Joint Time–Frequency Segmentation Algorithm for Transient Speech Decomposition and Speech Enhancement

Charturong Tantibundhit, *Member, IEEE*, Franz Pernkopf, *Member, IEEE*, and Gernot Kubin, *Member, IEEE*

***Abstract*—We develop an algorithm, the joint time–frequency segmentation algorithm, where the wavelet packet coefficients of the analyzed speech signal are represented as tiles of a time–frequency representation adapted to the characteristics of the signal itself. Further, our algorithm enables the decomposition of the speech signal into transient and non-transient components, respectively. Any block of wavelet packet coefficients, whose tiling height is larger than or equal to the tiling width belongs to the transient component and vice versa for the non-transient component. The transient component is selectively amplified and recombined with the original speech to generate the modified speech with energy adjusted to be equal to the original speech. The intelligibility of the original and modified speech is evaluated by 16 human listeners. Word recognition rate results show that the modified speech significantly improves speech intelligibility in background noise, i.e., by 10% absolute at 0 dB to 27% absolute at −30 dB.**

***Index Terms*—Speech enhancement, transient component, speech intelligibility, wavelet packet transform, joint time–frequency (TF) segmentation.**

I. Introduction

DURING the past decades, there has been a vast increase in research focused on improving the intelligibility of speech presented in background noise, which can be divided into two categories. Speech enhancement approaches of the first category aim to increase the intelligibility of speech already corrupted with noise by minimizing its effect as much as possible. Many speech enhancement approaches in the past decades belong to this category [1], e.g., signal subspace approaches [2] and spectral subtraction [3]. These approaches have been applied to the noisy speech arriving at the listener, where the properties of noise, e.g., its spectrum are often assumed to be available [4]. Although these approaches show impressive improvements [2], [3], they may not work well under the conditions of unknown noise. However, some of these methods use adaptive noise estimation which rectifies performance.

Speech enhancement approaches of the second category are focused on increasing the intelligibility of clean speech before it is degraded by noise [4], where the clean speech is assumed to be available for processing before getting transmitted to the listener located in the noisy environment [5]–[8]. One possible application of speech enhancement in this category is radio communication between a centrally located coordinator in a quiet environment with field operators in noisy environments. Sauert and Vary [5] process the clean speech (referred to as the far-end speech) before it is transmitted to the listener in a noisy environment (referred to as the near-end listener). Their algorithm raises the average speech spectrum over the average noise spectrum. Their results show the improvement of speech intelligibility in background noise. However, the noise spectrum is assumed to be known [5] similar to several speech enhancement approaches in the first category.

Several researchers have developed speech enhancement approaches applied to clean speech without requiring detailed knowledge about the background noise [6]–[8]. These approaches showed that the amplification of speech features is important to speech perception in particular for consonant and transition regions [9]. Hazen and Simpson [6] manually annotated consonant regions and formant transitions of the clean speech composed of nonsense syllables and sentences and then amplified these regions to increase speech intelligibility presented in background noise. They preferred the manual approach to avoid errors possibly occurring during automatic speech segmentation. They found that the emphasis of consonants and transition regions provided a significant improvement of speech intelligibility of about 10% absolute in signal-to-noise ratio (SNR) levels of 0 and −5 dB. However, this approach isolates the consonants and transitions manually and cannot be applied automatically to improve speech intelligibility [7].

Yoo *et al.* [7], which will be referred to as Y's algorithm throughout the paper, have developed an approach to capture speech features automatically. First, the original speech is high-pass filtered at 700 Hz to remove the first formant. Three time-varying bandpass filters are applied to capture the three strongest formants of the high-pass filtered speech referred to as the quasi-steady-state (QSS) component. The QSS component is expected to include quasi-steady or slowly changing short-time spectra

Manuscript received May 03, 2009; revised September 23, 2009. First published October 30, 2009; current version published July 14, 2010. The work of C. Tantibundhit was supported by the Asean-European University Network (ASEA-UNINET) under a Postdoctoral Technology Grants Scholarship. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Richard C. Rose.

C. Tantibundhit is with the Department of Electrical and Computer Engineering, Thammasat University, Pathumthani 12120, Thailand (e-mail: tchartur@engr.tu.ac.th).

