

Research Article

Signal Adaptive System for Space/Spatial-Frequency Analysis

Veselin N. Ivanović (EURASIP Member) and Srdjan Jovanovski

Department of Electrical Engineering, University of Montenegro, Cetinjski put bb, 81000 Podgorica, Montenegro

Correspondence should be addressed to Veselin N. Ivanović, very@ac.me

Received 17 August 2009; Revised 12 November 2009; Accepted 2 December 2009

Recommended by Mark Kahrs

This paper outlines the development of a multiple-clock-cycle implementation (MCI) of a signal adaptive two-dimensional (2D) system for space/spatial-frequency (S/SF) signal analysis. The design is based on a method for improved S/SF representation of the analyzed 2D signals, also proposed here. The proposed MCI design optimizes critical design performances related to hardware complexity, making it a suitable system for real time implementation on an integrated chip. Additionally, the design allows the implemented system to take a variable number of clock cycles (CLKs) (the only necessary ones regarding desirable—2D Wigner distribution—presentation of autoterms) in different frequency-frequency points during the execution. This ability represents a major advantage of the proposed design which helps to optimize the time required for execution and produce an improved, cross-terms-free S/SF signal representation. The design has been verified by a field-programmable gate array (FPGA) circuit design, capable of performing S/SF analysis of 2D signals in real time.

Copyright © 2009 V. N. Ivanović and S. Jovanovski. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. Introduction

Systems used in nonstationary 1D and 2D signals processing are based on the developed mathematical methods (distributions), defined in their 1D, [1–7], and 2D, [8–19], forms, respectively. The short-time Fourier transform (STFT), its energetic version-spectrogram (SPEC), and the Wigner distribution (WD) are the conventional mathematical tools commonly used in nonstationary signal analysis, [1–5, 8–10, 20]. Having in mind the technology limitations in the hardware design, the 1D systems based on these methods are analyzed, usually in their single-clock-cycle (parallel) implementation forms, [21–25]. However, conventional methods exhibit serious drawbacks. 1D and 2D STFT and the corresponding SPECS have a low concentration around signals' instantaneous and local frequency, respectively, [1–5, 8–10, 20], whereas 1D and 2D WD generate emphatic interference effects (cross-terms) in the case of multicomponent signal analysis, [1–6, 10–16]. To deal with the drawbacks of conventional methods, various mathematical tools for nonstationary signal analysis have been defined during the last two decades, [1–5, 10–12]. Some of them are computationally quite complex and, therefore, sometimes

cannot be implemented in real time. For example, some of the reduced interference methods, [1–4], are introduced to suppress cross-terms in the case of multicomponent signal analysis with preservation of marginal properties of the WD. However, they are defined in the computationally very intensive ways that result in the calculation of the 2D convolution in the 1D signals case and the 4D convolution in the 2D signal case. Quite computationally complex definition of these methods seriously compromise possibilities of their realization, [1, 3–5]. On the other side, the reduced interference method proposed in [5], named the S-method (SM), extended to the 2D form in [12] and frequently used, [26–28], reduces cross-terms with preservation of the WD autoterms. In addition, it is defined in a computationally simple way that requires calculation of the 1D convolution in the 1D signals case, [5], and the 2D convolution in the 2D signal case, [12]. In this way, it simultaneously enables an efficient real time implementation. Parallel designs for 1D signal analysis and time-varying filtering, based on the SM, have been developed in [29–31]. But, parallel designs are quite complex and require duplication of basic calculation elements when they are employed more than once. Also, they can process signals with the predefined duration only.

The 1D MCI hardware design with a fixed number of CLKs, based on the SM, is recently developed, [32]. It overcomes the drawbacks of parallel architectures, qualifying itself to be an optimal solution for wide range of practical implementations. However, this solution significantly decreases the processing speed, thus making it inconvenient in some other applications. Further, in order to preserve each WD autoterm separately, the SM based systems should correspond to the widest signal component, [5, 12], that can negatively influence the cross-terms reduction, calculation complexity, and the processing speed, [29–32].

Corresponding 2D S/SF systems are more complex than the 1D ones and sometimes their parallel implementation forms, like the one based on the 2D SM, [33], could not be implemented. Additionally, the chip dimensions, power consumptions and cost are significantly increased, while the processing speed is lowered, especially if the MCI designs of such systems are preferred, [34]. Therefore, here we propose a way to overcome the drawbacks of the 2D parallel implementation forms and the 2D MCI forms. For that purpose, a special signal adaptive MCI hardware design for S/SF analysis has been developed (and verified) based on a 2D mathematical method for an improved S/SF representation of analyzed 2D signals (also proposed here). The proposed hardware design allows the implemented system to take a variable number of CLKs in different frequency-frequency points within the execution and, therefore, to produce a pure cross-terms-free S/SF signal representation that retains the desirable autoterms presentation of the 2D WD. In this way, the design optimizes the execution time, overcoming the main drawback of the MCI designs in comparison to the parallel ones. In addition, the design optimizes the hardware complexity of the implemented system, giving one the possibility to implement it by using standard devices like FPGA. Such a practical design with all implementation and verification details is presented here.

The paper is organized as follows. In Section 2, the 2D mathematical method for improved S/SF signal representation is proposed theoretically and compared with the commonly used S/SF distributions (S/SFDs) regarding the signal presentation, calculation complexity and noise influence suppression. The proposed method based hardware design is developed in Section 3. Testing and verification results are elaborated in Section 4. Trade-offs and comparisons of the proposed design with other designs for S/SF analysis (the parallel one and the MCI one with a fixed number of CLKs) are discussed in Section 5.

2. Theoretical Background

As mentioned, the 2D STFT, its energetic version (2D SPEC) and the 2D WD are conventional mathematical methods, used in S/SF signal analysis. They are defined, in vector notation, as [8–10, 12–16]

$$\begin{aligned} \text{STFT}(\vec{n}, \vec{k}) \\ = \sum_{\vec{m}} w(\vec{m}) f(\vec{n} + \vec{m}) e^{-j(2\pi/N) \vec{k} \vec{m}}, \end{aligned} \quad (1)$$

$$\begin{aligned} \text{WD}(\vec{n}, \vec{k}) \\ = \sum_{\vec{m}} w(\vec{m}) w(-\vec{m}) f(\vec{n} + \vec{m}) f^*(\vec{n} - \vec{m}) e^{-j2(2\pi/N) \vec{k} \vec{m}}, \end{aligned} \quad (2)$$

where $w(\vec{m}) = w(m_1, m_2)$ denotes a 2D, usually even and real-valued lag-window of $N \times N$ duration, centered at the point $\vec{n} = (n_1, n_2)$ and used to truncate the analyzed signal $f(\vec{n})$. However, these S/SFDs exhibit drawbacks that seriously limit their applicability. The 2D STFT and 2D SPEC have a low concentration around signal's local frequency [8–10, 12]. On the other hand, based on direct 2D STFT-to-2D WD relationship, [12], readily following from (1)-(2):

$$\begin{aligned} \text{WD}(\vec{n}, \vec{k}) &= \sum_{\vec{i}} \text{STFT}(\vec{n}, \vec{k} + \vec{i}) \text{STFT}^*(\vec{n}, \vec{k} - \vec{i}) \\ &= \left| \text{STFT}(\vec{n}, \vec{k}) \right|^2 \\ &\quad + 2 \sum_{i_1=0}^{N/2} \sum_{i_2=1}^{N/2} \text{Re}\{\text{STFT}(\vec{n}, k_1 + i_1, k_2 + i_2) \\ &\quad \quad \text{STFT}^*(\vec{n}, k_1 - i_1, k_2 - i_2)\} \\ &\quad + 2 \sum_{i_1=1}^{N/2} \sum_{i_2=0}^{N/2} \text{Re}\{\text{STFT}(\vec{n}, k_1 + i_1, k_2 - i_2) \\ &\quad \quad \text{STFT}^*(\vec{n}, k_1 - i_1, k_2 + i_2)\}, \end{aligned} \quad (3)$$

the 2D WD significantly improves the 2D SPEC concentration (obtained from (3) for $i_1 = i_2 = 0$), reaching the maximum concentration of each signal component separately and resulting in an optimal autoterms' presentation [10, 12–16]. However, based on the full frequency-frequency domain 2D convolution (3) of 2D STFT elements, the 2D WD simultaneously generates emphatic cross-terms in the case of multicomponent signals. Reduced interference S/SFDs that preserve marginal properties suppress this problem. However, as presented in the Introduction, they are defined in the computationally very intensive ways, even in the 1D case [1–5], that seriously limit their applicability. The 2D SM, [12], reduces cross-terms with preservation of the 2D WD autoterms. However, in order to preserve each autoterm separately, it should correspond to the widest signal component. This can be inappropriate for most S/SF points and can negatively influence the cross-terms reduction and calculation complexity, [5, 12], as well as the noise influence suppression, [7].

2.1. New Method for Improved S/SF Representation. To reduce 2D WD cross-terms (or, to completely eliminate them in the case of non-overlapping signal's components), the 2D convolution in (3) must be terminated outside the 2D STFT autoterms' domains, corresponding to regions of support $D_i(\vec{n}, \vec{k})$, $i = 1, \dots, q$ of a q -component signal, Figure 1. However, in order to preserve 2D WD autoterms

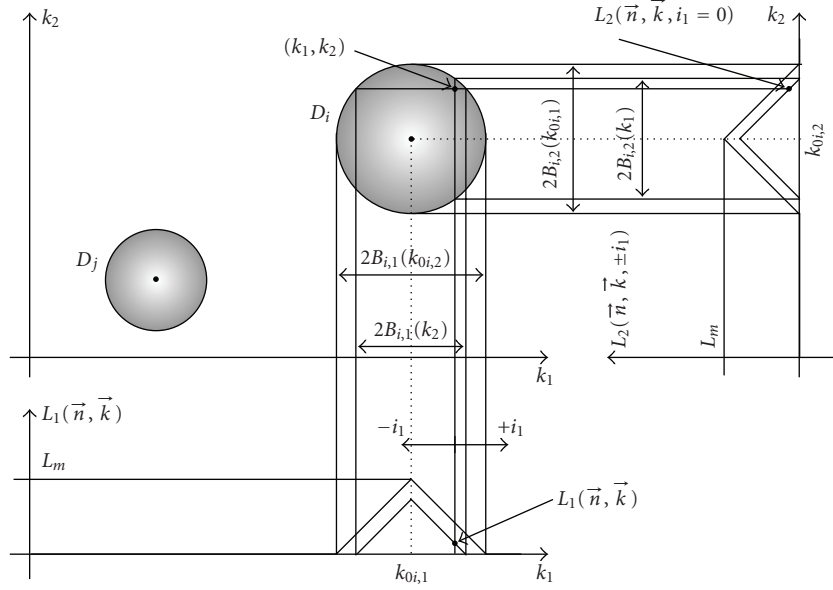


FIGURE 1: Illustration of the proposed 2D method calculation.

