m and F_q^m can be expressed as

$$\begin{aligned} F_x^m(Z) &= \frac{T_x^m(Z)}{\prod_{p=1}^{m} D_p(Z)} \\ F_q^m(Z) &= \frac{T_q^m(Z)}{\prod_{p=1}^{m} D_p(Z)} \end{aligned} \tag{31}$$

In the case of lowpass modulators, the terms $D_p(Z)$ are first-order filters; in the case of bandpass modulators, they are second-order filters.

The proposed topology used for the simulation is shown in Fig. 18. The integrators are replaced by their denominators and the coefficients are replaced by the numerators of the filters.

The equivalency is ensured if:

$$\begin{aligned} F_x^m(Z) &= \frac{N_x^m(Z) + N_F^m(Z)\left(\frac{\left(N_x^{m-1}(Z) + N_F^{m-1}(Z)(...)\right)}{D_{m-1}(Z)}\right)}{D_m(Z)} \\ F_q^m(Z) &= \frac{N_q^m(Z) + N_F^m(Z)\left(\frac{\left(N_q^{m-1}(Z) + N_F^{m-1}(Z)(...)\right)}{D_{m-1}(Z)}\right)}{D_m(Z)} \end{aligned} \tag{32}$$

These equations will be written recursively:

	continuous	D.T.	Bilinear (k=8)	Bilinear (k=128)	**ODT (k=8)**
simulation time	34s	**1s**	**6s**	130s	**6s**
STF	good	bad	bad	**good**	**good**

TABLE III
COMPARISON OF SIMULATION TIMES AND STF WITH THE THREE METHODOLOGIES

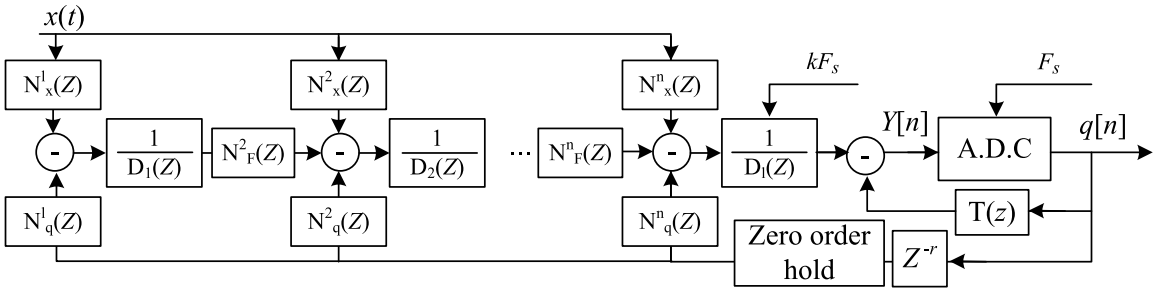

Fig. 18. ODT topology for simulation

$$\begin{aligned}\frac{T_x^m(Z)}{\prod_{p=1}^{m} D_p(Z)} &= \frac{1}{D_m(Z)}\left(N_F^m(Z)\frac{T_x^{m-1}(Z)}{\prod_{p=1}^{m-1} D_p(Z)} + N_x^m(Z)\right)\\ \frac{T_q^m(Z)}{\prod_{p=1}^{m} D_p(Z)} &= \frac{1}{D_m(Z)}\left(N_F^m(Z)\frac{T_q^{m-1}(Z)}{\prod_{p=1}^{m-1} D_p(Z)} + N_q^m(Z)\right)\end{aligned} \tag{33}$$

resulting in:

$$\begin{aligned}T_x^m(Z) &= N_F^m(Z)T_x^{m-1}(Z) + N_x^m(Z)\prod_{p=1}^{m-1} D_p(Z)\\ T_q^m(Z) &= N_F^m(Z)T_q^{m-1}(Z) + N_q^m(Z)\prod_{p=1}^{m-1} D_p(Z)\end{aligned} \tag{34}$$

In case $m = 1$ (first stage), the equations (34) will result in:

$$\begin{aligned}T_x^1(Z) &= N_x^1(Z)\\ T_q^1(Z) &= N_q^1(Z)\end{aligned} \tag{35}$$

Equations (34) are linear equations where the unknown parameters are the coefficients of the polynomials N_F^m, N_x^m, N_q^m. If the delay r is chosen correctly so that $0 < D = kd - r < 1$, then the order of $T_x^m(Z)$ and $T_q^m(Z)$ is the same as the order of $\prod_{p=1}^{m-1} D_p(Z)$, and the constant term is zero (this can be seen from table II). It is also clear from (34) that the constant term of N_F^m, N_x^m and N_q^m is zero. The degree of the resulting solution depends on the degree of the numerators and denominators. It should be noticed that this system always provides a solution. Indeed, increasing the degree of N_F^m, N_x^m and N_q^m by one adds three unknown parameters and only two equations. Consequently, there is a minimal degree for which the number of unknown parameters is at least equal to the number of equations. Our goal is to obtain the minimum order solution of these equations.

