ELSEVIER

Signal Processing 60 (1997) 163–179

SIGNAL PROCESSING

Joint recursive implementation of time–frequency representations and their modified version by the reassignment method

Cédric Richard*, Régis Lengellé

Laboratoire LM2S, Université de Technologie de Troyes, 12 rue Marie Curie, BP 2060, F 10010 Troyes Cedex, France

Received 13 June 1996; revised 23 December 1996 and 17 March 1997

Abstract

Cohen's class time–frequency distributions (CTFDs) have significant potential for the analysis of non-stationary signals, even if the poor readability of their representations makes visual interpretations difficult. To concentrate signal components, Auger and Flandrin recently generalized the reassignment method (first applied to the spectrogram in the 1970s) to any bilinear representations. Unfortunately, this process is computationally expensive. In order to reduce computation time and to improve representations readability, we first introduce a new fast algorithm which allows the recursive evaluation of classical spectrograms and spectrograms modified by the reassignment method. In a second step, we show that rectangular, half-sine, Hamming, Hanning and Blackman functions can be used as running 'short-time' windows. Then the previous algorithm is extended to CTFDs. We show that the windows mentioned above can also be used to compute recursively reassigned smoothed pseudo-Wigner–Ville distributions. Finally, we show that the constraints on candidate windows are not very restrictive: any function (assumed periodic) can be used in practice as long as it admits a 'short enough' Fourier series decomposition. © 1997 Elsevier Science B.V.

Zusammenfassung

Cohen's Klasse der Zeit-Frequenz-Verteilungen spielt eine entscheidende Rolle in der Analyse nicht-stationärer Signale, auch wenn die schlechte Lesbarkeit ihrer Darstellungen die visuelle Interpretation erschwert. Zur Konzentration von Signalkomponenten entwickelten Auger und Flandrin kürzlich die Rückzuteilungsmethode *reassignment method* (erstmals angewendet auf Spektrogramme in den Siebzigern) für einige bilineare Darstellungen. Unglücklicherweise ist dieser Prozeß numerisch aufwendig. Um Rechenzeit zu verringern und die Lesbarkeit der Darstellungen zu verbessern, schlagen wir einen neuen schnellen Algorithmus vor, welcher die rekursive Evaluierung klassischer Spektrogramme und von Spektrogrammen, die durch die *reassignment method* modifiziert wurden, erlaubt. In einem zweiten Schritt zeigen wir, daß Rechteck-, Halbsinus-, Hamming-, Hanning- und Blackmanfenster als gleitende 'Kurzzeit-Fenster' eingesetzt werden können. Daran anschließend wird der zuvor beschriebene Algorithmus zur CTDF erweitert. Wir zeigen weiterhin, daß die oben genannten Fenster zur Berechnung rekursiver *reassigned*, geglätteter Pseudo-Wigner–Ville Verteilungen benutzt werden können und daß letztendlich die Anforderungen an geeignete Fensterfunktionen nicht sehr hoch sind: jede beliebige Funktion (als periodisch vorausgesetzt) kann in der Praxis benutzt werden, sofern sie eine 'hinreichend kurze' Fourier-Reihenzerlegung gestattet. © 1997 Elsevier Science B.V.

*Corresponding author. Tel.: + 33.3.25.71.56.77; fax: + 33.3.25.71.56.99; e-mail: cedric.richard@univ-troyes.fr.

0165-1684/97/$17.00 © 1997 Elsevier Science B.V. All rights reserved

Résumé

Les distributions de la classe de Cohen jouent un rôle primordial dans l'analyse temps–fréquence des signaux non-stationnaires, même si la présence de termes d'interférences nuit à la lisibilité des représentations obtenues. Afin d'améliorer cette situation, Auger et Flandrin ont récemment élargi l'emploi de la technique de réallocation, d'abord proposée pour le spectrogramme, à l'ensemble des représentations bilinéaires. Néanmoins, l'utilisation de cette méthode reste délicate en raison de la lourdeur des calculs mis en oeuvre. Afin de réduire le temps de calcul et d'améliorer conjointement la lisibilité des représentations, nous proposons dans un premier temps un algorithme rapide, basé sur une approche récursive, permettant l'évaluation simultanée de spectrogrammes classiques et réalloués. Nous montrons que les fenêtres d'analyse rectangulaire, sinusoïdale, de Hamming, Hanning et Blackman vérifient les contraintes imposées par l'implémentation récursive. Cette approche est ensuite étendue à l'évaluation récursive des distributions de pseudo-Wigner–Ville lissées classiques et réallouées. Nous vérifions alors que les fonctions enumérées ci-dessus peuvent jouer le rôle de fenêtres de lissage temporel. Finalement, l'accent est porté sur le fait que les contraintes imposées aux fenêtres sont peu restrictives: toute fonction peut être utilisée si toutefois elle admet une décomposition en série de Fourier de faible dimension. © 1997 Elsevier Science B.V.

Keywords: Cohen's class distributions; Reassignment method; Fast algorithms; Recursive approach

1. Introduction

Cohen's class time–frequency distributions (CTFD), which includes the spectrogram and the Wigner–Ville distribution, have been widely used to analyze non-stationary signals [12]. Two major problems have been extensively addressed: the difficulty of adjusting readability and the large computational cost of these methods.

Adjusting readability requires both a good concentration of the signal components on the time–frequency map and the absence of misleading interferences. To make visual interpretation easy, one can choose an appropriate CTFD and empirically adjust its parameters, giving an advantage either to signal concentration or interferences elimination. However, several approaches allow to automatically improve the readability of time-frequency representations. To remove cross-components, Flandrin introduced various weighting functions in computing the Wigner–Ville distribution (WVD) [11]. Sun et al. [18] and Grall-Maës and Beauseroy [13] used appropriate image processing techniques. Jones and Baraniuk [5, 14] and Jones and Parks [15] proposed efficient adaptive methods, which are computationally expensive when applied to long signals. Finally, the reassignment method, first applied 18 years ago to the spectrogram by Kodera et al. can produce a good localization of the signal components [16]. This method increases readability by relocating the representation values away from their location, thus creating a *reassigned CTFD*. Recently, Auger and Flandrin generalized the reassignment process, applying it to any bilinear time–frequency and time-scale representations, and simplified its implementation by proposing a new formulation [4].

In order to efficiently compute long signals time–frequency distributions, several ideas to reduce computation time have been recently proposed. Martin and Flandrin [17], Boashash and Black [8] used symmetry properties of distributions and Barry clever matrix manipulations [6]. For the evaluation of the Wigner–Ville distribution, Cunningham and Williams proposed a decimation algorithm which shifts the signal so that the resulting twiddle multiplication number is reduced [9]. In [10], the same authors also defined approximations to real-valued CTFDs using spectrograms that admit fast evaluation. Finally, Amin introduced recursivity in the implementation of time–frequency distributions: in [1], he proposed a class of kernels which allow the representation to be updated with a number of computations independent of the window extent. A family of kernels which permit the local autocorrelation function to be evaluated recursively is also considered in [2]. Among all these approaches, the use of recursive methods seems to be most efficient for reducing

computation time when applied to long signals (biomedical signal analysis, etc.).

In this paper, we propose a new fast algorithm which allows the evaluation of classical CTFDs and CTFDs modified by the reassignment process. This algorithm uses a recursive approach and includes both the CTFD computation and the reassignment stage. The paper is organized as follows. In Section 2, we present a recursive algorithm for the spectrogram computation. Then, we discuss the associated running windows which can be used and we introduce a new on-line recursive algorithm dedicated to reassigned spectrograms. In Section 3, these results are extended to specific discrete reassigned Cohen's class distributions. Our approach is discussed in Section 4.

2. Recursivity for the reassigned spectrogram

2.1. Spectrogram recursive implementation

(i) *Problem formulation.* The spectrogram appeared in the 1940s under the sonagram form [12] and is still extensively used although its time and frequency resolutions are bounded. For a discrete complex signal $x(k)$, this representation can be defined as follows:

$$S_x\left(n+\frac{N}{2},\omega\right)=|F_x^w(n,\omega)|^2, \tag{1}$$

where

$$F_x^w(n,\omega)=\sum_{k=1}^{N} x(n+k)w(k)\mathrm{e}^{-\mathrm{j}\omega k}. \tag{2}$$

The above definition is chosen in order to simplify the recursive formulation introduced in the next sections. The variable ω denotes the angular frequency ($\omega=2\pi f$), and w is an N-sample 'short-time' analysis window which plays a central role in adjusting time and frequency resolutions.