presentation, the 2D convolution should be performed inside these regions, including only non-zero summation terms from (3). For these purposes, (3) must be reordered and limited as

$$\begin{aligned}
 & \text{CTFWD}(\vec{n}, \vec{k}) \\
 &= \sum_{i_1=0}^{L_1(\vec{n}, \vec{k})} \left(\text{Re}\{\text{STFT}(\vec{n}, k_1 + i_1, k_2) \text{STFT}^*(\vec{n}, k_1 - i_1, k_2)\} \right. \\
 &\quad + 2 \sum_{i_2=1}^{L_2(\vec{n}, \vec{k}, +i_1)} \text{Re}\{\text{STFT}(\vec{n}, k_1 + i_1, k_2 + i_2) \\
 &\quad \quad \times \text{STFT}^*(\vec{n}, k_1 - i_1, k_2 - i_2)\} \Big) \\
 &+ \sum_{i_1=1}^{L_1(\vec{n}, \vec{k})} \left(\text{Re}\{\text{STFT}(\vec{n}, k_1 + i_1, k_2) \text{STFT}^*(\vec{n}, k_1 - i_1, k_2)\} \right. \\
 &\quad + 2 \sum_{i_2=1}^{L_2(\vec{n}, \vec{k}, -i_1)} \text{Re}\{\text{STFT}(\vec{n}, k_1 + i_1, k_2 - i_2) \\
 &\quad \quad \times \text{STFT}^*(\vec{n}, k_1 - i_1, k_2 + i_2)\} \Big), \tag{4}
 \end{aligned}$$

where $\vec{L}(\vec{n}, \vec{k}) = (L_1(\vec{n}, \vec{k}), L_2(\vec{n}, \vec{k}, \pm i_1)) \leq \vec{L}_m$ is the signal adaptive width of the rectangular convolution 2D window, centered at (k_1, k_2) , $k_1, k_2 = 0, \dots, N-1$ and introduced to limit (3). It takes $\vec{L}(\vec{n}, \vec{k}) = (B_{i,1}(\vec{n}, k_2) - |k_1 - k_{0i,1}(\vec{n})|, B_{i,2}(\vec{n}, k_1 \pm i_1) - |k_2 - k_{0i,2}(\vec{n})|)$ inside the region $D_i(\vec{n}, \vec{k})$ and $\vec{L}(\vec{n}, \vec{k}) = \vec{0}$ elsewhere. $2B_{i,1}(\vec{n}, k_2)$,

$2B_{i,2}(\vec{n}, k_1 \pm i_1)$ denote widths of $D_i(\vec{n}, \vec{k})$ (in k_1 and in k_2) for a given k_2 and a given $k_1 \pm i_1$ ($i_1 = 0, \dots, L_1(\vec{n}, \vec{k})$), respectively, whereas \vec{L}_m is the maximum width of $\vec{L}(\vec{n}, \vec{k})$, determined by the widest $D_i(\vec{n}, \vec{k})$, $i = 1, \dots, q$, and its corresponding local frequency \vec{k}_{0i} , Figure 1. For each point (k_1, k_2) , this means the following.

- (i) The summation in $+i_2$ (limited by $L_2(\vec{n}, \vec{k}, +i_1)$), for each i_1 , $i_1 = 0, \dots, L_1(\vec{n}, \vec{k})$, is performed until $|\text{STFT}(\vec{n}, k_1 \pm i_1, k_2 \pm i_2)|^2 < R^2$, $i_2 = 0, \dots, L_2(\vec{n}, \vec{k}, +i_1)$, is detected, resuming the summation in $-i_2$. (Reference level R^2 determines the support regions D_i , $i = 1, \dots, q$, in the manner that 2D STFTs whose absolute values are below R will be neglected in calculation (4).)
- (ii) The summation in $-i_2$ (limited by $L_2(\vec{n}, \vec{k}, -i_1)$), for each i_1 , $i_1 = 1, \dots, L_1(\vec{n}, \vec{k})$, is performed until $|\text{STFT}(\vec{n}, k_1 \pm i_1, k_2 \mp i_2)|^2 < R^2$, $i_2 = 0, \dots, L_2(\vec{n}, \vec{k}, -i_1)$, is detected, resuming the summation for the next i_1 .
- (iii) The summation in i_1 is performed until $|\text{STFT}(\vec{n}, k_1 \pm i_1, k_2)|^2 < R^2$, $i_1 = 0, \dots, L_1(\vec{n}, \vec{k})$, is detected, corresponding to the detection of the first zero summation term not multiplied by 2 and resulting in completion of the calculation in the observed point (k_1, k_2) .

Note that (4) includes variable number of summation terms—the only necessary ones regarding the total energy of each autoterm separately—in different points (k_1, k_2) , $k_1, k_2 = 0, 1, \dots, N-1$. In this way, (4) is reduced to the 2D SPEC (i.e., to $|\text{STFT}(\vec{n}, \vec{k})|^2$), outside regions $D_i(\vec{n}, \vec{k})$,

$i = 1, \dots, q$ (where $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = 0$) and to the 2D WD inside them, producing the 2D cross-terms-free WD (2D CTFWD) signal representation. In addition, the definition (4) includes, as special cases, the 2D SPEC, 2D WD and 2D SM, that follow, respectively, for $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = 0$, $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = N/2$, $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = L_m$, for all (\vec{n}, \vec{k}) .

2.2. Reference Level R^2 Determination. Here, a reference level R^2 will be determined based on a priori knowledge about the signal's range. It is especially applicable in the cases when the signal is obtained as output of an A/D converter and used in the hardware implementations of S/SF algorithms, considered in the sequel. In this case the signal must be within the *a priori* prescribed range, in order to optimally use the available converter and hardware registers. Then, the R^2 determination is possible on the *a priori* basis as a few percent of the maximum expected 2D SPEC's value.

Note that if the *a priori* knowledge about the signal's range is not reliable, then the reference level can be defined as a few percent of the 2D SPEC's maximum value at the considered time instant \vec{n} , [30, 35]. The reference level can also be calculated based on methods used in digital image processing for similar reasons, [36]. Based on the extensive experimental works, it has been shown that the method is quite insensitive with respect to the reference level R^2 . We have found that the R^2 values within the interval of the 0.1%–10% of the maximum 2D SPEC's value are quite appropriate.

Example. The proposed method has been verified by considering the 2D test signal:

$$\begin{aligned} f_1(n_1, n_2) &= \cos[20\pi(n_1T - 0.75)^2 + 22\pi(n_2T - 0.75)^2] \\ &\quad + 0.5e^{j[-100\cos(\pi n_1T/2) + 100\cos(\pi n_2T/2)]}, \\ f_2(n_1, n_2) &= \cos\{1000\pi[(n_1T + 0.5)^2 + (n_2T - 0.5)^2]\}. \end{aligned} \quad (5)$$

This signal consists of dual components: infinite duration $f_1(n_1, n_2)$, considered in the range $|n_1T| < 0.75$, $|n_2T| < 0.75$, and $f_2(n_1, n_2)$ having comparatively small domain of $|n_1T + n_2T| < 0.1$, $|n_2T - n_1T - 1| < 0.1$ (2D Fourier transform of $f_2(n_1, n_2)$ is usually treated when one analyses this component separately). Therefore, signal (5) represents a very interesting, commonly considered 2D test signal, [12, 34]. A sampling interval of $T = 1/64$, the Hanning lag-window $w(\vec{m})$, $N = 64$, maximum convolution window width of $L_m = 18$ and the reference level R^2 of the 1% of the SPEC's maximum value are used in simulations. Simulation results, computed at the point $(n_1T, n_2T) = (-0.25, -0.25)$, are presented in Figure 2. Low 2D SPEC's concentration, high 2D WD resolution, but with the emphatic cross-terms presence, high quality 2D SM

autoterms' representation with the only partially suppressed cross-terms, as well as the improved S/SF representation, based on the proposed method, can be noticed from Figures 2(a)–2(d). Figure 2(f) readily proves the pure CTFWD signal representation achieved by the proposed method.

2.3. Noisy Signal Analysis. In practice, signals are always exposed to the additive noise influence. To evaluate noise influence to the proposed method, let us consider the noisy signal $x(\vec{n}) = f(\vec{n}) + \varepsilon(\vec{n})$, where $\varepsilon(\vec{n})$ represents the additive white Gaussian complex noise with variance σ_ε^2 . The variance of the real-valued 2D CTFWD estimator (4) is defined by

$$\begin{aligned} \sigma_{xx}^2(\vec{n}, \vec{k}) &= \text{var}[\text{CTFWD}_x(\vec{n}, \vec{k})] \\ &= E\{\text{CTFWD}_x^2(\vec{n}, \vec{k})\} - (E\{\text{CTFWD}_x(\vec{n}, \vec{k})\})^2. \end{aligned} \quad (6)$$

It consists of two parts: the signal-and-noise-dependent one, $\sigma_{f\varepsilon}^2(\vec{n}, \vec{k})$, existing only inside regions $D_i(\vec{n}, \vec{k})$, $i = 1, \dots, q$, and the noise-only-dependent one, $\sigma_{\varepsilon\varepsilon}^2(\vec{n}, \vec{k})$ existing everywhere (in all frequency-frequency points), $\sigma_{xx}^2(\vec{n}, \vec{k}) = \sigma_{f\varepsilon}^2(\vec{n}, \vec{k}) + \sigma_{\varepsilon\varepsilon}^2(\vec{n}, \vec{k})$. Following the procedure from the 1D noisy signal case, [7, 37, 38], after several straightforward transformations, in the case of rectangular lag-window $w(\vec{m})$, the variance (6) can be derived as

$$\begin{aligned} \sigma_{xx}^2(\vec{n}, \vec{k}) &= \begin{cases} 2N\sigma_\varepsilon^2 \sum_{\vec{i} = -\vec{L}(\vec{n}, \vec{k})}^{\vec{L}(\vec{n}, \vec{k})} \text{SPEC}_f(\vec{n}, \vec{k} + \vec{i}) + \sigma_{\varepsilon\varepsilon}^2(\vec{n}, \vec{k}), & \text{for } (\vec{n}, \vec{k}) \in D_i(\vec{n}, \vec{k}), i = 1, \dots, q, \\ N^2\sigma_\varepsilon^4, & \text{for } (\vec{n}, \vec{k}) \notin D_i(\vec{n}, \vec{k}), i = 1, \dots, q, \end{cases} \end{aligned} \quad (7)$$

where $\sigma_{\varepsilon\varepsilon}^2(\vec{n}, \vec{k}) = N^2\sigma_\varepsilon^4[\sum_{i_1=1}^{L_1(\vec{n}, \vec{k})}(L_2(\vec{n}, \vec{k}, +i_1) + L_2(\vec{n}, \vec{k}, -i_1)) + L_2(\vec{n}, \vec{k}, 0) + 1]$ and $\sigma_{\varepsilon\varepsilon}^2(\vec{n}, \vec{k}) = N^2\sigma_\varepsilon^4$ outside regions of support $D_i(\vec{n}, \vec{k})$, $i = 1, \dots, q$ (when $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = 0$).