If we take into account that the system (34) usually has more equations than unknown terms, then the proposed algorithm can be expressed as follows:

- choose the minimum order for N_F^m, N_x^m and N_q^m,
- solve the system (34) in a mean square sense,
- verify if the solutions provided by the above are valid,
- increase the order of N_F^m, N_x^m and N_q^m by one and start again if the equation system is not solved.

The previous loop will end when the number of unknown parameters is equal to the number of equations. In the case of lowpass modulators, the degree of N_F^m, N_x^m and N_q^m will be $m-1$ and the degree of N_x^0 and N_q^0 is 1. It can be noticed that for orders higher than 2, the terms N_F^m, N_x^m and N_q^m are no longer simple coefficients as it is the case in usual models but when $m > 2$ they are polynomials. This is why a third-order modulator will be considered in the next example. In the case of bandpass modulators the degree of N_F^m, N_x^m and N_q^m is usually 2 for the first three stages. It will be 3 for the last stage.

B. Intermediary signal transfer functions evaluation

As in section III-B, it is possible to define a transfer function between the modulator input and the outputs of the filters. We consider a sine input signal at frequency φ. This signal is aliased into a signal at frequency $\varphi - mF_s$ at the modulator output. Due to the sampling function, the feedback signal contains components at frequencies $\varphi - mF_s + kF_s$. We consider in this evaluation the component for which $k = m$, i.e. a component at frequency φ. The output of filter k is due to 2 terms: one coming from from the input signal x and another coming from the output signal q

For the CT model (Fig. 16), the global transfer function from input to filter k output, defined previously, can be expressed as:

$$TF_{CT}(m,\varphi) = g_x^m(2j\pi\varphi) - g_q^m(2j\pi\varphi)b(2j\pi\varphi)STF_{CT}(\varphi) \tag{36}$$

where g_x^m, g_q^m, and $STF_{CT}(\varphi)$ have been defined respectively in (29), (28), and (20), and b(s) is the transfer function of the D.A.C defined in (7).

On the other hand, this transfer function for the ODT modulator can be expressed as:

$$\begin{aligned} &TF_{ODT}(m,\varphi) = F_x^m(e^{2j\pi\varphi}) \\ &-F_q^m(e^{2j\pi\varphi})B(e^{2j\pi\varphi})STF_{ODT}(\varphi) \end{aligned} \tag{37}$$

where F_x^m, F_q^m, and $STF_{ODT}(\varphi)$ have been defined respectively in (30) and (21), and B(Z) is the transfer function of the D.A.C defined in (8).

C. Application to the simulation of a low pass modulator

We consider a classical third-order lowpass modulator. Its topology is given in Fig. 19 with $g_1 = 0.182$, $g_2 = 0.46$, $g3 = 1.495$, $t_1 = 0.664$. These values can be easily found using, for example, the tools of Schreier [1]. The chosen D.A.C delay is equal to half the sampling period ($dT_s = 0.5$), and k is equal to 8. With these values, $r = 4$ and $D = 0$.

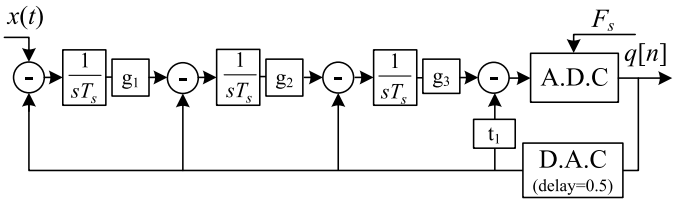

Fig. 19. Classical third-order lowpass modulator

By applying the section IV-A methodology, the equivalent ODT model is given by Fig. 20 with

$$\begin{aligned} &N_x^1(Z) = Z^{-1} \\ &N_q^1(Z) = Z^{-1} \\ &N_F^2(Z) = Z^{-1} \\ &N_x^2(Z) = 1.13e^{-2}Z^{-1} \\ &N_q^2(Z) = 1.011Z^{-1} \\ &N_F^3(Z) = 1.5Z^{-1} - 0.5Z^{-2} \\ &N_x^3(Z) = 2.18e^{-4}Z^{-1} + 3.27e^{-4}Z^{-2} \\ &N_q^3(Z) = 1.029Z^{-1} - 0.028Z^{-2} \end{aligned} \tag{38}$$