F. Pernkopf and G. Kubin are with the Signal Processing and Speech Communication Laboratory (SPSC), Graz University of Technology, 8010 Graz, Austria (e-mail: pernkopf@tugraz.at; g.kubin@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2009.2035037

of vowel formants and consonant hubs [7]. The QSS component is subtracted from the high-pass filtered speech resulting in the transient component. The transient component is selectively amplified and recombined with the original speech to generate the modified speech with energy adjusted to be equal to the energy of the original speech. The intelligibility of the modified speech in background noise is compared to that of the original speech using a psychoacoustic test based on the modified rhyme test protocol [10]. The modified speech significantly improves speech intelligibility at low SNRs, i.e., up to 32% absolute at -25 dB, and has minimal effect on intelligibility at higher SNR levels. They concluded that the transient component is important to speech perception [7]. However, the resulting transient component appears to retain a significant amount of formant energy during what would appear to be QSS regions of the speech [8]. In addition, their approach focuses on the high pass filtered speech.

Tantibundhit *et al.* [8], which will be referred to as T's algorithm throughout the paper, have developed another approach to capture speech features directly from the original speech signal. They decompose speech into three components, i.e., the tonal, transient, and residual components, respectively. The modified discrete cosine transform (MDCT) is used to capture constant or slowly varying frequency information in speech referred to as the tonal component. The wavelet transform is used to capture abrupt changes in speech referred to as the transient component. The residual component is expected to have small energy with a flat spectrum. The hidden Markov model (HMM) [11] and the hidden Markov tree model (HMT) [12], [13] are used to capture statistical dependencies between the MDCT coefficients and between the wavelet coefficients, respectively. The transient component is used to enhance speech intelligibility in background noise in the same fashion as done in Y's algorithm. The psychoacoustic test results have shown that the transient component significantly improves speech perception in background noise at low SNR levels, i.e., up to 18% absolute at -25 dB and has minimal effect at higher SNR levels. Although, this approach decomposes the transient component more effectively than Y's algorithm, i.e., by removing vowel formants more effectively and emphasizing abrupt changes represented as vertical edges in time–frequency, the obtained transient component suffers from pre-echo distortion artifacts of the MDCT [14] when estimating the tonal component. This may explain the lower improvements of speech intelligibility compared to Y's algorithm [8].

In this paper, we develop an approach to capture the transient component in speech signals even more effectively. Specifically, first, we want to avoid pre-echo distortion artifacts due to the use of the MDCT. Second, we aim to develop an unconstrained multiresolution analysis, where both time and frequency tilings are adapted directly to the characteristics of the speech signal instead of using the fixed time–frequency tilings of the MDCT and the wavelet transform as in [8].

Herley *et al.* [15] proposed a joint space–frequency segmentation using balanced wavelet packet trees for least-cost image representation. When applying their algorithm to time-domain signals instead of space-domain images, it allows arbitrary multiresolution analysis both in time and in frequency based on the minimum quantization error (cost). Their goal is to achieve a desired bit rate with minimum distortion or a desired distortion with the lowest bit rate [15]. The wavelet packet decomposition is used to transform the signal to a desired decomposition level resulting in blocks of coefficients (at the considered level) and blocks of transformed coefficients (at the coarser level). The costs of each block and its transformed blocks (low-frequency and high-frequency) are calculated based on the quantization error. A time split is performed for the considered block if its cost is lower than the cost of its transformed block and vice versa for a frequency split. They showed that, at a desired bit rate, their approach provided a lower distortion with higher peak signal-to-noise ratio (PSNR) in dB than other image coding approaches [15]. A similar algorithm has been independently studied and proposed by Thiele and Villemoes [16]. The time and frequency splits of Herley *et al.* are selected such that a minimum coding rate is achieved. They do not consider the segmentation adapted to the signal characteristics.

Tantibundhit *et al.* [17] modified the algorithm of Herley *et al.* such that it can provide an unconstrained multiresolution analysis in time and frequency adapted to the characteristics of the signal. To achieve this, they adopted a cost function based on the entropy-based approach of Coifman and Wickerhauser [18] to calculate the cost of the block coefficients and the transformed block coefficients. Experimental results on a synthetic signal, composed of a high-frequency sinusoid and a single impulse, show that their algorithm outperforms several time–frequency representations such as the (best basis) wavelet packet decomposition [18], the (best basis) MDCT [18], and the algorithm of Herley *et al.* They suggest that their algorithm referred to as the joint time–frequency segmentation algorithm might be useful in speech decomposition problems.