Note that by performing noisy signal analysis of the proposed method, we have also unified the noisy signal analysis of its special cases (2D SPEC for $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = 0$, 2D WD for $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = N/2$, and 2D SM for $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = L_m$, for all (\vec{n}, \vec{k})). Based on this observation, the following conclusions can readily be derived.

- (1) The proposed method minimizes the estimator's variance $\sigma_{xx}^2(\vec{n}, \vec{k})$ outside regions $D_i(\vec{n}, \vec{k})$, $i = 1, \dots, q$. Knowing that it also produces the optimal autoterms presentation, the proposed method optimizes the peak signal-to-noise ratio in (k_1, k_2) points existing outside regions of support. (The peak signal-to-noise ratio assumes the ratio of the squared peak value of the S/SFD and estimator's variance $\sigma_{xx}^2(\vec{n}, \vec{k})$. This is required in many practical applications where the peak values of S/SFD are used to estimate local frequency of an analyzed signal. In this case, we are not interested in the local signal-to-noise ratio, especially in the frequency-frequency points existing outside regions of support (where S/SFDs are equal to zero). In these points, the peak signal-to-noise ratio represents the measure of possible false peak detection (wrong local frequency estimation, [7]).)
- (2) The estimator's variance (7) increases inside regions $D_i(\vec{n}, \vec{k})$, $i = 1, \dots, q$. However, by taking optimal number of summation terms in (4), the proposed method decreases $\sigma_{xx}^2(\vec{n}, \vec{k})$ and, therefore, improves the peak signal-to-noise ratio regarding the 2D WD and 2D SM in the frequency-frequency points existing inside these regions.
- (3) The 2D SPEC minimizes $\sigma_{xx}^2(\vec{n}, \vec{k})$ in the (k_1, k_2) points existing inside regions $D_i(\vec{n}, \vec{k})$, $i = 1, \dots, q$. However, at the same time it produces low signal's concentration around local frequencies. Therefore, the proposed method can improve 2D SPEC's estimation for the case of highly nonstationary 2D signals, even in the regions of support points (k_1, k_2) .

These conclusions have been tested numerically. To this end, signals (5) are exposed to the additive white Gaussian noise with high variance $\sigma_e^2 = 1$. The same parameters as in the noiseless case are used here. The considered S/SF representations of the noisy signal are given in Figure 3. They readily prove theoretically derived conclusions. Note that, depending on the noise distribution and the R^2 selection, there can exist particular frequency-frequency points outside regions of support in which $|\text{STFT}(\vec{n}, k_1 \pm i_1, k_2 \pm i_2)|^2 \geq R^2$ and/or $|\text{STFT}(\vec{n}, k_1 \pm i_1, k_2 \mp i_2)|^2 \geq R^2$, $i_1 = 0, \dots, L_1(\vec{n}, \vec{k})$, $i_2 = 0, \dots, L_2(\vec{n}, \vec{k}, -i_1)$, are satisfied. This implies the nonzero values of $L_1(\vec{n}, \vec{k})$ and/or $L_2(\vec{n}, \vec{k}, -i_1)$ in these points. However, it does not significantly influence the S/SF representation based on the proposed method, Figures 3(d) and 3(e). In line with this conclusion, note that greater values of the reference level R^2 (about 5%–10% of the expected 2D SPEC's maximum value) almost remove these effects, Figure 3(e).

2.4. Calculation Complexity of the Proposed Method and Comparisons. In this subsection, the proposed method will be compared, regarding the calculation complexity, with the conventional S/SFDs (2D SPEC and 2D WD) and

TABLE 1: Numbers of complex operations by frequency-frequency point required by the considered S/SFDs. (1) 2D WD (2) calculated using the FFT routines, (2) 2D SM using the recursive 2D STFT calculation (with 2 complex summations and a complex multiplication, [12]), (3) Proposed method with the 2D STFT calculation based on the FFT routines, (4) Proposed method using the recursive 2D STFT calculation, where $\bar{L}_2(\vec{n}, \vec{k}) = \sum_{i_1=1}^{L_1(\vec{n}, \vec{k})} (L_2(\vec{n}, \vec{k}, +i_1) + L_2(\vec{n}, \vec{k}, -i_1))/2$. Multiplications by 2 are not considered, because the time needed for their execution is much smaller than the time needed for other operations.

S/SFD	Additions	Multiplications
1	$4(\log_2 N + 1)$	$2(\log_2 N + 2)$
2	$2 + L_m^2 + L_m$ $2\log_2 N + \bar{L}_2 +$	$L_m^2 + L_m + 3/2$ $\log_2 N + \bar{L}_2 +$
3	$L_1(\vec{n}, \vec{k}) +$ $L_2(\vec{n}, \vec{k}, 0)/2$	$L_1(\vec{n}, \vec{k}) +$ $(L_2(\vec{n}, \vec{k}, 0) + 1)/2$
4	$2 + \bar{L}_2 + L_1(\vec{n}, \vec{k}) +$ $L_2(\vec{n}, \vec{k}, 0)/2$	$\bar{L}_2 + L_1(\vec{n}, \vec{k}) +$ $L_2(\vec{n}, \vec{k}, 0)/2 + 3/2$

computationally the simplest reduced interference method (2D SM, see the Introduction). The proposed method based S/SF representation takes variable number of necessary operations (regarding 2D WD autoterms quality) in different frequency-frequency points (k_1, k_2) , $k_1, k_2 = 0, 1, \dots, N - 1$: the minimal one outside the regions of support (where a great part of these points commonly lie), the higher one inside these regions, and the possible maximum one only in the central points of the widest region (see Figure 8 and the discussion from Section 5). In this way, the method improves the calculation complexity of the considered S/SFDs for $\text{mean}_{k_1, k_2=0, 1, \dots, N-1} \{\bar{L}_2(\vec{n}, \vec{k}) + L_1(\vec{n}, \vec{k}) + L_2(\vec{n}, \vec{k}, 0)/2\} < 2\log_2 N + 5/2$, Table 1. For example, in the analyzed signal (5) case and for $L_m = 18$, $N = 64$, average number of complex operations per frequency-frequency point, required by the proposed method, are $30842/N^2 = 7.5$ additions and $28794/N^2 = 7$ multiplications. Its complexity slightly differ from the SPEC case (obtained from the proposed method for $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = 0$), but also significantly improve calculation complexity of the 2D WD, with 28 complex additions and 16 complex multiplications per frequency-frequency point, and the 2D SM, with 344, 343.5 corresponding operations, Table 1. In addition, among the considered S/SFDs, only the proposed one produces a pure 2D CTFWD signal representation in the practically only important case of multicomponent signals having different autoterms widths, Figure 2. In line with these conclusions, the 2D S/SFD having the same calculation complexity as the proposed method is represented in Figure 2(e). However, it produces only a low concentrated 2D signal representation, very close to the 2D SPEC one.

Although the proposed method significantly improves calculation complexity of the 2D SM and 2D WD, note that it is also quite numerically intensive, Table 2. Generally, all S/SFDs, beyond the 2D STFT, are numerically quite complex and require significant execution time. This fact makes these S/SFDs unsuitable for real time analysis and

seriously restrict their applicability. However, hardware implementations (such as the one developed here), when possible, can overcome this drawback, enabling application of these S/SFDs in numerous additional problems in practice.

3. Signal Adaptive Hardware Implementation Approach

2D CTFWD definition (4) adapted for real time implementation should include only real multiplications. To this end, we express it as a sum of two computational lines, $\text{CTFWD}(\vec{n}, \vec{k}) = \text{CTFWD}_R(\vec{n}, \vec{k}) + \text{CTFWD}_I(\vec{n}, \vec{k})$, used respectively for processing the real and imaginary parts of 2D STFTs (from already available 2D STFT or 2D FFT modules, [20–24, 31]). According to (4), these lines have identical forms. $\text{CTFWD}_R(\vec{n}, \vec{k})$ is

$$\begin{aligned} & \text{CTFWD}_R(\vec{n}, \vec{k}) \\ &= \sum_{i_1=0}^{L_1(\vec{n}, \vec{k})} \left(\text{STFT}_{\text{Re}}(\vec{n}, k_1 + i_1, k_2) \text{STFT}_{\text{Re}}(\vec{n}, k_1 - i_1, k_2) \right. \\ & \quad + 2 \sum_{i_2=1}^{L_2(\vec{n}, \vec{k}, +i_1)} \text{STFT}_{\text{Re}}(\vec{n}, k_1 + i_1, k_2 + i_2) \\ & \quad \quad \times \text{STFT}_{\text{Re}}(\vec{n}, k_1 - i_1, k_2 - i_2) \Big) \\ &+ \sum_{i_1=1}^{L_1(\vec{n}, \vec{k})} \left(\text{STFT}_{\text{Re}}(\vec{n}, k_1 + i_1, k_2) \text{STFT}_{\text{Re}}(\vec{n}, k_1 - i_1, k_2) \right. \\ & \quad + 2 \sum_{i_2=1}^{L_2(\vec{n}, \vec{k}, -i_1)} \text{STFT}_{\text{Re}}(\vec{n}, k_1 + i_1, k_2 - i_2) \\ & \quad \quad \times \text{STFT}_{\text{Re}}(\vec{n}, k_1 - i_1, k_2 + i_2) \Big) \end{aligned} \quad (8)$$

($\text{STFT}(\vec{n}, \vec{k}) = \text{STFT}_{\text{Re}}(\vec{n}, \vec{k}) + j\text{STFT}_{\text{Im}}(\vec{n}, \vec{k})$), whereas $\text{CTFWD}_I(\vec{n}, \vec{k})$ is obtained by replacing the real parts of the 2D STFTs with their imaginary parts in (8). Note that, in different points (k_1, k_2) , $k_1, k_2 = 0, 1, \dots, N-1$, the summation in (8) involve variable number of $\text{CN}(\vec{n}, \vec{k}) = \sum_{i_1=1}^{L_1(\vec{n}, \vec{k})} (L_2(\vec{n}, \vec{k}, +i_1) + L_2(\vec{n}, \vec{k}, -i_1) + 2) + L_2(\vec{n}, \vec{k}, 0) + 1$ terms. They are obtained by multiplying 2D STFT elements, symmetrically distributed around the observed point (k_1, k_2) in the 2D frequency-frequency plane.