Evaluating the spectrogram by the direct application of Eq. (2) requires the calculation of an N-sample FFT at each time instant n, which can be computationally expensive. This justifies the search for a recursive $F_x^w(n,\omega)$ expression such as

$$F_x^w(n+1,\omega)=C\,F_x^w(n,\omega)\mathrm{e}^{\mathrm{j}\omega}+D_x(n,\omega), \tag{3}$$

where C is a complex constant and D_x an additive correction term depending on the signal x, the angular frequency ω and the current time instant. Our problem is now to establish a sufficient condition on the running window w allowing the above recursive formulation.

(ii) *Presentation of the recursion.* Introducing Eq. (2) into Eq. (3) and after a straightforward manipulation, we obtain

$$\sum_{k=1}^{N} x(n+k+1)[w(k)-Cw(k+1)]\mathrm{e}^{-\mathrm{j}\omega k}$$
$$=D_x(n,\omega)+C[x(n+1)w(1)$$
$$-x(n+N+1)w(N+1)\mathrm{e}^{-\mathrm{j}\omega N}].$$

This equation must be verified by any signal $x(k)$. Then, an obvious sufficient condition is

$$w(k-1)=Cw(k)$$

$$D_x(n,\omega)=C[x(n+N+1)w(N+1)\mathrm{e}^{-\mathrm{j}\omega N}$$
$$-x(n+1)w(1)].$$

Consequently, the recursive formulation of the spectrogram is given by

$$F_x^w(n+1,\omega)=CF_x^w(n,\omega)\mathrm{e}^{\mathrm{j}\omega}-Cx(n+1)w(1)$$
$$+x(n+N+1)w(N)\mathrm{e}^{-\mathrm{j}\omega N}. \tag{4}$$

Let us analyze now the functional equation $w(k-1)=Cw(k)$ in order to define the corresponding family of candidate windows.

$$w(k-1)=Cw(k)\ \Leftrightarrow\ w(k)=C^{-k}w(0). \tag{5}$$

Therefore, using the complex notation $C=r\mathrm{e}^{\mathrm{j}\theta}$, the solution family W is defined as

$$W=\{w\,|\,w(k)=\alpha r^{-k}\mathrm{e}^{-\mathrm{j}\theta k},\ \alpha\ \text{complex}\}. \tag{6}$$

We can notice that W is composed of exponential functions.

(iii) *Generalization of the recursive scheme.* Our purpose is now to extend the family of candidate windows which allow a recursive computation of the spectrogram. If the window w can be expanded

into exponential functions $r_m^{-k}e^{-j\theta_m k}$, i.e., if w belongs to W_{ext} defined as follows:

$$W_{ext} = \left\{ w \,|\, w(k) = \sum_{m=1}^{M} \alpha_m r_m^{-k} e^{-j\theta_m k},\ \alpha_m \text{ complex} \right\}, \tag{7}$$

we can show that the recursive strategy of the spectrogram can be generalized according to

$$F_x^w(n+1,\omega) = \sum_{m=1}^{M} F_x^{w_m}(n+1,\omega), \tag{8}$$

where $w_m(k) = \alpha_m r_m^{-k} e^{-j\theta_m k}$, $C_m = r_m e^{j\theta_m}$

and

$$F_x^{w_m}(n+1,\omega) = C_m F_x^{w_m}(n,\omega)e^{j\omega} - C_m x(n+1)w_m(1) + x(n+N+1)w_m(N)e^{-j\omega N}. \tag{9}$$

Therefore, any window can be used in practice as long as it admits a 'short-enough' Fourier series decomposition.

The following truncating windows, well-known in spectral analysis for their properties, can be advantageously used since they allow a recursive spectrogram computation: rectangular ($M = 1$), half-sine ($M = 2$), Hanning ($M = 3$), Hamming ($M = 3$) and Blackman ($M = 5$). In Table 1, we introduce the way to implement these windows.

(iv) *Evaluation of the proposed algorithm.* The gain in computational time will be directly related to the complexity of the recursive algorithm, which must be compared to the direct implementation of Eq. (2). Thus, we suggest to define the relative efficiency (RE) of the proposed algorithm as follows:

RE = (TNRM using the direct application of Eq. (2)) *divided by* (TNRM using our algorithm),

where TNRM is the Total Number of Real Multiplications required to compute the representation at time n.

As an example, we estimate now the relative efficiency of the recursive algorithm when w is a rectangular window. The evaluation of $F_x^w(n,\omega)$ at the N frequency samples $\{2\pi k/N\}_{k=0,\dots,N-1}$ results from an N-sample discrete Fourier transform. Consequently, the direct application of Eq. (2) requires $(N/2)\log_2(N/2)$ complex multiplications and $N\log_2 N$ complex additions for each time step [7]. Evaluating the squared modulus (see Eq. (1)) requires $2N$ real multiplications and N additions.

Table 1
Usual truncating windows recursive implementation

Window	Expression	M	w_m	C_m
Rectangular	$w(k) = \begin{cases} 1 & \text{for } 1 \leqslant k \leqslant N, \\ 0 & \text{elsewhere.} \end{cases}$	1	$w_1(k) = 1$	$C_1 = 1$
Half-sine	$w(k) = \begin{cases} \sin(\pi k/N + 1) & \text{for } 1 \leqslant k \leqslant N, \\ 0 & \text{elsewhere.} \end{cases}$	2	$w_1(k) = (1/2j)e^{j\pi k/(N+1)}$ $w_2(k) = (-1/2j)e^{-j\pi k/(N+1)}$	$C_1 = e^{-j\pi/(N+1)}$ $C_2 = e^{j\pi/(N+1)}$
Hanning($\alpha = 0.5$) Hamming ($\alpha = 0.54$)	$w(k) = \begin{cases} \alpha + (\alpha - 1)\cos\left(\dfrac{2\pi k}{N+1}\right) & \text{for } 1 \leqslant k \leqslant N \\ 0 & \text{elsewhere.} \end{cases}$	3	$w_1(k) = \alpha$ $w_2(k) = \frac{1}{2}(\alpha - 1)e^{j2\pi k/(N+1)}$ $w_3(k) = \frac{1}{2}(\alpha - 1)e^{-j2\pi k/(N+1)}$	$C_1 = 1$ $C_2 = e^{-j2\pi/(N+1)}$ $C_3 = e^{j2\pi/(N+1)}$
Blackman	$w(k) = \begin{cases} 0.42 - 0.5\cos\left(\dfrac{2\pi k}{N+1}\right) + 0.08\cos\left(\dfrac{4\pi k}{N+1}\right) \\ \quad \text{for } 1 \leqslant k \leqslant N \\ 0 \quad \text{elsewhere.} \end{cases}$	5	$w_1(k) = 0.42$ $w_2(k) = (-0.5/2)e^{j2\pi k/(N+1)}$ $w_3(k) = (-0.5/2)e^{-j2\pi k/(N+1)}$ $w_4(k) = (0.08/2)e^{j4\pi k/(N+1)}$ $w_5(k) = (0.08/2)e^{-j4\pi k/(N+1)}$	$C_1 = 1$ $C_2 = e^{-j2\pi/(N+1)}$ $C_3 = e^{j2\pi/(N+1)}$ $C_4 = e^{-j4\pi/(N+1)}$ $C_5 = e^{j4\pi/(N+1)}$

Table 2
Efficiency of the recursive spectrogram algorithm, in real operations numbers

	Real additions number (A_R) (per time-step)		Real multiplications number (M_R) (per time-step)		Condition on N for
	Direct method	Recursive method	Direct method	Recursive method	RE $\geqslant 1$
Rectangular	$3N\log_2 N$	$5N+2$	$2N\log_2 N$	$6N$	8
Half-sine	$3N\log_2 N+2N$	$11N+12$	$2N\log_2 N+4N$	$10N+16$	13
Hamming Hanning	$3N\log_2 N+2N$	$17N+16$	$2N\log_2 N+4N$	$14N+20$	39

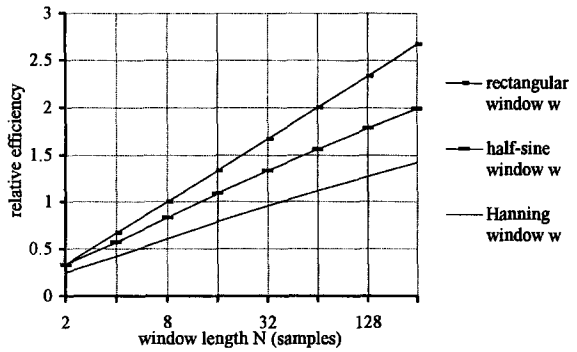

Fig. 1. Relative efficiency (RE) of the spectrogram recursive algorithm.