If we had used a classical bilinear transformation of the filters, coefficients N_x^2 and N_x^3 would have been zero and coefficients N_q^2, N_q^3 and N_F^3 would have been Z^{-1}. However if we increase the value of k, then N_x^2 and N_x^3 tend towards zero and the other coefficients tend towards Z^{-1}.

Fig. 21 compares the shape of the outputs of each filter obtained by a simulation with Simulink of the CT model and the ODT model. The input signal was a sampled and held signal at kF_s. It can be seen that the ODT model provides the real value of each filter output for each sub-sampling time.

In addition to section III-B, it is possible to evaluate the transfer function between the modulator input and each filter output. The definition was given in section IV-B: a sine input signal at frequency φ is applied at the modulator input and we consider the component at this same frequency at each filter output.

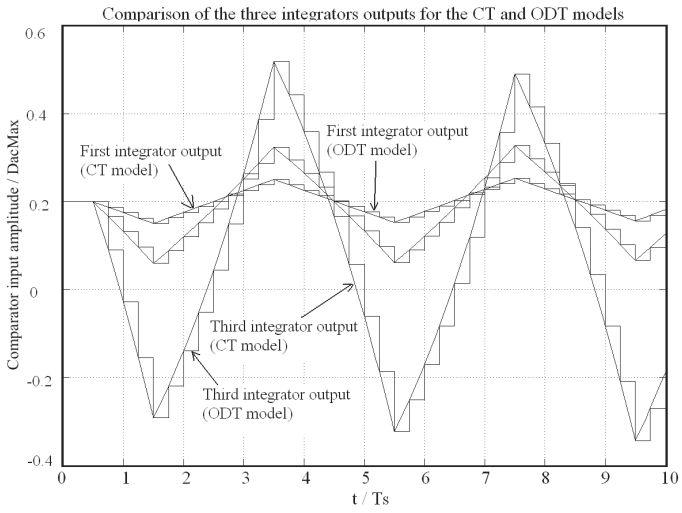

Fig. 21. Intermediary signals shape

As an example, Fig. 22 compares the STF between the modulator input and the first integrator output. It is obtained with the CT model and the ODT model, without STF correction, and for a CT input signal. We could verify that the ODT model gives a good approximation of the output of all filters.

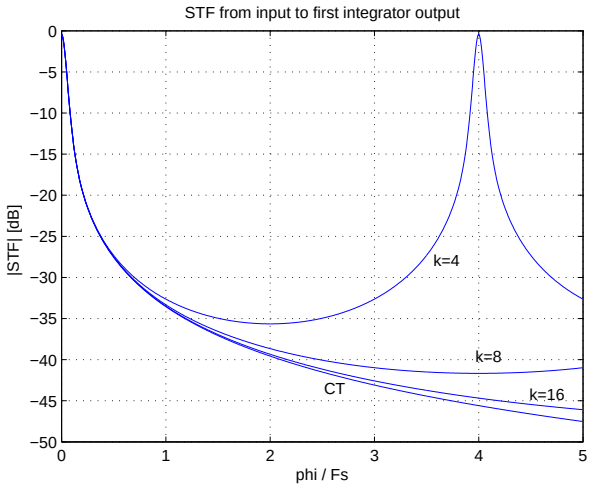

Fig. 22. Transfer function between input and first integrator output

As in [12], it can be seen that taking k equal to 4 times the ratio between the input signal bandwidth and $F_s/2$ provides accurate results for the simulation of intermediary signals.

The simulations times are almost the same as the ones obtained in table III. Simulating 100000 samples of a second-order modulator requires 7 seconds (it was 6 seconds when intermediary signals were not expected).