Further, the joint time–frequency segmentation algorithm has been further modified, i.e., instead of calculating the entropy (cost) directly from the coefficients in each block, the coefficients are windowed by the Hann window with 50% overlap then the averaged energy for each window is calculated. The cost of each coefficient block is the entropy calculated based on these averaged and windowed energies. This algorithm is applied for the transient decomposition of speech signals [19], i.e., the speech signal is transformed using the wavelet packet transform to a desired level resulting in blocks of coefficients, where the numbers of coefficients of each block in every level is equal to the numbers of coefficients at the coarsest level. The entropy (cost) of each block is calculated. A heuristic is developed, which results in a combination of time–frequency tilings with minimum costs. The transient component is estimated by the inverse transform of the significant coefficient, where the tiling height of the block coefficients is larger than or equal to the tiling width. The transient component is used to enhance speech intelligibility in background noise. The psychoacoustic test results have shown that the transient component is important to speech perception and can improve speech intelligibility up to 31% absolute at -30 dB compared to the original speech. In addition, this algorithm can improve speech intelligibility, even if the intelligibility is already high (for SNRs better than -10 dB [8]), while speech intelligibility of speech modified by Y's algorithm and T's algorithm is not better than that of the original speech at 0 and -5 dB.

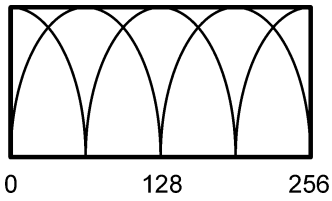


Fig. 1. Block of 256 wavelet packet coefficients windowed by the Hann window.

This paper presents the algorithm, briefly introduced in previous work [19], for the application of speech enhancement in a much more elaborated and comprehensive way. Additionally, we perform a direct experimental comparison with Y's and T's algorithms.

The paper is organized as follows: Details of our algorithm including the modified forward and backward algorithms are described in Section II. Examples of speech decomposition results are presented in Section III. The direct comparison of the transient components from our algorithm, Y's algorithm, and T's algorithm is also discussed and explained in this section. An approach to generate the modified speech is also described in this section. The experimental setup of a modified rhyme test used to evaluate the intelligibility is shown in Section IV. The psychoacoustic test results comparing the original speech to the modified speech generated with our algorithm, Y's algorithm, and T's algorithm are presented in Section V. Implications of the results are discussed in Section VI. Finally, conclusions and future work are presented in Section VII.

II. Joint Time–Frequency Segmentation Algorithm

The original signal $x_{\mathrm{orig}}(t)$ sampled at 11.025 kHz, is transformed using the wavelet packet transform [20] limited to the coarsest level J composed of 256 coefficients (23.2 ms). The Daubechies-16 (Db16) wavelet is chosen as a mother wavelet because it gives a better estimation of the transient component across 300 monosyllabic consonant-vowel-consonant (CVC) rhyming words of House *et al.* [21] compared with the results from other mother wavelets (Db4, Db8, Db14, Db18, and Db20) based on visual inspection and informal listening experiments.

From the finest level (level 1) to the coarsest level (level J), the wavelet packet coefficients in each bin are divided into blocks of coefficients, each of which is composed of 256 coefficients. Then, all of the blocks of coefficients are windowed by the Hann window based on the idea of Learned [22]. In a classification task, the use of all wavelet packet coefficients in the block may lead to miss strong time-dependent features such as the transient information. Hence, it may be beneficial to calculate a windowed energy [22]. A window size of 128 coefficients (11.6 ms) with 50% overlap is chosen resulting in a half-window at the beginning and at the end of the block and three full windows, respectively, as illustrated in Fig. 1. The averaged energy for each set of windowed coefficients is calculated resulting in five averaged energies. Finally, the entropy [23] of each block is calculated based on these averaged energies following the entropy-based cost function proposed by Coifman and Wickerhauser [18]. This is referred to as cost of the coefficient block.

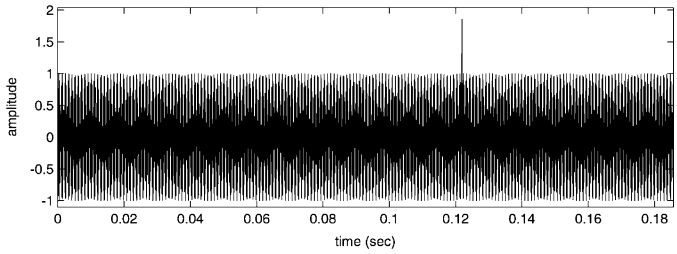


Fig. 2. A 2048-sample synthetic signal composed of high frequency (5 kHz) sinusoid and a single impulse located at sample index 1345.