Then the total numbers of real additions (A_R) and multiplications (M_R) are

$$A_R = 3N\log_2(N), \qquad M_R = 2N\log_2(N).$$

Using the recursion given by Eqs. (3) and (4), the same computation requires N complex multiplications to evaluate the product of $F_x^w(n, 2\pi k/N)_{k=1,\dots,N}$ by $e^{j\omega}$. $(N+1)$ complex additions are needed to compte $D_x(n,\omega)$ and add it to the previous result. Then, the number of real operations can be decomposed as

$$A_R = 5N+2, \qquad M_R = 6N.$$

In Table 2, we compare the total numbers of real operations required to compute the spectrogram when using the recursion or the Eq. (2) direct implementation. Fig. 1 represents the recursive algorithm relative efficiency (RE) for several usual windows as a function of their respective length. These results clearly show that our recursive approach allows to reduce computation time.

2.2. Recursion in the spectrogram reassignment stage

(i) *Problem formation.* To improve the spectrogram readability by concentrating the signal components on the time–frequency map, Kodera et al. proposed to relocate the representation values away from their computation location [16]. The following spectrogram definition, which uses the Rihaczek distribution, is the starting point of their idea:

$$S_x(n,\omega) = \frac{1}{2\pi}\sum_{n'}\sum_{\omega'} Ri^*_{w^*}(n',\omega')Ri_x(n-n',\omega-\omega'),$$

where $Ri_x(n,\omega) = x(n)X^*(\omega)e^{-jn\omega}$.

The above definition shows that, at any location (n,ω), the spectrogram can be considered as the average of the weighted Rihaczek distribution values at the neighboring points $(n-n',\ \omega-\omega')$. Because this smoothing leads to broadening of the signal components, the authors suggested to change the attribution point of the average, assigning it to the center of gravity $(\hat{n}_x, \hat{\omega}_x)$ of the weighted distribution. This process yields a reassigned spectrogram (RS_x) whose value at any location is the sum of all representation values relocated to this point. Recently, Auger and Flandrin [4] simplified the reassignment algorithm by proposing new expressions to compute the relocation positions as a function of the time–frequency

coordinates (n, ω). It should be noted that these expressions presented below are equivalent to the original one introduced in [16]:

$$\hat{n}_x(n,\omega) = n - \frac{N}{2} + \mathrm{Re}\left\{\frac{F_x^{Tw}(n,\omega)F_x^{w}(n,\omega)^*}{|F_x^{w}(n,\omega)|^2}\right\}, \tag{10}$$

$$\hat{\omega}_x(n,\omega) = \omega - \mathrm{Im}\left\{\frac{F_x^{Dw}(n,\omega)F_x^{w}(n,\omega)^*}{|F_x^{w}(n,\omega)|^2}\right\}, \tag{11}$$

where

$$F_x^{Tw}(n,\omega) = \sum_{i=1}^{N} x(n+i)iw(i)\mathrm{e}^{-\mathrm{j}\omega i} \tag{12}$$

and

$$F_x^{Dw}(n,\omega) = \sum_{i=1}^{N} x(n+i)w'(i)\mathrm{e}^{-\mathrm{j}\omega i}. \tag{13}$$

In the above expression, $w'(i)$ is the derivative of the continuous-time function w, evaluated at time i.

$F_x^{w}(n,\omega)$ is recursively evaluated during the spectrogram computation and is also used in the reassignment process.

(ii) *Recursivity of the reassignment operators.* In order to introduce a fast algorithm dedicated to the reassignment stage computation, we want now to establish recursive expressions for Eqs. (12) and (13).

After a straightforward manipulation, Eq. (12) can be written as

$$F_x^{Tw}(n+1,\omega) = \left[\sum_{k=1}^{N}[k-1]x(n+k)w(k-1)\mathrm{e}^{-\mathrm{j}\omega k}\right]\mathrm{e}^{\mathrm{j}\omega} + Nx(n+N+1)w(N)\mathrm{e}^{-\mathrm{j}\omega N}. \tag{14}$$

When w verifies $w(k-1) = C\,w(k)$, Eq. (14) becomes

$$F_x^{Tw}(n+1,\omega) = C[F_x^{Tw}(n,\omega) - F_x^{w}(n,\omega)]\mathrm{e}^{\mathrm{j}\omega} + Nx(n+N+1)w(N)\mathrm{e}^{-\mathrm{j}\omega N}. \tag{15}$$

$F_x^{Tw}(n+1,\omega)$ is now easily determined by using $F_x^{Tw}(n,\omega)$, $F_x^{w}(n,\omega)$ and an additive correction term.

Let us now simplify Eq. (13) when w belongs to the class W which has been defined in Eq. (6):

$$w(k) = \alpha r^{-k}\mathrm{e}^{-\mathrm{j}\theta k} \Rightarrow w'(k) = -\alpha(\ln r + \mathrm{j}\theta)r^{-k}\mathrm{e}^{-\mathrm{j}\theta k} = -\ln(C)w(k).$$

Then $F_x^{Dw}(n+1,\omega)$ computation requires only one complex multiplication when using $F_x^{w}(n+1,\omega)$ value:

$$F_x^{Dw}(n+1,\omega) = -\ln(C)F_x^{w}(n+1,\omega). \tag{16}$$

It can also be shown that windows which belong to W_{ext} allow a recursive implementation of the reassignment operators. This can be done by using the linearity of the Fourier transform.

If

$$w(k) = \sum_{m=1}^{M} w_m(k) \quad \text{and} \quad w_m(k-1) = C_m w_m(k),$$

Eqs. (15) and (16) can be modified accordingly:

$$F_x^{\chi}(n+1,\omega) = \sum_{m=1}^{M} F_x^{\chi_m}(n+1,\omega), \tag{17}$$

where χ symbolizes Dw and Tw (see definitions (12) and (13)). Using the same notation, the $F_x^{\chi_m}$ $(m = 1, \ldots, M)$ functions are recursively evaluated using Eqs. (15) and (16) expressions.

(iii) *On-line recursive algorithm.* Our purpose is now to introduce a fast algorithm which allows the evaluation of classical (S_x) and reassigned spectrograms (RS_x). This algorithm is shown in Fig. 2.

In this algorithm, the two representations S_x and RS_x are evaluated using the recursive expressions established in Section 2. The results are stored in two matrices initially set to zero. At each time-step, the reassignment stage is executed if and only if the current (n, ω) spectrogram value exceeds an arbitrarily fixed threshold ε. This criterion guarantees the existence of Eqs. (10) and (11) and avoids the useless reassignment of representation values close to zero [4].

(iv) *Evaluation of the proposed algorithm.* To estimate the gain in computation time of the recursive approach, we use the same strategy as in Section 2.1 (iv). The total numbers of real operations required

MAIN FUNCTION

At each time instant (n + N) do

Acquisition of the sample x(n + N)

For each pulsation $\omega = \{2\pi k/N\}_{k=0,\ldots,N-1}$

Recursive evaluation of $F_x^w(n,\omega)$ using $F_x^w(n-1,\omega)$

$\mapsto$ see eq. (8) and (9)

Evaluation of the spectrogram : $S_x(n+N/2,\omega) = |F_x^w(n,\omega)|^2$

If $|F_x^w(n,\omega)|^2 > \varepsilon$

REASSIGNMENT FUNCTION computation

End if

End for

End do

REASSIGNMENT FUNCTION

Recursive evaluation of $F_x^{Tw}(n,\omega)$ using $F_x^{Tw}(n-1,\omega)$ and $F_x^w(n-1,\omega)$

$\mapsto$ see eq. (15) and (17)

Evaluation of $F_x^{Dw}(n,\omega)$ using $F_x^w(n,\omega)$

$\mapsto$ see eq. (16) and (17)

Evaluation of $\hat{n}_x(n,\omega)$ and $\hat{\omega}_x(n,\omega)$

$\mapsto$ see eq. (10) and (11)

Evaluation of the reassigned spectrogram :

$RS_x(\hat{n}_x,\hat{\omega}_x) = RS_x(\hat{n}_x,\hat{\omega}_x) + |F_x^w(n,\omega)|^2$

Fig. 2. Recursive reassigned spectrogram algorithm.

to compute the reassigned spectrogram are recapitulated in Table 3. Fig. 3 represents our procedure relative efficiency (RE) for several usual windows as a function of their respective length. These theoretical results clearly show the computational efficiency of this recursive strategy.