The 2D CTFWD hardware implementation will be presented through the implementation of its real computational line and the control logic, Figures 4 and 5, since the imaginary computational line is identical with the real one, whereas the control units and configuration signals are unique. The design principle follows the developed form of (8), where each summation term is executed during

TABLE 2: Total numbers of complex additions and complex multiplications, required by the considered S/SFDs, for different N . (1) 2D SPEC (the proposed method with $L_1(\vec{n}, \vec{k}) = L_2(\vec{n}, \vec{k}, \pm i_1) = 0$), (2) 2D WD calculated using the FFT routines, (3) 2D SM using the recursive 2D STFT calculation, (4) Proposed method using the recursive 2D STFT calculation.

N		1	2	3	4
64	Additions	8192	114688	1409024	30842
	Multiplications	6144	65536	1406976	28794
128	Additions	32768	524288	5636096	86477
	Multiplications	24576	294912	5627904	78285
256	Additions	131072	2359296	22544384	230630
	Multiplications	98304	1310720	22511616	197860
512	Additions	524288	10485760	90177536	578450
	Multiplications	393216	5767168	90046464	447380

the corresponding step which takes one CLK cycle. In this way, we are able to balance the amount of work done in each CLK, resulting in minimization of the CLK cycle time. During the first CLK, when $\vec{L}(\vec{n}, \vec{k}) = \vec{0}$, the 2D SPEC is calculated from the 2D STFT element, $\text{STFT}(\vec{n}, \vec{k})$, situated in the central point of the convolution window, Figure 6(a). Residual summation terms, obtained for the increased indexes i_1 and/or i_2 in subsequent CLKs (second, third, ...) and existing only in regions of support frequency-frequency points, are the conditional ones. They are used to improve the S/SFD concentration with the goal to achieve the 2D WD one. Therefore, the 2D CTFWD real time implementation requires variable number of CLKs by frequency-frequency point (i.e., by one convolution window position) to be executed, where only 2D SPEC execution CLK (the first one) remains the unconditional one in each point (k_1, k_2) , $k_1, k_2 = 0, 1, \dots, N-1$.

In general, each 2D CTFWD element is produced by sliding 2D convolution window over the 2D STFT input elements and by computing the 2D CTFWD output value according to the input elements and the algorithm (8). The obtained result is a 2D CTFWD element assigned to the center of the 2D convolution window, Figure 6(a). Following these observations, the architecture for real time design of the 2D CTFWD real computational line consists of several main functional units, Figure 4. The STFT-to-CTFWD gateway, Figure 5, represents a functional kernel of the proposed architecture. It is used to produce 2D CTFWD outputs based on the 2D STFT input elements obtained from the convolution window register block. The $(2L_m + 1) \times (2L_m + 1)$ convolution window register block and $2L_m$ first-in-first-out (FIFO) delays mutually implement the 2D convolution window function, presented in detail in Figure 6. The convolution window register block determines the address order of the 2D STFT input elements for which the corresponding 2D CTFWD output will be computed according to (8). FIFO delays provide sliding of the 2D convolution window over 2D STFT input elements, as presented in Figure 6.

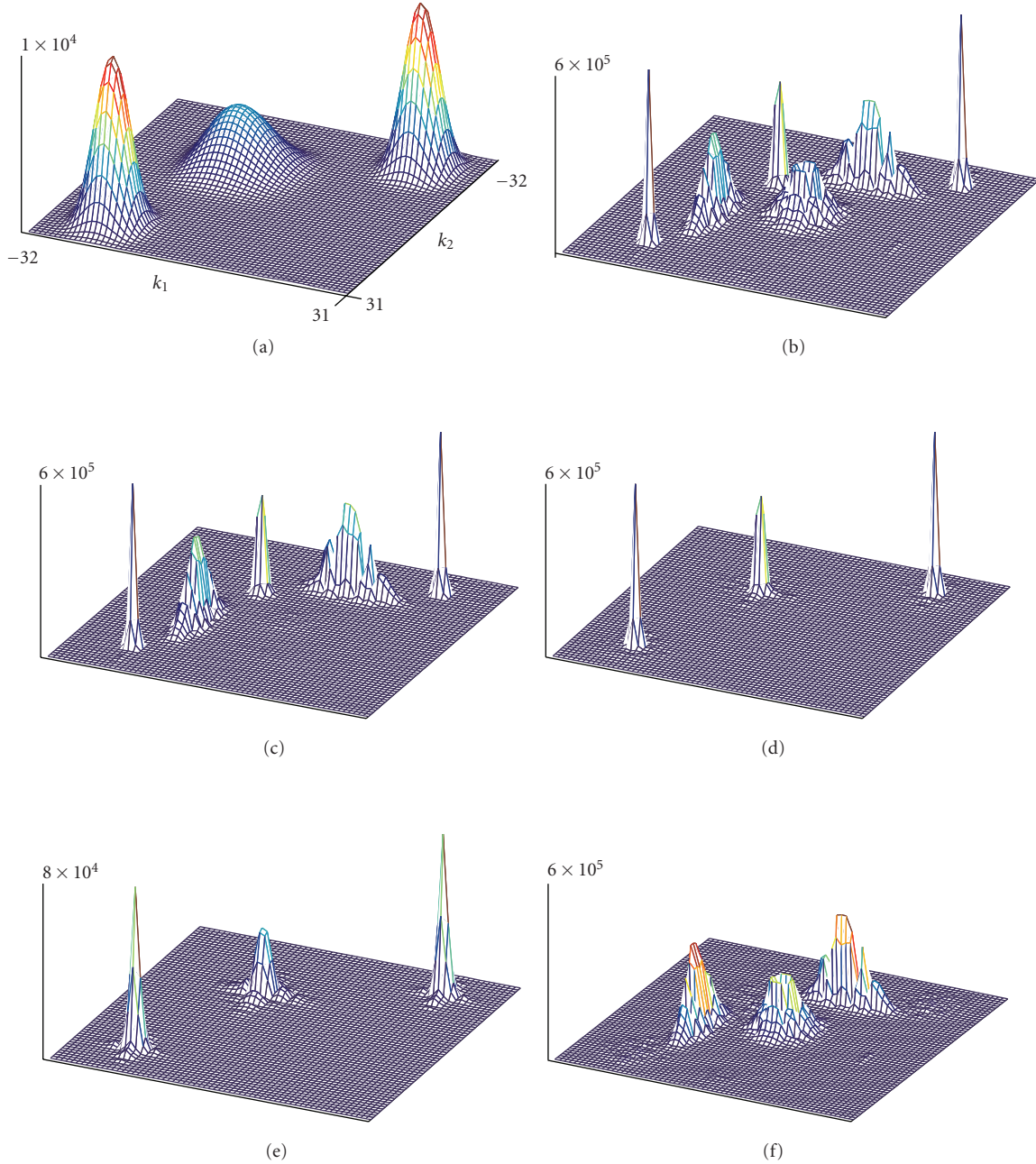


FIGURE 2: S/SF representations of signal (5): (a) 2D SPEC, (b) 2D WD, (c) 2D SM corresponding to the widest signal component, (d) Proposed method based representation, (e) S/SF representation having the same calculation complexity as the proposed method, (f) Difference between graphics (b), (d).

The operation principle can be described as follows.

(1) The STFT_IN elements are imported to the input memory owing to each double CLK cycle, corresponding to the minimal execution time required by a frequency-frequency point (2D SPEC execution time). This period simultaneously determines sampling rate of the analyzed analogue 2D signal. By each *STFT_Load* cycle, an STFT_IN element is moved from the input memory to the convolution window area that is sliced over input 2D STFTs for one position right, as shown in Figure 6.

(2) According to the actual convolution window position and the algorithm (8), the corresponding 2D CTFWD output element is calculated in $CN(\vec{n}, \vec{k}) + 1$ CLKs. In the i th CLK ($i = 1, \dots, CN(\vec{n}, \vec{k})$), the corresponding summation term from (8) (the first input of the cumulative pipelined adder CumADD) is produced, generating the CTFWD(\vec{n}, \vec{k}) element at the CumADD output after $CN(\vec{n}, \vec{k})$ CLKs. The $(CN(\vec{n}, \vec{k}) + 1)$ st CLK is the completion one. Therefore, the period of the *STFT_Load* cycle must be variable

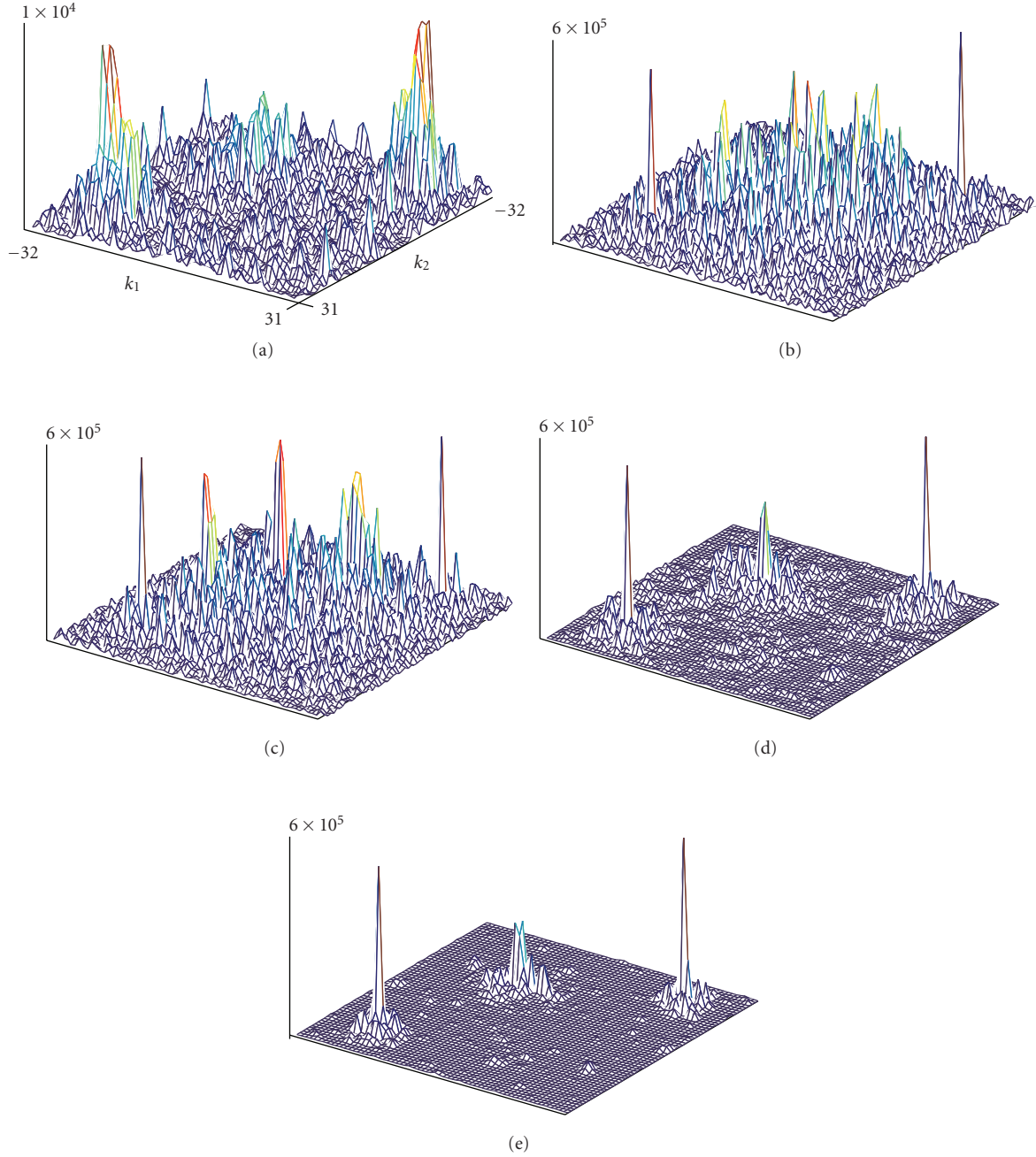


FIGURE 3: S/SFDs of noisy signal (5): (a) 2D SPEC, (b) 2D WD, (c) 2D SM corresponding to the widest signal component, (d), (e) proposed method based S/SFDs corresponding to R^2 of 1% and 5% of the expected 2D SPEC's maximum value.