D. Extension to non-idealities simulations

The ODT model can be extended to the simulation of non ideal behaviors. Some of them can be directly deduced from the ODT model while others need some mathematical developments. To start with, discrete-time noise sources can be easily added at the input or the output of all filters. This can

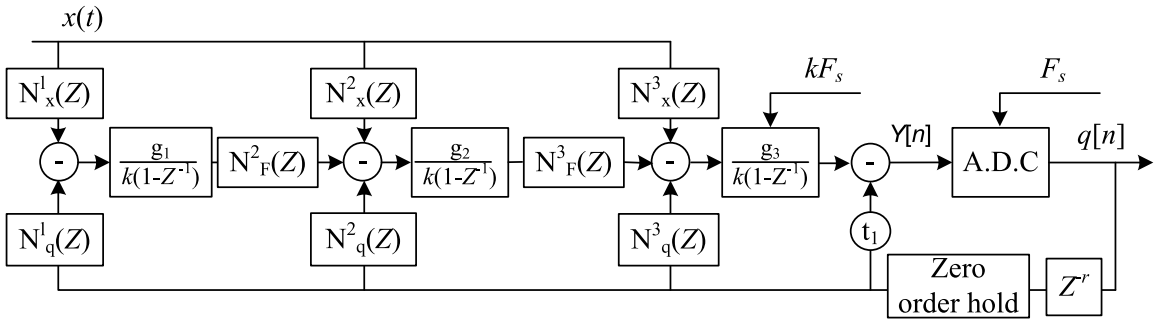

Fig. 20. Equivalent ODT modulator

help evaluate, for example, the effect of the high frequency noise components that are aliased in the signal band. The spectrum shape of the noise sources has to be derived from the CT noise models. The effect of the clock jitter [25] could also be included in this model. However, this would require a mathematical extension of the methodologies proposed, for example, in [26]. It is also possible to model the real transfer function of the amplifiers used in the modulator, as long as they can be expressed by their Laplace transform. This can be easily done by replacing the ideal filters G_1 to G_n in Fig. 16 by their real transfer functions. Some non linearities can also be introduced, by including, for example, a non-linear transfer function at the output of the amplifiers. Nevertheless, such a model cannot simulate correctly, for example, non linear capacitances in an integrator. In any case, high level models can never replace a transistor-level simulation where all non-linearities and parasitical elements are considered.

V. Conclusion

A methodology for time-domain simulations of continuous-time modulators was proposed. This methodology is based on a fixed step discretization of each output sample. Compared with variable step methods, the fixed step method ensures an exact equivalency in terms of noise transfer function, even for large simulation steps. The optimal step value is chosen from the bandwidth of the input signal. This allows for the simulation of the response to input signals with a bandwidth larger than half the modulator sampling frequency. This methodology can also help simulate the behavior of all the modulator outputs of the filters on top of the modulator output signal. Simulations are very fast as they use discretized equations. STF considerations have shown that the ODT method describes the behavior of a CT modulator better than a classical transformation method such as the bilinear transform.

Appendix A
Maple code to solve (1)

The following code was used to solve (1) and to fill in table I. In this code, which deals with first-order terms ($\frac{1}{s-a}$), the delay d is between 0 and 1. Larger delays can be considered by changing the sixth line of the code.

```
with(inttrans);readlib(invztrans);
g := 1/(((s*Ts-a))) ;
assume(d>0,Ts>0);
b := exp(-d*Ts*s);
t1 := invlaplace(g/s*b,s,t);
t2 := subs(Heaviside(t-d*Ts)
      =Heaviside(t-1*Ts),t1);
t3 := simplify(ztrans(subs(t=n*Ts,t2),n,z)
      *(1-z^(-1)));
f := convert(t3,parfrac,z);
```

Appendix B
Matlab code to solve (34)