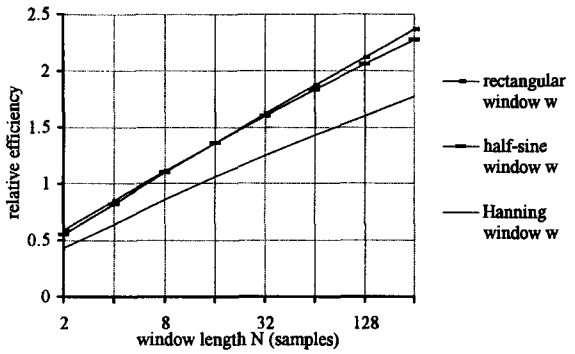

Fig. 3. Relative efficiency (RE) of the ressigned spectrogram recursive algorithm.

In order to evaluate its performance in practical applications, we apply our algorithm to a 1024-sample computer-generated signal. This signal is composed of a sine wave and a chirp components. It also includes two signals with constant amplitudes and instantaneous frequencies describing, respectively, a parabola and two sine periods. Fig. 4 represents the instantaneous frequency laws of the four signal components, giving a reference to appreciate the time–frequency representations.

We now apply our recursive procedure. The chosen analysis window is a 128-sample half-sine function. We simultaneously obtain the two representations given by Fig. 5. It should be noticed that these contour plots use the same levels corresponding to one-fifth, one-tenth and one-fifteenth of the distribution maximum value.

Fig. 5 clearly demonstrates that a great improvement is achieved by the use of the reassignment

Table 3
Efficiency of the recursive reassigned spectrogram algorithm, in real operations numbers

	Real additions number (A_R) (per time-step)		Real multiplications number (M_R) (per time-step)		Condition on N for
	Direct method	Recursive method	Direct method	Recursive method	RE $\geqslant 1$
Rectangular	$6N\log_2 N + 4N$	$16N + 2$	$4N\log_2 + 6N$	$16N + 2$	6
Half-sine	$9N\log_2 N + 9N$	$32N + 16$	$6N\log_2 N + 16N$	$28N + 24$	7
Hamming Hanning	$9N\log_2 N + 9N$	$46N + 20$	$6N\log_2 N + 16N$	$36N + 30$	13

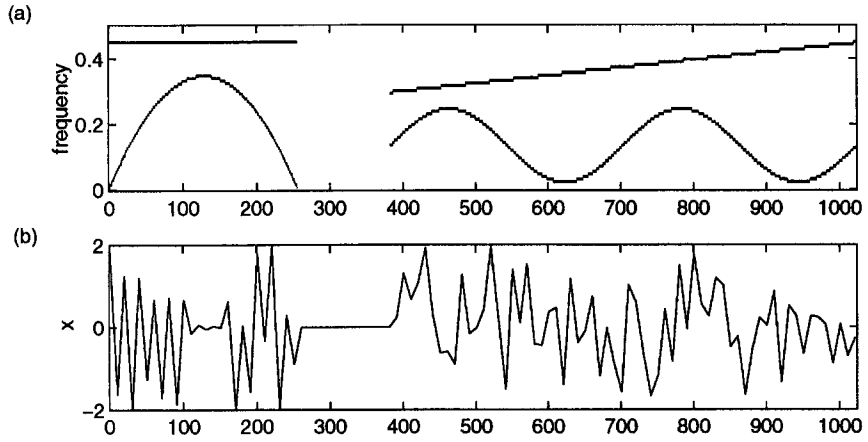

Fig. 4. (a) Instantaneous frequency laws of the four signal components. (b) Computer-generated signal.

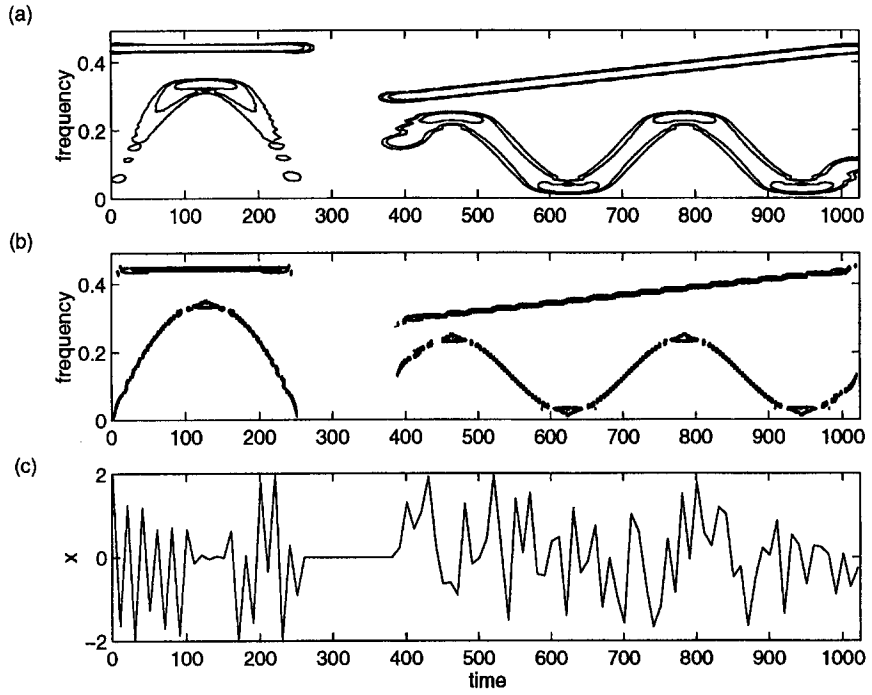

Fig. 5. (a) Recursive spectrogram windowed by a half-sine function ($N = 128$). (b) Version modified by the recursive reassignment stage. (c) Computer-generated signal.

method. The signal components are well concentrated and perfectly localized on the time–frequency map, allowing an easy interpretation of the representation. Moreover, the measure of the computation time required to calculate this representation by the direct and recursive methods (see Table 4) confirms the theoretical computational gain (RE) represented in Fig. 3. These results show

Table 4
Computation times required to compute the representations given by Fig. 5 (using MATLAB on a DEC station 3000)

Computation time (s)	Recursive approach	Direct application of the definitions	Gain
Spectrogram Computation	1.47	2.91	1.98
Reassignment Stage	3.93	8.16	2.07
Total	5.40	11.07	2.05

that our algorithm allows to reduce computation time of spectrograms modified by the reassignment method.

3. Extension to Cohen's class TFDs

3.1. CTFDs recursive implementation

(i) *Problem formulation.* The discrete Wigner–Ville distribution, which has been extensively studied in recent years, is defined as follows [9]:

$$\mathrm{WV}_x(n,\omega) = \sum_{m=-\infty}^{+\infty} R_x(n,m)\mathrm{e}^{-\mathrm{j}2\omega m},$$

where $R_x(n,m) = x(n+m)\,x^*(n-m)$ is the instantaneous autocorrelation function of the signal x.

This distribution is well known for its high resolution in the time–frequency plane and the large number of properties it satisfies [12]. Unfortunately, its use in practical applications is limited due to the numerous non-negligible cross-components generated by its bilinear structure. However, to get around this difficulty, one can apply a bidimensional filter ψ to the instantaneous autocorrelation function R_x. This operation leads to the following definition of Cohen's class time–frequency distributions (CTFD) [4, 12, 17]:

$$\mathrm{CTFD}_x^{\psi}(n,\omega) = \sum_{m=-\infty}^{+\infty}\sum_{p=-\infty}^{+\infty} \psi(p,m) \times R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m},$$

where ψ is called the autocorrelation-domain kernel in [10].