(in different frequency-frequency points) and $CN(\vec{n}, \vec{k}) + 1$ times greater than the CLK period.

(3) In the corresponding CLKs, signals $x_{\pm i_1, \pm i_2}$ and $x_{\pm i_1, \mp i_2}$ are generated as: $x_{\pm i_1, \pm i_2} = 1$ if $|\text{STFT}(\vec{n}, k_1 \pm i_1, k_2 \pm i_2)|^2 > R^2$ and $x_{\pm i_1, \pm i_2} = 0$ otherwise, that is, $x_{\pm i_1, \mp i_2} = 1$ if $|\text{STFT}(\vec{n}, k_1 \pm i_1, k_2 \mp i_2)|^2 > R^2$ and $x_{\pm i_1, \mp i_2} = 0$ otherwise. They, respectively, determine nonzero values of $\text{STFT}(\vec{n}, k_1 \pm i_1, k_2 \pm i_2)$ and of $\text{STFT}(\vec{n}, k_1 \pm i_1, k_2 \mp i_2)$, $i_1 = 0, 1, \dots, L_1(\vec{n}, \vec{k})$, $i_2 = 0, 1, \dots,$

$L_2(\vec{n}, \vec{k}, \pm i_1)$ and simultaneously produce the *RegionSup* control signal, $\text{RegionSup} = x_{i_1, \pm i_2} \cdot x_{-i_1, \mp i_2}$. Zero value of the *RegionSup* signal implies following actions: (1) Through the participation in the *CumADD_CLK* signal generation, disables the corresponding term, non-existing inside the regions of support domains, to enter the summation (8), and (2) Through the participation in the *RESET_L* and the *High_Count_CLK* signals generation, terminates the summation in $+i_2$ and in $-i_2$ (for each i_1 , $i_1 = 0, 1, \dots, L_1(\vec{n}, \vec{k})$), resuming the summation in

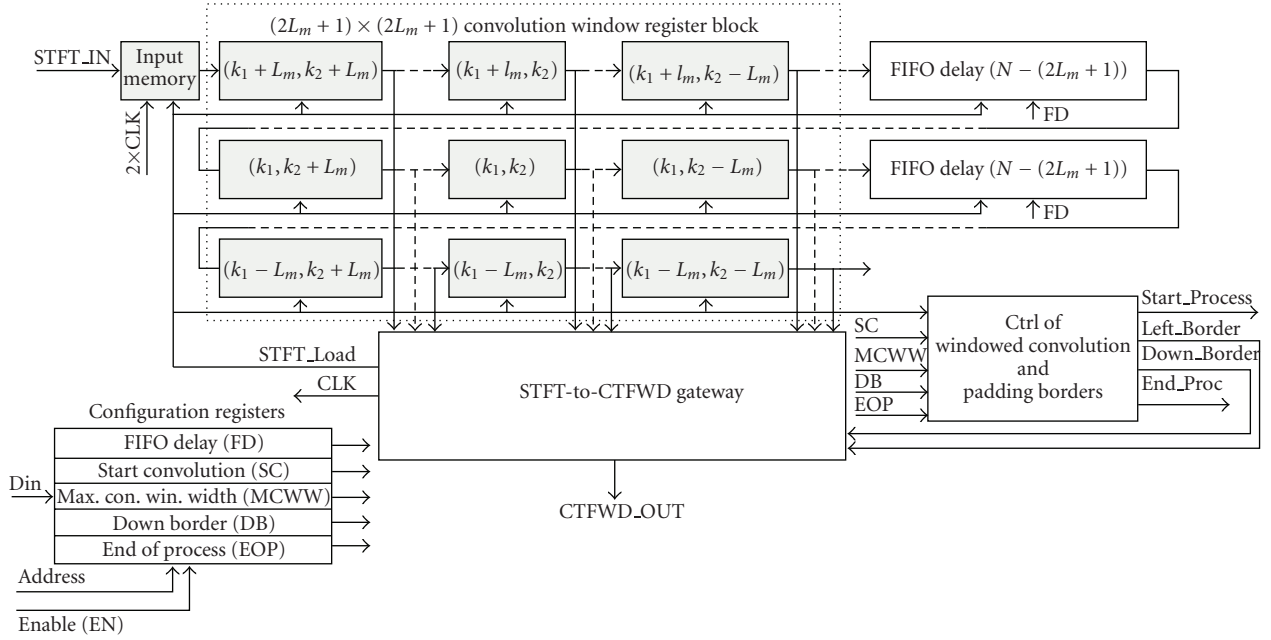


FIGURE 4: Proposed signal adaptive hardware design. In registers, the S/SF position of the stored 2D STFT element is denoted.

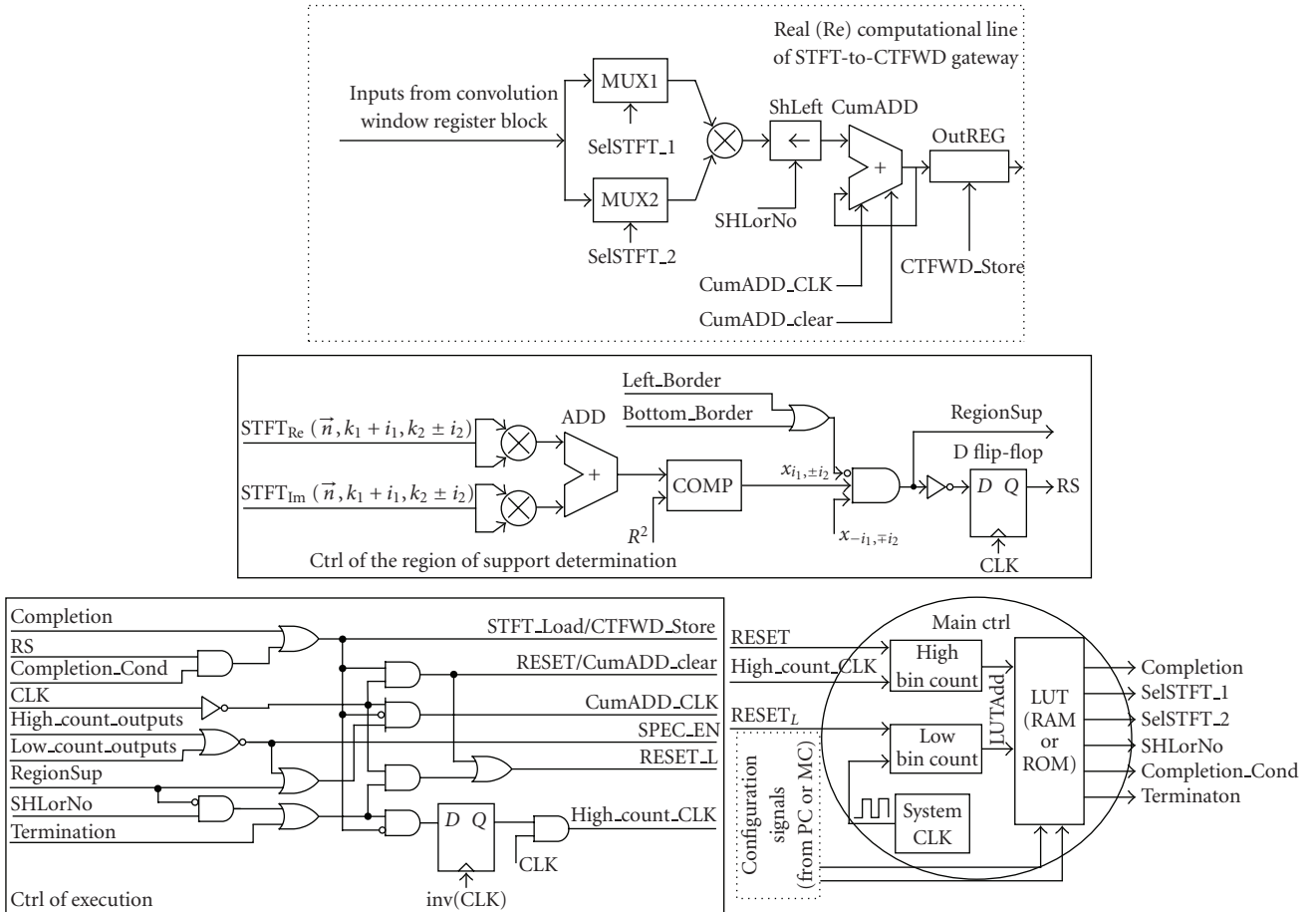


FIGURE 5: STFT-to-CTFWD gateway architecture with the necessary control logic for the signal adaptive real time design.