The following code can be used to solve (34). This code was tested only in the case when the size of s1 was the same as the size of s2.

```
%-------------------------------------------
%   resolution of the equation   s1 = ax + by
%                                s2 = cx + dz
%   with s1,s2,a,b polynomial inputs
%
%-------------------------------------------

function [x,y,z] = polysolve2(s1,a,b,s2,c,d)

or=0 ;
while ((s1(1)==0) && (s2(1)==0))
    s1=s1(2:end) ;  s2=s2(2:end) ;
    or=or+1 ;
end

ls1=length(s1); ls2=length(s2);
lngsolx=ls1-length(a)+1;
lngsoly=ls1-length(b)+1;
lngsolz=ls2-length(d)+1;

if (ls1<lngsolx+lngsoly)
  s1=[s1 zeros(1, lngsolx+lngsoly-ls1)] ;
end
if (ls2<lngsolx+lngsolz)
  s2=[s2 zeros(1, lngsolx+lngsolz-ls2)] ;
end
ls1=length(s1); ls2=length(s2);

for dl=0:ls1+ls2-(lngsolx+lngsoly+lngsolz)

    mat=zeros(lngsolx+lngsoly+lngsolz,ls1+ls2) ;

    for k=1:lngsolx
      mat(k,k:k+length(a)-1)=a  ; end
    for k=1:lngsoly
      mat(k+lngsolx,k:k+length(b)-1)=b  ; end
    for k=1:lngsolx
      mat(k,k+ls1:k+ls1+length(c)-1)=c  ; end
    for k=1:lngsolz
      mat(k+lngsolx+lngsoly,k+ls1:k+ls1+length(d)-1)=d  ; end

    x0=[s1 s2]/mat  ;
    err=max(abs((x0*mat-[s1 s2]))) ;
    if (err<1e-8)  break; end
    lngsolx=lngsolx+1 ; lngsoly=lngsoly+1 ; lngsolz=lngsolz+1 ;
    s1=[s1 0] ;   s2=[s2 0] ; ls1=ls1+1 ; ls2=ls2+1 ;
end

x=[zeros(1,or) x0(1:lngsolx)];
```

```
y=[zeros(1,or) x0(lngsolx+1:lngsolx+lngsoly)];
z=[zeros(1,or) x0(lngsolx+lngsoly+1:end)];
```

REFERENCES

[1] R. Schreier and G.C. Temes, *Understanding Delta-Sigma Data Converters.* New Jersey: Wiley, 2005, chap. 4.

[2] P. Benabes, M. Keramat, and R. Kielbasa, "A methodology for designing continuous-time sigma-delta modulators," in *Proc. of European design and test conference*, March 1997, pp. 46-50.

[3] M. Ortmanns and F. Gerfers, *Continuous-Time Sigma-Delta A/D Conversion.* Berlin: Springer, 2006.

[4] A. Naderi, M. Sawan and Y. Savaria, "On the Design of Undersampling Continuous-Time Band-Pass Delta-Sigma Modulators for Gigahertz Frequency A/D Conversion," *To be published in IEEE Trans. Circuits and Systems I: Regular Papers*.

[5] M. Keller, A. Buhmann, F. Gerfers, M. Ortmanns and Y. Manoli, "On the Implicit Anti-Aliasing Feature of Continuous-Time Cascaded SigmaDelta Modulators," *IEEE Trans. Circuits and Systems I: Regular Papers*, Vol. 54, pp. 2639-2645, Dec. 2007.

[6] V.M. Yantchev, I. Katardjiev, "Propagation characteristics of the fundamental symmetric Lamb wave in thin aluminum nitride membranes with infinite gratings," *Journal of applied physics* , Vol. 98 N8, pp. 084910.1-084910.7, 2005.

[7] K. Francken and G.E. Gielen, "A High-Level Simulation and Synthesis Environment for $\Sigma\Delta$ Modulators," *IEEE Trans. Computer-aided Design of Integrated Circuits and Systems*, Vol. 22, pp. 1049-1061, Aug. 2003.

[8] G. Suarez, M. Jimenez and F.O. Fernandez, "Behavioral Modeling Methods for Switched-Capacitor $\Sigma\Delta$ Modulators," *IEEE Trans. Circuits and Systems I: Regular Papers*, Vol. 54, pp. 1236-1244, June 2007.

[9] Y. Guo and L. Peng, "Efficient Look-Up-Table-Based Modeling for Robust Design of $\Sigma\Delta$ ADCs," *IEEE Trans. Circuits and Systems I: Regular Papers*, Vol. 54, pp. 1513-1528, July 2007.

[10] M. Vogels, K. Francken, E. Martens and G. Gielen, "Efficient Time-Domain Simulation of Continuous-Time Sigma Delta A/D Converters Using Analytical Integration," in *Proc. IEEE Int. Symp. on Circuits And Systems (ISCAS)*, May 2002, Vol. 4, pp. 237-240.

[11] M. Keller, A. Buhmann, M. Ortmanns and Y. Manoli, "A method for the Discrete-Time simulation of Continuous-time Sigma-Delta Modulators," in *Proc. IEEE Int. Symp. on Circuits And Systems (ISCAS)*, May 2007, Vol. 1, pp. 241-244.

[12] R. Sobot, S. Stapleton and M. Syrzycki, "Tunable Continuous-Time Bandpass $\Sigma\Delta$ Modulators With Fractional Delays," *IEEE Trans. Circuits and Systems II: Express Briefs*, Vol. 53, pp. 264-273, Feb. 2006.