We consider now that ψ has a finite support S_ψ defined as follows:

$$S_\psi = \{(p,m)\in Z:\ |p| \leqslant L, |m| \leqslant N-1\},$$

and satisfies

$$\psi(p,-m) = \psi^*(p,m) \quad \text{for all } (p,m)\in S_\psi. \tag{18}$$

This last hypothesis preserves the general aspect of our problem and ensures the distribution to be real-valued. Moreover, it allows a faster computation of CTFDs, as it is shown by Eqs. (19) and (20) [8, 17]. As a consequence, ψ will be considered as conjugate-symmetric for the remainder of section 3.

It can be written after some manipulations:

$$\mathrm{CTFD}_x^{\psi}(n,\omega) = 2\,\mathrm{Re}[\mathrm{TF}_x^{\psi}(n,\omega)] - \sum_{p=-L}^{L}\psi(p,0)R_x(n+p,0), \tag{19}$$

where

$$\mathrm{TF}_x^{\psi}(n,\omega) = \sum_{m=0}^{N-1}\sum_{p=-L}^{L}\psi(p,m)R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m}. \tag{20}$$

Evaluating CTFDs by the direct application of Eq. (20) requires the calculation of the smoothed autocorrelation function and an N-sample FFT at each time instant. This algorithm is computationally expensive and, consequently, cannot be easily used in a real-time context. Our purpose is now to introduce a fast method, based on a recursive approach, dedicated to TF_x^{ψ} evaluation.

(ii) *Presentation of the recursion.* A strategy equivalent to the one introduced in Section 2.1 (ii) is used to propose a recursive implementation of CTFDs. Eq. (20) can be written as

$$\mathrm{TF}_x^{\psi}(n+1,\omega) = \sum_{m=0}^{N-1}\sum_{p=-L+1}^{L+1}\psi(p-1,m)\,R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m}. \tag{21}$$

If the following condition is verified:

$$\psi(p-1,m) = C\psi(p,m), \tag{22}$$

where C is independent of m, Eq. (21) becomes

$$\mathrm{TF}_x^{\psi}(n+1,\omega) = C\mathrm{TF}_x^{\psi}(n,\omega) + \sum_{m=0}^{N-1} \varphi(n,m)\mathrm{e}^{-\mathrm{j}2\omega m}. \quad (23)$$

In the above expression, $\varphi(n, m)$ is given by

$$\varphi(n,m) = \psi(L,m)R_x(n+L+1,m) - C\psi(-L,m)R_x(n-L,m). \quad (24)$$

Therefore, $\mathrm{TF}_x^{\psi}(n+1,\omega)$ is determined using its previous value and an additive correction term, easily evaluated by an N-sample FFT.

In order to define the family of windows which allow recursion use in CTFDs evaluation, we have to solve the following functional equation:

$\psi(p-1,m) = C\psi(p,m) \Leftrightarrow \psi(p,m) = C^{-p}\psi(0,m) = r^{-p}\mathrm{e}^{-\mathrm{j}\theta p}h(m)$, using the complex notation $C = r\mathrm{e}^{\mathrm{j}\theta}$,

are replacing $\psi(0, m)$ by any function h of variable m.

The family G of solutions can be defined as

$$G = \{\psi \mid \psi(p,m) = g(p)h(m) \text{ where } g(p) = r^{-p}\mathrm{e}^{-\mathrm{j}\theta p}\}. \quad (25)$$

We can notice that G is composed of functions with separable (p, m) variables. Then, our algorithm only authorizes the evaluation of smoothed pseudo-Wigner–Ville distributions (SPWVD). This constraint on the distribution choice is not restrictive. With its separable kernel, the SPWVD allows the time and frequency smoothing to be adjusted independently, making this representation one of the most versatile of Cohen's class distributions. However, it should be mentioned that this class of kernels does not allow a directional time–frequency smoothing, which is sometimes useful (when analyzing multicomponent chirp signals for example).

(iii) *Generalization of the recursive scheme.* As in Section 2.1(iii), we wish now to extend the candidate windows family G. Thus, if G_{ext} is defined as follows:

$$G_{\mathrm{ext}} = \{\psi \mid \psi(p,m) = h(m)\sum_{t=1}^{T} g_t(p) \text{ where } g_t(p) = r_t^{-p}\mathrm{e}^{-\mathrm{j}\theta_t p}\},$$

it is straightforward to show that each G_{ext} candidate will permit a recursive implementation of the SPWVD:

$$\mathrm{TF}_x^{\psi}(n+1,\omega) = \sum_{t=1}^{T} \mathrm{TF}_x^{\psi_t}(n+1,\omega),$$

where

$$\psi_t(p,m) = h(m)r_t^{-p}\mathrm{e}^{-\mathrm{j}\theta_t p}, \quad C_t = r_t\mathrm{e}^{\mathrm{j}\theta_t} \text{ and}$$

$$\mathrm{TF}_x^{\psi_t}(n+1,\omega) = C_t\mathrm{TF}_x^{\psi_t}(n,\omega) + \sum_{m=0}^{N-1} \varphi_t(n,m)\mathrm{e}^{-\mathrm{j}2\omega m}.$$

In the above expression, $\varphi_t(n, m)$ is given by

$$\varphi_t(n,m) = [g_t(L)R_x(n+L+1,m) - C_t g_t(-L)R_x(n-L,m)]h(m).$$

It can also be shown that usual truncating windows are elements of G_{ext}: rectangular ($T = 1$), half-sine ($T = 2$), Hamming ($T = 3$), Hanning ($T = 3$) and Blackman ($T = 5$) windows, considered now as functions of variable p and post multiplied by the function $h(m)$. One can see Table 1 to use these windows in the SPWVD recursion.

(iv) *Evaluation of the proposed algorithm.* To evaluate the gain in computational time of the SPWVD recursive algorithm, we apply a similar strategy to the one introduced in the first section. The relative efficiency of our method is now defined as

RE = (TNRM required to compute the representation at time n by the direct application of (19) and (20)) *divided by* (TNRM required to compute the representation at time n using our algorithm).

As an example, we estimate now the relative efficiency of our approach when g is a rectangular window and h any complex-valued function. At each time-step, N instantaneous autocorrelation values must be evaluated, which requires N complex multiplications ($M_c = N$). In the case of the direct application of Eq. (20), these instantaneous autocorrelations are summed for any m values. This computation requires $2NL$ complex additions ($A_c = 2NL$). Then, the result is windowed by the

Table 5
Efficiency of the recursive SPWVD algorithm, in real operations numbers

	Real additions number (A_R) (per time-step)		Real multiplications number (M_R) (per time-step)		Condition on L for RE $\geqslant 1$
	Direct method	Recursive method	Direct method	Recursive method	($N = 64$)
Rectangular	$3N\log_2 N + 4(L+1)N$	$3N\log_2 N + 7N$	$2N\log_2 N + 7N$	$2N\log_2 N + 7N$	1
Half-sine	$3N\log_2 N + 4(2L+1)N$	$6N\log_2 N + 22N$	$2N\log_2 N + (8L+7)N$	$4N\log_2 N + 25N$	4
Hamming Hanning	$3N\log_2 N + 4(2L+1)N$	$9N\log_2 N + 27N$	$2N\log_2 N + (8L+7)N$	$6N\log_2 N + 27N$	6

complex function h ($M_c = N$) and transformed by means of an N-sample FFT. The SPWVD is obtained applying Eq. (19) ($M_R = N$, $A_R = N$). Thus, the total numbers of real additions (A_R) and multiplications (M_R) are

$$A_R = 4(L+1)N + 3N\log_2(N),$$

$$M_R = 7N + 2N\log_2(N).$$

Using the recursion given by Eq. (23) and since $C = 1$, the same computation requires N complex additions and $2N$ complex multiplications to evaluate φ (see Eq. (24)). Then, the FFT of φ is determined and the real part of the result is added to the real part of the previous TF_x^{ψ} value ($A_R = N$). The SPWVD is obtained applying Eq. (19) ($M_R = N$, $A_R = N$). Then, the total number of real operations can be decomposed as

$$A_R = 7N + 3N\log_2(N),$$

$$M_R = 7N + 2N\log_2(N).$$

We recapitulate in Table 5 the total numbers of real operations required to compute the SPWVD when using the recursive formulation or the Eq. (20) direct implementation.

Fig. 6 represents the relative efficiency of our algorithm for several commonly used windows. In this example, we chose to evaluate the distribution on 127 autocorrelation values ($N = 64$) at each time step. One can notice that we have not studied the case when g is a rectangular window: direct and recursive approaches require the same number of real multiplications so that RE is always equal to one. However, computing the recursive algorithm needs less real additions, which makes it more efficient.