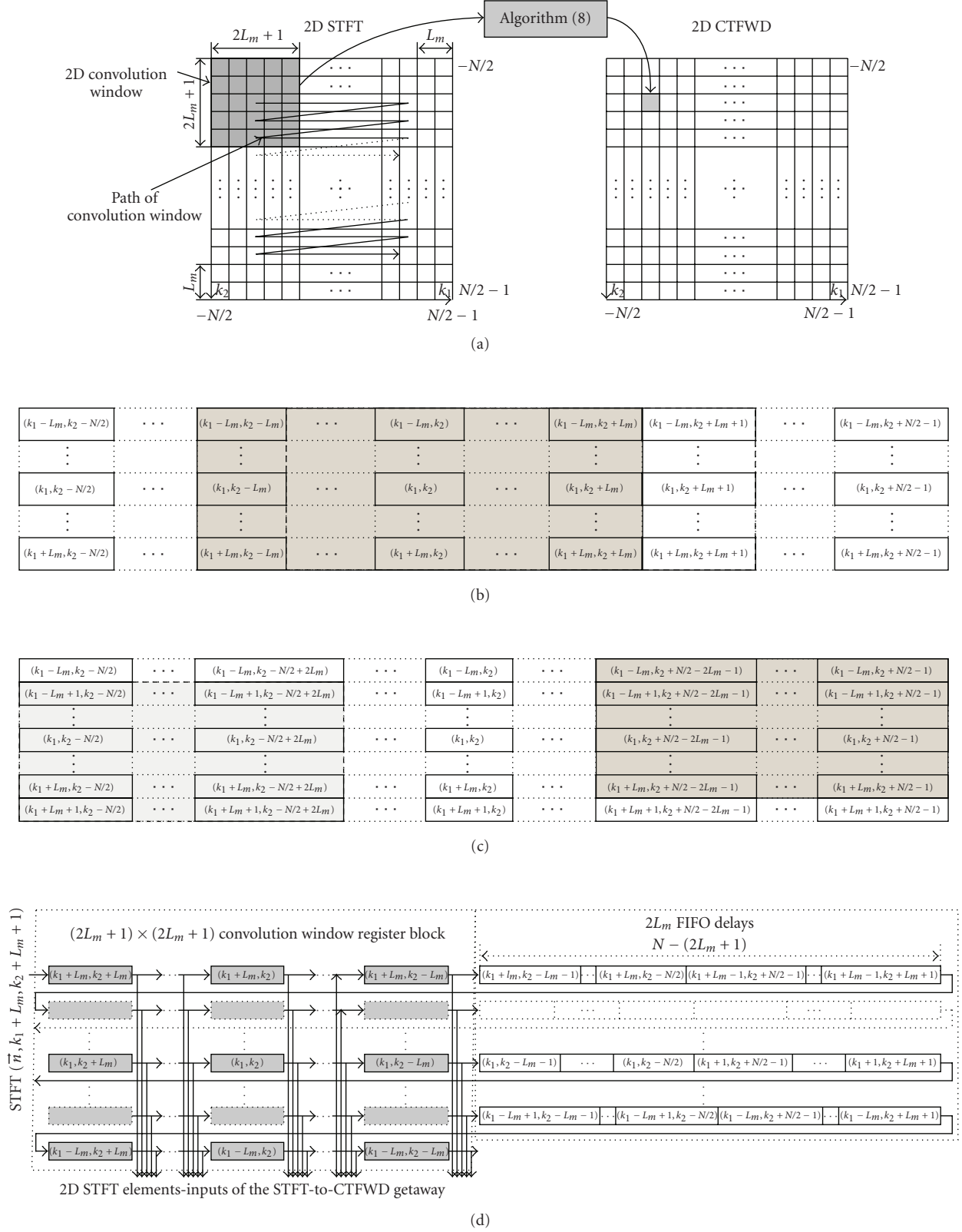


FIGURE 6: (a) Procedure of 2D convolution window sliding and of 2D CTFWD producing; (b) Actual convolution window position corresponding to the point (k_1, k_2) (thick solid line) and its next position (thick dashed line); (c) Actual convolution window position (thick solid line) and its next-left border-position (thick dashed line); (d) Convolution window function, implemented in real time. Cells correspond to the 2D STFT elements, denoted by their position in frequency-frequency plane.

$-i_2$ and for the next i_1 , respectively. Further, zero value of the *RegionSup* signal, reached in the CLK cycle when *SHLorNo*=0 (corresponding to the first zero summation term from (8) not multiplied by 2), terminates the summation (8) in i_1 , resulting in the calculation completion (in the next CLK) for the observed point (k_1, k_2) . In that line, the RS signal, $RS = \text{inv}(\text{RegionSup})$, participates in the *STFT_Load/CTFWD_Store*, *CumADD_clear/RESET* signals generation. With a latency of half of a CLK, the system is reset and the execution for the next frequency-frequency point begins. In this way, the *RegionSup* signal allows the proposed design to optimize the number of CLKs taken in different frequency-frequency points within the execution. The *SPEC_EN* signal provides execution of the unconditional (2D SPEC execution) cycle, even if $x_{0,0} = 0$.

A look-up table memory (LUT), Table 3, manages the execution. Its locations consist of the 4-bit control signals area and MUX addresses. Functions of generated control signals are given in Table 4. Binary counter *Low_Bin_Count* generates LUT's low addresses, controlling the summation (8) in $+i_2$ and in $-i_2$. Binary counter *High_Bin_Count* sets LUT's high addresses, controlling the summation (8) in i_1 . Operations at the bordering positions, as well as the whole S/SF process, are managed by the *Start_Process*, *Left_Border*, *Bottom_Border* and *End_Process* signals, Table 4. These signals are generated by considering input parameters from the Configuration registers, Table 5, as well as the synchronization conditions related to the CLK and *STFT_Load* cycles. They are produced in the modules that consist of variable length up-down binary counters and binary magnitude comparators whose binary references are parameters from the Configuration registers.

The longest path that determines the fastest CLK cycle time corresponds to the generation of the *RegionSup* signal in a half of a cycle, through a multiplier, an adder and a comparator ($T_c/2 = T_m + T_a + T_{\text{comp}}$, where T_c , T_m , T_a , T_{comp} are CLK cycle, multiplication, addition, and comparison times, resp.). In this way, we enable participation of the *RegionSup* signal in the *CumADD_CLK* signal generation in the second half of the same CLK (see timing diagram from Figure 8). Maximum output register lengths for each used digital unit are given in Table 6. They are derived as functions of the 2D STFT data length (l), maximum convolution window width (L_m) and the number of frequency points (N). Note that critical point is the width of the *CumADD/OutREG*. The longest path depends on the 2D STFT data length only.

4. Testing and Verification of the Proposed Design

The proposed system for S/SF analysis has been verified with an FPGA device real time design. The FPGA implementation approach is chosen in line with the fact that recently high performance devices for solving practical problems in signal processing tend to be implemented in an FPGA, instead of in a DSP, chip. This is possible because the gate densities available in FPGAs now allow fairly sophisticated DSP algorithms to be implemented within a single chip, [39, 40].

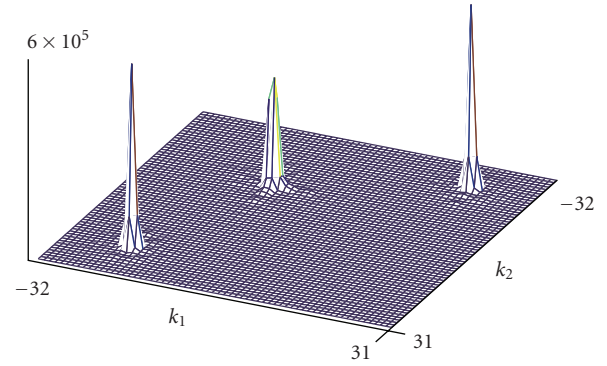


FIGURE 7: S/SF representation of the analyzed signal (5) in $(n_1T, n_2T) = (-0.25, -0.25)$, obtained by using the proposed hardware design with $L_m = 5$, implemented in FPGA device.

Additionally, languages used for the hardware description of FPGA chips provide a high level behavioral design methodology, requiring high flexibility from the targeted device, so that the synthesis tool will be able to produce both efficient device utilization and high performance. Also, usage of recent FPGA chips with huge internal memory blocks is considered to be a powerful tool for implementation of convolution and delay operators that are of essential importance in implementation of DSP algorithms.

Hardware realization of the proposed system with $L_m = 5$ is performed by using the EP2S15F672C5 device from the Stratix II family. Rates of utilization of device's silicon resources are given in Table 7. Before programming the selected FPGA device, the compilation and simulation have been performed by processing the test signal (5). The 2D STFT elements (their real and imaginary parts), numerically computed in Section 2, normalized to the range $[0 \ 255]$, and rounded to the 8-bit integers, are imported to the designed system input. Results of the real time implementation are presented in Figure 7. Accuracy of these results can easily be checked by comparison with the numerical results given in Figure 2(d). Also, the obtained results can be readily proven by analyzing simulation results given in Figure 8. In line with the simulation results analysis, note that the developed signal adaptive system with $L_m = 5$ requires convolution window register blocks (for storing the real and imaginary parts of the input 2D STFT elements) of $(2L_m + 1) \times (2L_m + 1) = 121$ parallel-in-parallel-out registers. However, presentation of the content of all these registers in a single figure is impossible. Therefore, in Figure 8 we present contents of the central registers and their neighborhood registers, named in line with their relative position with respect to the central ones. For example, in the first marked instant, corresponding to the outside regions of support frequency-frequency point, $\text{CTFWD_OUT} = 2\text{D SPEC} = (-3)^2 + 9^2 = 90$ has been derived from (8) in two CLKs, since $x_{0,0} = 0$ ($3^2 + 9^2 = 90 < R^2 = 94$). Note that multiplication and shift operations are parallel, while adding has a latency of half of a CLK. In the second marked instant, corresponding to the marginal frequency-frequency point from the detected region of support, $\text{CTFWD_OUT} = 1^2 + (-11)^2 = 122$ has

TABLE 3: Content of LUT memory locations for given L_m . $ADD_{M,M}$ denotes address of central element of the convolution window register block, symbol \ll denotes shift left logical operation and $r = \text{Length}(SelSTFT_1)$. Control signals area contains following bits: (1) $SHLorNo$, (2) $Completion_Cond$, (3) $Termination$, (4) $Completion$.