[13] P. Benabes and A. Beydoun, "Fixed-Step Simulation of Continuous-Time Sigma Delta Modulators," in *Proc. IEEE Int. Symp. on Circuits And Systems (ISCAS)*, May 2008, pp. 1886-1889.

[14] W. Gao, O. Shoaei and W.M. Snelgrove, "Excess loop delay effects in continuous-time delta-sigma modulators and the compensation solution," in *Proc. IEEE Int. Symp. on Circuits And Systems (ISCAS)*, May 1997, Vol. 1, pp. 65-68.

[15] O. Oliaei, "Design of continuous-time sigma-delta modulators with arbitrary feedback waveform," *IEEE Trans. Circuits and Systems II: Analog and Digital Signal Processing*, Vol. 50, pp. 437-444, Aug. 2003.

[16] H. Aboushady and M-M. Lourat, "Systematic Approach for Discrete-Time to Continuous-Time Transformation of sigma-delta Modulators," in *Proc. IEEE Int. Symp. on Circuits And Systems (ISCAS)*, May 2002, Vol. 4, pp. 229-232.

[17] S. Loeda, H.M. Reekie, and B. Mulgrew, "On the Design of High-Performance Wide-Band Continuous-Time Sigma-Delta Converters Using Numerical Optimization," *IEEE Trans. Circuits and Systems II: Express Briefs*, Vol. 53, pp. 802-810, Apr. 2006.

[18] A. Yahia, P. Benabes, and R. Kielbasa, "A new technique to compensate the influence of the feedback DAC delay in continuous-time bandpass converters," *Proceedings of IEEE Instrumentation and Measurement Technology Conference*, May 2001, pp. 716-719.

[19] M. Keller, A. Buhmann, J. Sauerbrey, M. Ortmanns, Y. Manoli, "A Comparative Study on Excess Loop Delay Compensation Techniques for Continuous-Time Sigma-Delta Modulators," *To be published in IEEE Trans. Circuits and Systems I: Regular Papers*.

[20] A. Yahia, P. Benabes, and R. Kielbasa, "Bandpass sigma-delta modulators synthesis with high loop delay," in *Proc. IEEE Int. Symp. on Circuits And Systems (ISCAS)*, May 2001, Vol. 1, pp. 344-347.

[21] P. Benabes, M. Keramat, and R. Kielbasa, "Synthesis and analysis of sigma-delta modulators employing continuous-time filters," *Analog Integrated Circuits and Signal Processing*, vol. 23, pp. 141-152, March 2000.

[22] P. Benabes, P. Aldebert, A. Yahia, and R. Kielbasa, "Influence of the feedback D.A.C delay on a continuous-time bandpass sigma-delta converter," *Electronics Letters*, vol. 36, pp. 292-294, April 2000.

[23] P. Benabes, "Fast simulation of bandpass Continuous-Time Sigma delta Modulators," in *Proc. Joint 6th Int. IEEE Northeast Workshop on Circ. and Syst. NEWCAS-TAISA 2008*, June 2008, pp. 289-292.

[24] B.C. Kuo (1980). *Digital control systems.* 2nd Edition, OXFORD UNIV PR. 1992 ISBN 9780195120646.

[25] O. Shoaei, "Continuous-Time Delta-Sigma A/D Converters for High Speed Applications," Ph.D. dissertation, Carleton University, Nov. 1995.

[26] K. Reddy and S. Pavan, "Fundamental Limitations of Continuous-Time DeltaSigma Modulators Due to Clock Jitter," *IEEE Trans. Circuits and Systems I: Regular Papers*, Vol. 54, pp. 2184-2194, Oct. 2007.

[27] M. Ortmanns, F. Gerfers and Y. Manoli, "A continuous-time Sigma Delta Modulator with reduced sensitivity to clock jitter through SCR feedback," *IEEE Trans. Circuits and Systems I: Regular Papers*, Vol. 52, pp. 875-884, May 2005.

Philippe Benabes was born in NICE in 1967. He received the degree of "Diplôme d'Ingénieur de l'Ecole Centrale Paris" in 1989. From 1989 to 1991, he worked for Thomson Sintra ASM as a board designer. He received a Ph.D. degree in 1994 for the work "Wideband Bandpass $\Sigma\Delta$ Converters".
He is currently professor of electronics at Supélec and holds an "Habilitation à diriger les recherches". His research interests include both analog and digital electronics and more particulary the design of bandpass sigma-delta analog-to-digital converters.