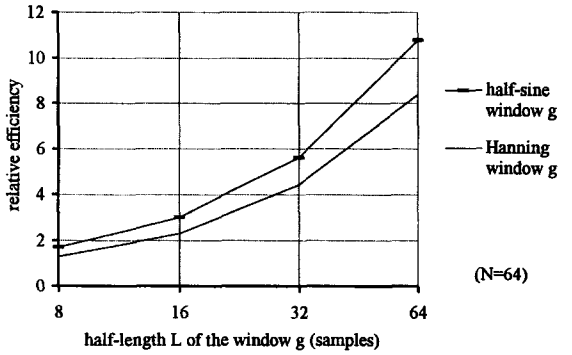

Fig. 6. Relative efficiency (RE) of the SPVWD recursive algorithm (N fixed to 64).

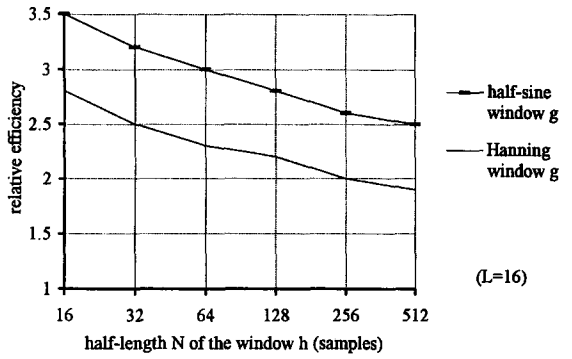

Fig. 7. Relative efficiency (RE) of the SPVWD recursive algorithm (L fixed to 16).

In Fig. 7, we present RE as a function of N when g is a 33-sample half-sine window ($L = 16$).

These theoretical results clearly demonstrate that the recursive approach allows to greatly improve the usual computational performances.

3.2. Reassignment stage recursive implementation

(i) *Problem formulation.* The SPWVD is characterized by a separable kernel which allows the time and frequency smoothing to be adjusted independently. Unfortunately, smoothing also produces a less accurate localization of the signal autocomponents in the time–frequency plane. In order to increase signal concentration, Auger and Flandrin showed that the reassignment method can be advantageously applied to any CTFDs and particularly to SPWVDs [4]. In that case, the following expressions are proposed to compute the relocation positions as a function of the time–frequency coordinates (n, ω):

$$\hat{n}_x(n,\omega) = n - \frac{\mathrm{SPWVD}_x^{\mathrm{Tgh}}(n,\omega)}{\mathrm{SPWVD}_x^{\mathrm{gh}}(n,\omega)}, \tag{26}$$

$$\hat{\omega}_x(n,\omega) = \omega + \mathrm{j}\,\frac{\mathrm{SPWVD}_x^{\mathrm{gDh}}(n,\omega)}{\mathrm{SPWVD}_x^{\mathrm{gh}}(n,\omega)}, \tag{27}$$

Using the same notations as in [4], the symbols gh, Tgh and gDh represent, respectively, the autocorrelation-domain kernels $g(p)h(m)$, $p\,g(p)h(m)$ and $g(p)h'(m)$, so that

$$\mathrm{SPWVD}_x^{\mathrm{gh}}(n,\omega) = \sum_{m=1-N}^{N-1}\sum_{p=-L}^{L} g(p)h(m)R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m}, \tag{28}$$

$$\mathrm{SPWVD}_x^{\mathrm{gDh}}(n,\omega) = \sum_{m=1-N}^{N-1}\sum_{p=-L}^{L} g(p)h'(m)R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m}, \tag{29}$$

$$\mathrm{SPWVD}_x^{\mathrm{Tgh}}(n,\omega) = \sum_{m=1-N}^{N-1}\sum_{p=-L}^{L} pg(p)h(m)R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m}. \tag{30}$$

At each time step, the reassignment method uses the current distribution value and requires the calculation of two additional SPWVDs. Let us find an algorithm to compute faster the reassignment operators given by Eqs. (26) and (27).

(ii) *Recursivity of the reassignment operators.* In Section 3.1(ii), we introduced a fast recursive approach to implement Eq. (28) when the distribution kernel gh belongs to G. Our purpose is now to extend this method to Eqs. (29) and (30) evaluations.

When the kernel $\mathrm{gh}(p, m)$ is conjugate-symmetric (see the notation above), i.e., satisfies $gh(p, -m) = gh(p, m)^*$, it can be shown that

$$[g(p)h'(-m)]^* = -g(p)h'(m),$$

i.e. $\mathrm{gDh}(p, -m)^* = -\mathrm{gDh}(p, m)$

and

$$[p\,g(p)h(-m)]^* = p\,g(p)h(m),$$

i.e. $\mathrm{Tgh}(p, -m)^* = \mathrm{Tgh}(p, m)$

Therefore, similar decompositions as the one introduced by Eqs. (19) and (20) (see also [8, 17]) can be applied to Eqs. (29) and (30). The following expressions allow a faster reassignment stage computation:

$$\mathrm{SPWVD}_x^{\mathrm{gDh}}(n,\omega) = 2\,\mathrm{Im}[\mathrm{TF}_x^{\mathrm{gDh}}(n,\omega)] - \sum_{p=-L}^{L} g(p)h'(0)R_x(n+p,0) \tag{31}$$

and

$$\mathrm{SPWVD}_x^{\mathrm{Tgh}}(n,\omega) = 2\,\mathrm{Re}[\mathrm{TF}_x^{\mathrm{Tgh}}(n,\omega)] - \sum_{p=-L}^{L} p\,g(p)h(0)R_x(n+p,0), \tag{32}$$

where $\mathrm{TF}_x^{\mathrm{gDh}}$ and $\mathrm{TF}_x^{\mathrm{Tgh}}$ are defined as

$$\mathrm{TF}_x^{\mathrm{gDh}}(n,\omega) = \sum_{m=0}^{N-1}\sum_{p=-L}^{L} g(p)h'(m)R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m} \tag{33}$$

and

$$\mathrm{TF}_x^{\mathrm{Tgh}}(n,\omega) = \sum_{m=0}^{N-1}\sum_{p=-L}^{L} p\,g(p)h(m)R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m}. \tag{34}$$

We use now the same approach as in Section 3.1(ii) to propose a recursive formulation of Eq. (33). After a straightforward calculus, we can write

$$\mathrm{TF}_x^{\mathrm{gDh}}(n+1,\omega) = \sum_{m=0}^{N-1}\sum_{p=-L+1}^{L+1} g(p-1)h'(m) \times R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m}. \quad (35)$$

Considering that gh belongs to the family G (g verifies $g(p-1) = C\,g(p)$), Eq. (35) becomes

$$\mathrm{TF}_x^{\mathrm{gDh}}(n+1,\omega) = C\,\mathrm{TF}_x^{\mathrm{gDh}}(n,\omega) + \sum_{m=0}^{N-1}\eta(n,m)\mathrm{e}^{-\mathrm{j}2\omega m}. \quad (36)$$

In the above recursion, η(n, m) is given by

$$\eta(n,m) = [g(L)R_x(n+L+1,m) - Cg(-L)R_x(n-L,m)]h'(m). \quad (37)$$

Let us introduce now a recursive expression of $\mathrm{TF}_x^{\mathrm{Tgh}}$. After some manipulations, it can be shown that

$$\mathrm{TF}_x^{\mathrm{Tgh}}(n+1,\omega) = \sum_{m=0}^{N-1}\sum_{p=-L+1}^{L+1}(p-1)g(p-1)h(m) \times R_x(n+p,m)\mathrm{e}^{-\mathrm{j}2\omega m}. \quad (38)$$

From the above equation and the relation $g(p-1) = C\,g(p)$, one can deduce that

$$\mathrm{TF}_x^{\mathrm{Tgh}}(n+1,\omega) = C[\mathrm{TF}_x^{\mathrm{Tgh}}(n,\omega) - \mathrm{TF}_x^{\mathrm{gh}}(n,\omega)] + \sum_{m=0}^{N-1}\mu(n,m)\mathrm{e}^{-\mathrm{j}2\omega m}, \quad (39)$$

where

$$\mu(n,m) = [Lg(L)R_x(n+L+1,m) + C(L+1)g(-L)R_x(n-L,m)]h(m). \quad (40)$$

The equations established previously are verified when the distribution kernel gh belongs to G. These expressions can be easily extended to G_{ext} family by using the same procedure as in the reassigned spectrogram case.