Address		Ctrl Signals Area				$SelSTFT_1$	$SelSTFT_2$
High	Low	1	2	3	4		
0	0	0	0	0	0	$ADD_{M,M} \ll r$	$ADD_{M,M}$
0	1	1	1	0	0	$ADD_{M,M+1} \ll r$	$ADD_{M,M-1}$
\vdots	\vdots	1	0	0	0	\vdots	\vdots
0	L_m	1	0	1	0	$ADD_{M,M+L_m} \ll r$	$ADD_{M,M-L_m}$
1	0	0	0	0	0	$ADD_{M+1,M} \ll r$	$ADD_{M-1,M}$
1	1	1	1	0	0	$ADD_{M+1,M+1} \ll r$	$ADD_{M-1,M-1}$
\vdots	\vdots	1	0	0	0	\vdots	\vdots
1	L_m	1	0	1	0	$ADD_{M+1,M+L_m} \ll r$	$ADD_{M-1,M-L_m}$
2	0	0	0	0	0	$ADD_{M+1,M} \ll r$	$ADD_{M-1,M}$
2	1	1	1	0	0	$ADD_{M+1,M-1} \ll r$	$ADD_{M-1,M+1}$
\vdots	\vdots	1	0	0	0	\vdots	\vdots
2	L_m	1	0	1	0	$ADD_{M+1,M-L_m} \ll r$	$ADD_{M-1,M+L_m}$
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
$2L_m - 1$	0	0	0	0	0	$ADD_{M+L_m,M} \ll r$	$ADD_{M-L_m,M}$
$2L_m - 1$	1	1	1	0	0	$ADD_{M+L_m,M+1} \ll r$	$ADD_{M-L_m,M-1}$
\vdots	\vdots	1	0	0	0	\vdots	\vdots
$2L_m - 1$	L_m	1	0	1	0	$ADD_{M+L_m,M+L_m} \ll r$	$ADD_{M-L_m,M-L_m}$
$2L_m$	0	0	0	0	0	$ADD_{M+L_m,M} \ll r$	$ADD_{M-L_m,M}$
$2L_m$	1	1	1	0	0	$ADD_{M+L_m,M-1} \ll r$	$ADD_{M-L_m,M+1}$
\vdots	\vdots	1	0	0	0	\vdots	\vdots
$2L_m$	L_m	1	0	1	0	$ADD_{M+L_m,M-L_m} \ll r$	$ADD_{M-L_m,M+L_m}$
$2L_m + 1$	0	0	0	0	1	0	0

TABLE 4: Function of control signals generated by the Main Ctrl and the ctrl of windowed convolution and padding borders.

Control signal	Effect
$SelSTFT_1,2$	Enable sharing of the STFT-to-CTFWD gateway for different 2D STFT inputs in different CLKs within the execution in the observed frequency-frequency point
$SHLorNo$	Provides multiplication by 2 of the partial product term according to (8)
$Completion_Cond$	Allows the RS signal to produce the completion CLK from the conditional one
$Termination\&Completion$	Provide termination of the summation (8) in $+i_2$ and in $-i_2$ and its completion, respectively, in frequency-frequency points in which the $RegionSup$ signal cannot achieve zero value and, therefore, cannot assume the role described in the operation principle 3. (In practical implementations, the relatively small L_m values, $5 \leq L_m \leq 7$, are usually applied. They provide the desired—2D WD—concentration (compare the representation from Figure 2(d), obtained for $L_m = 18$, with the representation from Figure 6, obtained for $L_m = 5$), but also significantly simplify hardware implementation (see Table 8). However, a predefined maximum convolution window width corresponding to these L_m values can be smaller than the theoretically required one in the points (k_1, k_2) existing around the local frequency. Therefore, in these points, the $RegionSup$ signal cannot achieve zero value, needed for the termination of the summation (8) in $+i_2$ and in $-i_2$ and its completion, according to the principle 3 of the hardware operation.)
$Left_Border\&Bottom_Border$	Through the participation in the $RegionSup$ signal generation, allow padding the left and bottom borders with $2L_m$ 2D SPEs (in accordance to the principles shown in Figure 6(c)).

TABLE 5: Parameters from configuration registers, expressed by the number of needed *STFT_Load* cycles.

Configuration register	Parameter specified and its description	Parameter's value
Start Convolution (SC)	Start of the convolution window operation	$L_m N + L_m$
FIFO Delay (FD)	Delay for generating data in row's time index	$N - (2L_m + 1)$
Conv. Win. Size (CWS)	Size of convolution window	$2L_m + 1$
Bottom Border (BB)	Bottom border position	$N \times (N - 1)$
End of Frame (EOF)	End of frame position	$N \times N - 1$

TABLE 6: Output registers lengths for used digital units depending on the parameters l , L_m , and N .

Length of	MUXs	MULTs	ADD	ShLeft	CumADD, OutREG
Parameters l, L_m, N	l	$2 \cdot l$	$\text{CEIL}(\log_2(2 \cdot (2^{2l} - 1)))$	$2 \cdot l + 1$	$\text{CEIL}(\log_2((2^{2l+1} - 1) \times (2L_m^2 + 3L_m + 1)))$

TABLE 7: Summarized resource utilization for real device and $L_m = 5$, $N = 64$ and 2D STFT data length $l = 8$.

Chip Family	Recommended Device	Logic utilization	Combinational ALUTs	Dedicated logic registers	Total registers
Stratix II	EP2S15F672C5	19%	1,027/12,480 (8%)	1,905/12,480 (15%)	1905
Total I/O Pins used	Total virtual pins used	Total block memory bits used	DSP block 9-bit elements	Total PLLs	Total DLLs
345/367 (94%)	0	75,066/419,328 (18%)	2/96 (2%)	0/6 (0%)	0/2 (0%)

TABLE 8: Hardware complexity of the considered implementations of 2D systems.

Implementation	No. of used functional units			No. of memory locations
	No. of adders	No. of multipliers	No. of shift left reg.	
Parallel (when it is possible)	$4L_m^2 + 4L_m + 1$	$4L_m^2 + 4L_m + 2$	$4L_m^2 + 4L_m$	$4NL_m + 4L_m + 8$
MCI with a fixed number of CLKs	3	2	2	$4NL_m + 2L_m^2 + 6L_m + 10$
Proposed signal adaptive	5	6	2	$2N^2 + 4NL_m + 2L_m^2 + 8L_m + 10$

TABLE 9: CLK cycle times and execution times (by frequency-frequency point) of the considered implementations of 2D systems. T_{cP} , T_{cSF} , T_{cSA} are CLK cycle times in the cases of the parallel design, MCI one with a fixed number of CLKs and the signal adaptive one, respectively, whereas T_s is the 1-bit shift time. Execution time of the proposed design has been calculated for the considered signal (5) and $N = 64$, $L_m = 5$. Total number of CLKs taken by the proposed design is 35326, whereas the average number of CLKs by frequency-frequency point is $35326/64^2 = 8.6245$.

Implementation	Clock cycle time	Execution time
Parallel (when it is possible)	$T_{cP} = 2T_m + (2L_m^2 + 2L_m + 2)T_a + T_s$	T_{cP}
Serial with a fixed number of CLKs	$T_{cSF} = T_m + 2T_a + T_s$	$(2L_m^2 + 2L_m + 2)T_{cSF}$
Proposed signal adaptive	$T_{cSA} = 2T_m + 2T_a + 2T_{comp}$	$8.6245 \times T_{cSA}$

been derived from (8) in four CLKs, since $x_{0,-1} = 0$ ($(-3)^2 + 9^2 = 90 < R^2 = 94$) and $x_{-1,0} = 0$ ($1^2 + 1^2 = 2 < R^2 = 94$). In the third marked instant, corresponding to the region of support frequency-frequency point, $\text{CTFWD_OUT} = (-10)^2 + 10^2 + 2 \times (10 \times 1 + (-10) \times (-11)) = 440$ has been derived from (8) in five CLKs, since $x_{0,-2} = 0$ ($(-3)^2 + 9^2 = 90 < R^2 = 94$) and $x_{-1,0} = 0$ ($1^2 + 1^2 = 2 < R^2 = 94$). Variable number of CLKs, taken by the proposed design in different frequency-frequency points, can easily be observed by considering the period of the *STFT_Load* cycle, or the period of the RESET

cycle in Figure 8. Illustration of the number of taken CLKs in the corresponding frequency-frequency points is given by the gray-scale shaded graph in Figure 9.

5. Comparative Analysis

The proposed MCI signal adaptive design will be compared with the other implementations of systems for S/SF signal analysis: the possible parallel one with a fixed CLK cycle,



FIGURE 8: Illustration of the simulation results around the frequency-frequency instant when the region of support is detected.

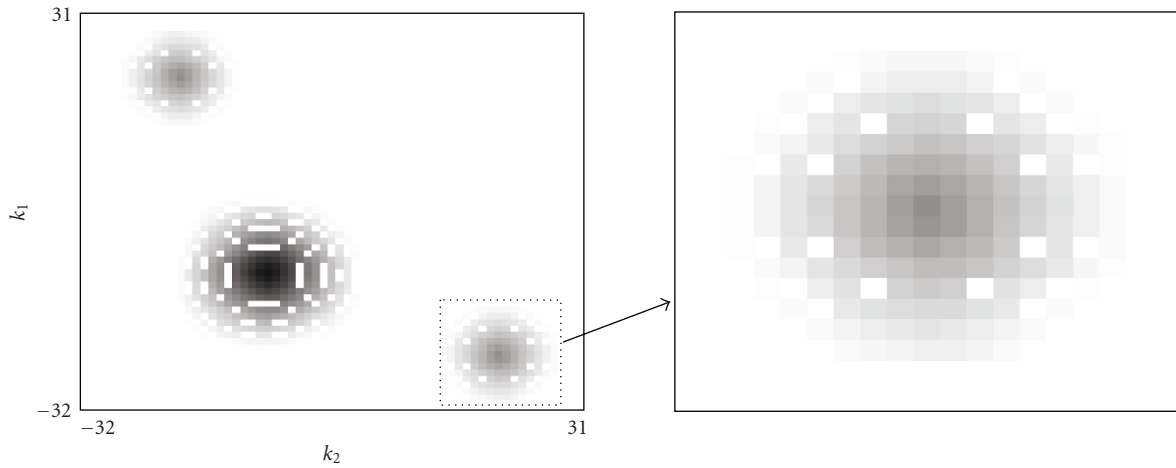


FIGURE 9: Illustration of the number of CLKs taken by the proposed design in corresponding frequency-frequency points during the execution. Different numbers of CLKs are illustrated by gray-scale shaded points, where the white ones denote minimal number of 2 CLKs, whereas the black ones denote maximum number of $2L_m^2 + 3L_m + 2 = 67$ CLKs.

[33], and the existing MCI one with a fixed number of CLKs, [34]. Trade-offs and comparisons of the considered implementation approaches are summarized in Tables 8 and 9. In general, MCI designs imply both minimal hardware requirements and much shorter CLK cycle time in comparison to the parallel design. Note that the minimal total number of used memory locations in parallel design is the consequence of the LUT memory absence in this case (i.e., absence of LUTs of $2L_m^2 + 3L_m + 2$ locations and of $2L_m^2 + 2L_m + 2$ locations in the cases of the proposed signal adaptive design, Table 3, and the MCI design with a fixed number of CLKs, [34], resp.). Also, the proposed signal adaptive design includes two input memories, capacity of maximum N^2 locations, used to store real and imaginary

parts of input 2D STFT elements and to allow importing (in process) of these elements with a variable (*STFT_Load*) cycle. Input memories additionally allow the fastest possible sampling interval of the analyzed analogue 2D signal in all considered implementation cases. However, observe that the total number of used memory locations remains quite small in all these cases.