MAIN FUNCTION

At each time instant (n + N + L - 1) do
- Acquisition of the sample x(n + N + L - 1)
- For each pulsation $\omega = \{2\pi k/N\}_{k=0,\ldots,N-1}$
  - Recursive evaluation of $\mathrm{TF}_x^{gh}(n,\omega)$ using $\mathrm{TF}_x^{gh}(n-1,\omega)$
    - ↦ see eq. (23) and (24)
  - Evaluation of $\mathrm{SPWVD}_x^{gh}(n,\omega)$
    - ↦ see eq. (19)
  - If $\mathrm{SPWVD}_x^{gh}(n,\omega) > \varepsilon$
    - *REASSIGNMENT FUNCTION* computation
  - End if
- End for

End do

REASSIGNMENT FUNCTION

Recursive evaluation of $\mathrm{SPWVD}_x^{gDh}(n,\omega)$ using $\mathrm{TF}_x^{gDh}(n-1,\omega)$
- ↦ see eq. (31), (36) and (37)

Recursive evaluation of $\mathrm{SPWVD}_x^{Tgh}(n,\omega)$ using $\mathrm{TF}_x^{Tgh}(n-1,\omega)$ and $\mathrm{TF}_x^{gh}(n-1,\omega)$
- ↦ see eq. (32), (39) and (40)

Evaluation of $\hat{n}_x(n,\omega)$ and $\hat{\omega}_x(n,\omega)$
- ↦ see eq. (26) and (27)

Evaluation of the reassigned SPWVD :

$\mathrm{RSPWVD}_x^{gh}(\hat{n}_x,\hat{\omega}_x) = \mathrm{RSPWVD}_x^{gh}(\hat{n}_x,\hat{\omega}_x) + \mathrm{SPWVD}_x^{gh}(n,\omega)$

Fig. 8. Recursive reassigned SPWVD algorithm.

(iii) *On-line recursive algorithm.* In Fig. 8, we present a fast algorithm dedicated to classical and reassigned SPWVDs (RSPWVD) evaluations.

The two representations SPWVD_x^{gh} and RSPWVD_x^{gh} are evaluated using the recursive expressions established in Section 3. As in the spectrogram case, the reassignment stage is executed if and only if the current (n, ω) SPWVD value exceeds an arbitrary fixed threshold ε. This criterion guarantees the existence of Eqs. (26) and (27) and avoids the useless reassignment of representation values close to zero [4].

(iv) *Evaluation of the proposed algorithm.* To estimate the computational efficiency of our algorithm, we use the same method as in Section 2.1(i). The total numbers of real operations required to compute reassigned SPWVDs are recapitulated in Table 6. These evaluations are done in the case of

Table 6
Efficiency of the recursive reassigned SPWVD algorithm, in real operations numbers

	Real additions number (A_R) (per time-step)		Real multiplications number (M_R) (per time-step)		Condition on L for RE $\geqslant 1$
	Direct method	Recursive method	Direct method	Recursive method	($N = 64$)
Rectangular	$9N\log_2 N + (12L + 11)N$	$9N\log_2 N + 19N$	$6N\log_2 N + (8L + 15)N$	$6N\log_2 N + 19N$	1
Half-sine	$9N\log_2 N + (16L + 13)N$	$18N\log_2 N + 66N$	$6N\log_2 N + (16L + 19)N$	$12N\log_2 N + 70N$	6
Hamming Hanning	$9N\log_2 N + (16L + 13)N$	$27N\log_2 N + 81N$	$6N\log_2 N + (16L + 19)N$	$18N\log_2 N + 82N$	9

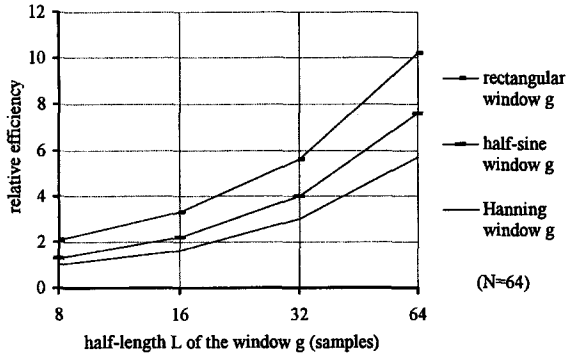

Fig. 9. Relative efficiency (RE) of the reassigned SPVWD recursive algorithm (N fixed to 64).

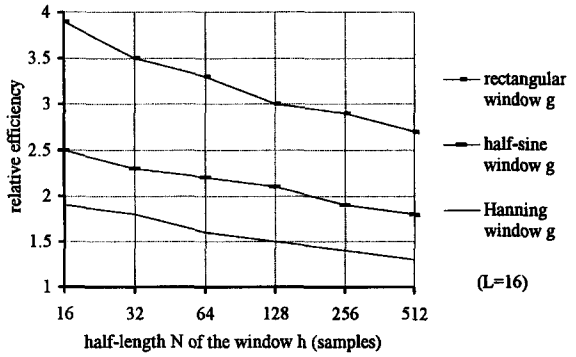

Fig. 10. Relative efficiency (RE) of the reassigned SPVWD recursive algorithm (L fixed to 16).

our recursive method and the direct application of Eqs. (19), (31) and (32).

Fig. 9 represents the relative efficiency (RE) of our algorithm as a function of the half-length L of the window g. In this example, $h(m)$ is a 127-sample complex-valued window ($N = 64$). In Fig. 10, we present RE as a function of N when g is a 33-sample half-sine window ($L = 16$). These theoretical results clearly show the recursive strategy computational efficiency.

We now apply our algorithm to the 1024-sample computer-generated signal defined in Section 2.2 (iv), in order to evaluate performances in practical applications. In the following examples, g is a rectangular window and h a 127-sample Hamming window. The two representations given by Fig. 11 are obtained simultaneously. It should be noticed that Figs. 11(a) and (b) use the same contour plots corresponding to one-fifth, one-tenth and one-fifteenth of the distribution maximum value.

Fig. 11(a) shows the SPWVD of our signal of reference when g is a 9-point rectangular window ($L = 4$). The signal components are well localized but the presence of misleading interferences makes the visual interpretation difficult. In Fig. 11(b), we can notice that the improvement given by the reassignment method is obvious: the auto and cross-components become perfectly concentrated. Let us now use a longer time-smoothing rectangular window ($L = 16$) in order to reduce the numerous oscillating cross-terms.

Fig. 12(a) shows that there are now less interferences but the signal concentration is weaker. Fig. 12(b) shows a great improvement in the readability when using the reassignment process. The signal auto-components are strongly localized and all cross-terms are removed, giving a nearly ideal

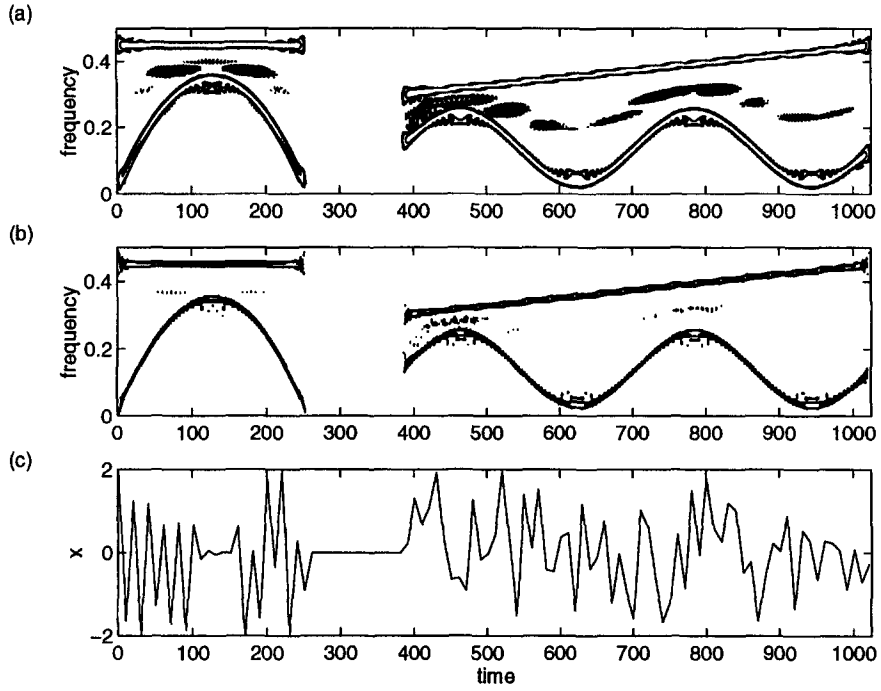

Fig. 11. (a) Recursive SPWVD when g is a rectangular window ($L = 4$) and h a Hamming window ($N = 64$). (b) Version modified by the recursive reassignment stage. (c) Computer-generated signal.

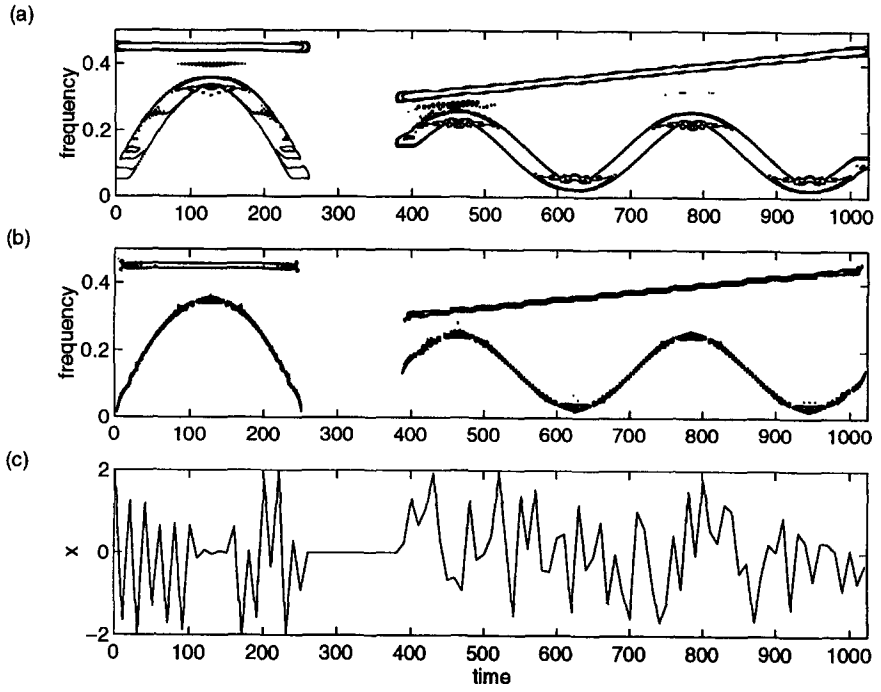

Fig. 12. (a) Recursive SPWVD when g is a rectangular window ($L = 16$) and h a Hamming window ($N = 64$). (b) Version modified by the recursive reassignment stage. (c) Computer-generated signal.

Table 7
Computation times required to compute the representations given by Fig. 12 (using MATLAB on a DEC station 3000)

Computation time (s)	Recursive approach	Direct application of the definitions	Gain
SPWVD computation	1.8	6.24	3.42
Reassignment stage	3.64	12.92	3.54
Total	5.4	19.16	3.50

representation of our computer-generated signal. Moreover, the measure of the computation time required to calculate this representation by the direct and the recursive methods (see Table 7) confirms the theoretical computational gain (RE) predicted in Fig. 9.

These results clearly show that our algorithm allows to reduce computation time of SPWVDs modified by the reassignment method. It should be noticed that the reassignment process must be associated with a 'large-enough' time-smoothing window g. It allows both to improve representations readability by removing interferences, and to increase the computational advantage of our recursive approach on usual methods.

4. Conclusion

In this paper, new fast algorithms which allow the evaluation of classical CTFDs and CTFDs modified by the reassignment method have been introduced. These algorithms use a recursive approach to compute both the time–frequency representation and the reassignment stage. In both TFD cases (spectrogram and CTFD), analysis windows must have a complex exponential structure in order to permit a recursive implementation. If this limitation is not restrictive in the case of the spectrogram, it only authorizes the evaluation of Cohen's class-smoothed pseudo-Wigner–Ville distributions. However, the families of candidate windows can be easily extended using the Fourier series expansion of the (assumed periodic) window over a family of exponential functions. Then, the recursive algorithms computational efficiency strongly depends on the number of decomposition terms, and any window can be chosen as long as it admits a 'short-enough' Fourier series decomposition. As an example, half-sine, Hamming, Hanning and Blackman functions, well-known in spectral analysis for their properties, can be advantageously used since they allow an efficient recursive implementation. The examples mentioned above show that the families of candidate windows are sufficiently rich. Moreover, the theoretical and experimental results introduced in that paper clearly demonstrate the computational efficiency superiority of our recursive approach on usual algorithms. Finally, it should be noticed that these recursions can be extended to other representations such as the Margenau–Hill distribution (see definition in [3, 4]) modified by the reassignment method.

References

[1] M.G. Amin, The running time–frequency distributions, Circuits Systems Signal Process. 14 (3) (1995) 401–414.
[2] M.G. Amin, Recursive kernels for time–frequency signal representations, IEEE Signal Process. Lett. 3 (1) (1996) 16–18.
[3] F. Auger, Représentation Temps-Fréquence des Signaux Non-Stationnaires: Synthèse et Contribution, Thèse Doctorat, ECN, Nantes, France, 1991.
[4] F. Auger, P. Flandrin, Improving the readability of time–frequency and time-scale representations by the reassignment method, IEEE Trans. Signal Process. 43 (5) (1995) 1068–1089.
[5] R.G. Baraniuk, D.L. Jones, Signal-dependent time–frequency analysis using a radially Gaussian kernel, Signal Processing 32 (1993) 263–284.
[6] D.T. Barry, Fast calculation of the Choi–Williams time frequency distribution, IEEE Trans. Signal Process. 40 (2) (1992) 450–455.
[7] M. Bellanger, Traitement Numérique du Signal – Théorie et Pratique, Masson, Paris, France, 1987.
[8] B. Boashash, P.J. Black, An efficient real-time implementation of the Wigner–Ville distribution, IEEE Trans. Acoust. Speech Signal Process. 35 (11) (1987) 1611–1619.
[9] G.S. Cunningham, W.J. Williams, Fast computation of the Wigner distribution for finite length signal, Proc. IEEE Internat. Conf. Acoust. Speech Signal Process., San-Francisco, USA, 1992, pp. 177–180.

[10] G.S. Cunningham, W.J. Williams, Fast implementations of generalized discrete time-frequency Distributions, IEEE Trans. Signal Process. 42 (6) (1994) 1496–1508.
[11] P. Flandrin, Some features of the time–frequency representations of multicomponents signals, Proc. IEEE Internat. Conf. Acoust. Speech Signal Process., San Diego, USA, 1984, pp. 41B.6.1–41B6.4.
[12] P. Flandrin, Temps–Fréquence, Hermes, Paris, France, 1993.
[13] E. Grall-maes, P. Beauseroy, Elimination of cross-compnents of the discrete Wigner–Ville distribution via a correlation method, Proc. 3rd Internat. Conf. Signal Process., Beijing, China, 1996, pp. 335–338.
[14] D.L. Jones, R.G. Baraniuk, An adaptive optimal-kernel time-frequency representation, IEEE Trans. Acoust. Speech Signal Process. 43 (10) (1995) 2361–2371.
[15] D.L. Jones, T.W. Parks, A high resolution data adaptative time–frequency reprersentation: optimal kernel design, IEEE Trans. Acoust. Speech Signal Process. 38 (1990) 2127–2135.
[16] K. Kodera, C. de Villedary, R. Gendrin, Analysis of time-varying signals with small BT values, IEEE Trans. Acoust. Speech Signal Process. 34 (1) (1978) 64–76.
[17] W. Martin, P. Flandrin, Wigner–Ville spectral analysis of nonstationary processes, IEEE Trans. Acoust. Speech Signal Process. 33 (6) (1985) 1461–1469.
[18] M. Sun, C.C. Li, L.N. Sekhar, R.J. Sclabassi, Elimination of cross-components of the discrete pseudo Wigner distribution via image processing, Proc. IEEE Internat. Conf. Acoust. Speech Signal Process., Glasgow, UK, 1989, pp. 2230–2233.