On the other side, MCI designs generally require longer execution time. However, the proposed signal adaptive design allows the implemented S/SF system to take variable number of CLKs (the only necessary ones that provide CTFWD signal representation quality) in different frequency-frequency points within the execution: the minimal one outside regions of support (where the greater

part of total frequency-frequency points commonly lie), the higher one inside these regions, and the possible maximum one only around the central points of each region of support, Figure 9. In this way, the proposed design can significantly improve the execution time with respect to the other designs, removing the main drawback of the MCI architectures in comparison to the parallel ones, [34]. For example, in the analyzed signal (5) case, when $L_m = 5$, $N = 64$ are applied, the proposed design improves execution time in comparison to the other corresponding designs for $T_s, T_{\text{comp}} \ll T_m < 2.935 \times T_a$, Table 9. Finally, only the proposed design produces a pure 2D CTFWD signal representation, optimizing both the calculation complexity and the noise influence suppression. Non-adaptive systems cannot produce so high S/SF representation quality, Figures 2, 3, and 6.

In accordance with the design principle considered here, the number of CLKs taken in different frequency-frequency points corresponds to the number of real summation terms (or to the number of real multiplications) in (8) incremented for the completion CLK and, in some cases (region of support points that include non-maximum number of summations), for some termination CLKs. Therefore, Figure 9 can simultaneously approximate the distribution of the number of necessary operations taken by frequency-frequency point. In line with this notation, observe that, in this case, white shaded frequency-frequency points correspond to 2 complex additions (used in the recursive 2D STFT calculation, [12], since 2D SPEC calculation includes only one summation term and does not include summation execution) and 3/2 complex multiplications (2 real multiplications-used, according to (8), in real and in imaginary computational line-corresponding to a half of a complex multiplication and a complex multiplication for the recursive 2D STFT calculation, [12]). In addition, note that this observation is in a full correspondence with theoretical analysis, summarized in Table 1.

6. Conclusion

Signal adaptive multiple-clock-cycle hardware design for S/SF signal analysis has been developed, based on the 2D method for improved S/SF signal representation, also proposed here. The proposed design optimizes critical design performances, related to hardware complexity, making it possible to perform S/SF analysis by using standard devices like FPGA. Allowing the developed 2D system to take a variable number of CLKs in different frequency-frequency points, the signal adaptive design optimizes the execution time, significantly improving it in comparison to the other designs (the parallel one and the existing MCI one with a fixed number of CLKs). In this way, the design simultaneously produces the optimal—2D WD—presentation of autoterms, as well as the pure cross-terms-free S/SF signal representation in the practically only important case of the nonstationary multicomponent signals having different widths of autoterms.

References

- [1] L. Cohen, *Time-Frequency Analysis*, Prentice-Hall, Upper Saddle River, NJ, USA, 1995.
- [2] W. J. Pielemeier, G. H. Wakefield, and M. H. Simoni, "Time-frequency analysis of musical signals," *Proceedings of the IEEE*, vol. 84, no. 9, pp. 1216–1230, 1996.
- [3] F. Hlawatsch and G. F. Boudreaux-Bartels, "Linear and quadratic time-frequency signal representations," *IEEE Signal Processing Magazine*, vol. 9, no. 2, pp. 21–67, 1992.
- [4] J. Jeong and W. J. Williams, "Kernel design for reduced interference distributions," *IEEE Transactions on Signal Processing*, vol. 40, no. 2, pp. 402–412, 1992.
- [5] L.J. Stanković, "A method for time-frequency analysis," *IEEE Transactions on Signal Processing*, vol. 42, pp. 225–229, 1994.
- [6] L.J. Stanković, "Multitime definition of the Wigner higher order distribution: L-Wigner distribution," *IEEE Signal Processing Letters*, vol. 1, no. 7, pp. 106–109, 1994.
- [7] L.J. Stanković, V. Ivanović, and Z. Petrović, "Unified approach to noise analysis in the Wigner distribution and spectrogram," *Annales des Télécommunications*, vol. 51, no. 11-12, pp. 585–594, 1996.
- [8] D. E. Dudgeon and R. M. Mersereau, *Multidimensional Digital Signal Processing*, Prentice-Hall, Upper Saddle River, NJ, USA, 1984.
- [9] J. W. Woods, *Multidimensional Signal, Image and Video Processing and Coding*, Elsevier, Amsterdam, The Netherlands, 2006.
- [10] L. D. Jacobson and H. Wechsler, "Joint spatial/spatial-frequency representation," *Signal Processing*, vol. 14, no. 1, pp. 37–68, 1988.
- [11] A. Jakobsson, S. L. Marple Jr., and P. Stoica, "Computationally efficient two-dimensional capon spectrum analysis," *IEEE Transactions on Signal Processing*, vol. 48, no. 9, pp. 2651–2661, 2000.
- [12] S. Stanković, L.J. Stanković, and Z. Uskoković, "On the local frequency, group shift, and cross-terms in some multidimensional time-frequency distributions: a method for multidimensional time-frequency analysis," *IEEE Transactions on Signal Processing*, vol. 43, no. 7, pp. 1719–1724, 1995.
- [13] Y. M. Zhu, F. Peyrin, and R. Goutte, "Transformation de wigner-ville: description d'un nouvel outil de traitement du signal et des images," *Annales des Télécommunication*, vol. 42, no. 3-4, pp. 105–118, 1987.
- [14] Y. M. Zhu, R. Goutte, and M. Amiel, "On the use of two-dimensional Wigner-Ville distribution for texture segmentation," *Signal Processing*, vol. 30, no. 3, pp. 329–353, 1993.
- [15] G. Cristobal, C. Gonzalo, and J. Bescos, "Image filtering and analysis through the Wigner distribution function," in *Advances in Electronics and Electron Physics*, P. W. Haekes, Ed., Academic Press, Boston, Mass, USA, 1991.
- [16] J. Hormigo and G. Cristobal, "High resolution spectral analysis of images using the pseudo-wigner distribution," *IEEE Transactions on Signal Processing*, vol. 46, no. 6, pp. 1757–1763, 1998.
- [17] S. R. DeGraaf, "SAR imaging via modern 2-D spectral estimation methods," *IEEE Transactions on Image Processing*, vol. 7, no. 5, pp. 729–761, 1998.
- [18] L.J. Stanković, S. Stanković, and I. Djurović, "Space/spatial-frequency analysis based filtering," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2343–2352, 2000.

- [19] I. Djurović and LJ. Stanković, "Nonparametric algorithm for local frequency estimation of multidimensional signals," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 467–474, 2004.
- [20] A. Papoulis, *Signal Analysis*, McGraw-Hill, New York, NY, USA, 1977.
- [21] K. J. R. Liu, "Novel parallel architectures for short-time Fourier transform," *IEEE Transactions on Circuits and Systems II*, vol. 40, no. 12, pp. 786–790, 1993.
- [22] K. J. R. Liu and C.-T. Chiu, "Unified parallel lattice structures for time-recursive discrete cosine/sine/Hartley transforms," *IEEE Transactions on Signal Processing*, vol. 41, no. 3, pp. 1357–1377, 1993.
- [23] M. G. Amin and K. D. Feng, "Short-time Fourier transforms using cascade filter structures," *IEEE Transactions on Circuits and Systems II*, vol. 42, no. 10, pp. 631–641, 1995.
- [24] K. Maharatna, A. S. Dhar, and S. Banerjee, "A VLSI array architecture for realization of DFT, DHT, DCT and DST," *Signal Processing*, vol. 81, no. 9, pp. 1813–1822, 2001.
- [25] B. Boashash and P. J. Black, "An efficient real time implementation of the Wigner-Ville distribution," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 35, no. 11, pp. 1611–1618, 1987.
- [26] P. Gonçalves and R. G. Baraniuk, "Pseudo affine wigner distributions: definition and kernel formulation," *IEEE Transactions on Signal Processing*, vol. 46, no. 6, pp. 1505–1516, 1998.
- [27] C. Richard, "Time-frequency-based detection using discrete-time discrete-frequency Wigner distributions," *IEEE Transactions on Signal Processing*, vol. 50, no. 9, pp. 2170–2176, 2002.
- [28] L. L. Scharf, "Toeplitz and Hankel kernels for estimating time-varying spectra of discrete-time random processes," *IEEE Transactions on Signal Processing*, vol. 49, no. 1, pp. 179–189, 2001.
- [29] D. Petranović, S. Stanković, and LJ. Stanković, "Special purpose hardware for time frequency analysis," *Electronics Letters*, vol. 33, no. 6, pp. 464–466, 1997.
- [30] S. Stanković and LJ. Stanković, "An architecture for the realization of a system for time-frequency signal analysis," *IEEE Transactions on Circuits and Systems II*, vol. 44, no. 7, pp. 600–604, 1997.
- [31] S. Stanković, LJ. Stanković, V. Ivanović, and R. Stojanović, "An architecture for the VLSI design of systems for time-frequency analysis and time-varying filtering," *Annales des Télécommunication*, vol. 57, no. 9-10, pp. 974–995, 2002.
- [32] V. N. Ivanović, R. Stojanović, and LJ. Stanković, "Multiple-clock-cycle architecture for the VLSI design of a system for time-frequency analysis," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 60613, 18 pages, 2006.
- [33] S. Stanković, I. Djurović, and V. Vuković, "System architecture for space-frequency image analysis," *Electronics Letters*, vol. 34, no. 23, pp. 2224–2225, 1998.
- [34] V. N. Ivanović and R. D. Stojanović, "An efficient hardware design of the flexible 2-D system for space/ spatial-frequency signal analysis," *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 3116–3125, 2007.
- [35] LJ. Stanković and J. F. Böhme, "Time-frequency analysis of multiple resonances in combustion engine signals," *Signal Processing*, vol. 79, no. 1, pp. 15–28, 1999.
- [36] R. C. Gonzales and R. E. Woods, *Digital Image Processing*, Prentice-Hall, Upper Saddle River, NJ, USA, 2002.
- [37] LJ. Stanković and S. Stanković, "On the Wigner distribution of discrete-time noisy signals with application to the study of quantization effects," *IEEE Transactions on Signal Processing*, vol. 42, no. 7, pp. 1863–1867, 1994.
- [38] S. B. Hearon and M. G. Amin, "Minimum-variance time-frequency distribution kernels," *IEEE Transactions on Signal Processing*, vol. 43, no. 5, pp. 1258–1262, 1995.
- [39] A. Iborra, C. Fernández, B. Álvarez, and J. M. Fernández-Merono, "FPGA solution of low cost applications of real-time AVI systems," *Dedicated System Magazine*, vol. Q2, pp. 79–84, 2001.
- [40] Altera Corp., "Altera Cooperation, Datasheet," San Jose, Calif, USA, May 2007, <http://www.altera.com%5Cliterature/lit-stx2.jsp>.