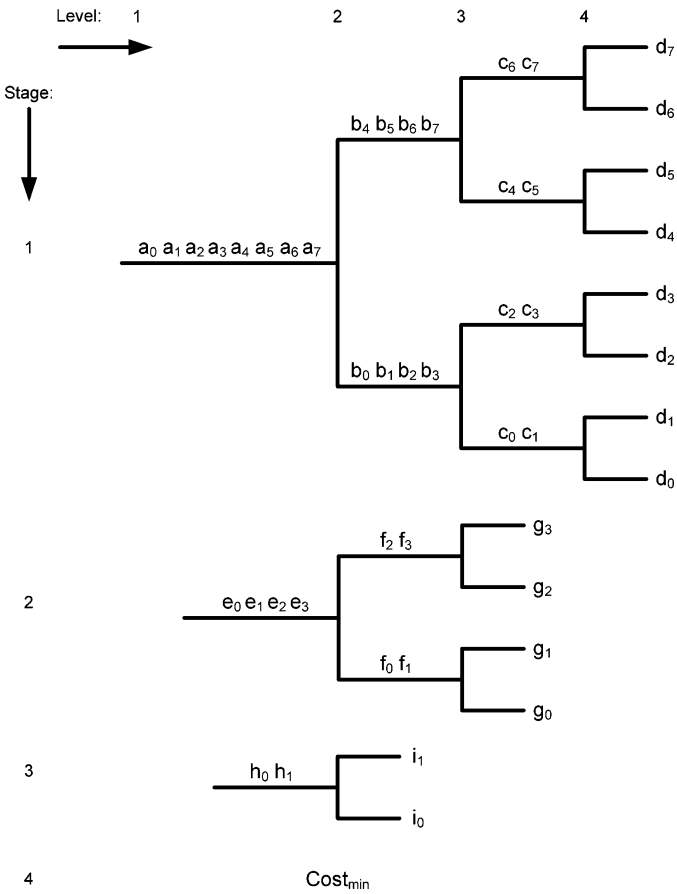


Fig. 3. Block decomposition in the time–frequency plane as done in the modified forward algorithm graphically (regenerated from Fig. 5 of Herley *et al.* [15]).

The next step is to evaluate all of the possible combinations of time–frequency tilings in every level (level 1 to level J) and find the combinations of time–frequency tilings that achieve the minimum cost. This can be done by the modified forward and backward algorithms. We introduce a synthetic signal to be used for our explanations. A 2048-sample synthetic signal, composed of a high-frequency (5 kHz) sinusoid and a single impulse located at the sample index 1345, is illustrated in Fig. 2. The tilings are expected to be split in frequency for the 5 kHz sinusoid and to be split in time for the location of the single impulse. Both algorithms are explained in Sections II-A and II-B and are summarized as pseudocode in Algorithm 1 and 2, respectively.

A. Modified Forward Algorithm

Starting from the first stage, and moving from level 1 to level J ($J = 4$ for the example in Fig. 3), the sum of the cost of two adjacent blocks of a considered level and the sum of the corresponding two transformed blocks (low-frequency and high-frequency) at the next level are compared. The alphanumeric symbol in the figure represents the cost of the coefficient block and its subscript represents the coefficient block number. For this example, in every level, there are coefficient blocks resulting in eight cost values shown in stage 1 of Fig. 3. Specifically, for level 1 versus level 2, the partial cost $\mathrm{sum}(a_0, a_1)$ is compared with $\mathrm{sum}(b_0, b_4)$ and the partial cost $\mathrm{sum}(a_2, a_3)$ is compared with $\mathrm{sum}(b_1, b_5)$, and so on. We write the partial costs of the winners as

$$e_i = \min\{\mathrm{sum}(a_{2i}, a_{2i+1}), \mathrm{sum}(b_i, b_{i+4})\}. \quad (1)$$

Similarly, for level 2 versus level 3, $\mathrm{sum}(b_0, b_1)$ is compared with $\mathrm{sum}(c_0, c_2)$ and $\mathrm{sum}(b_2, b_3)$ is compared with $\mathrm{sum}(c_1, c_3)$, and so on. We write the cost of the winners as

$$f_i = \min\{\mathrm{sum}(b_{2i}, b_{2i+1}), \mathrm{sum}(c_{2i-p}, c_{2i+2-p})\} \quad (2)$$

where $p = 2(i/2 - \lfloor i/2 \rfloor)$ and $\lfloor i/2 \rfloor$ is the largest integer not greater than $i/2$, and finally

$$g_i = \min\{\mathrm{sum}(c_{2i}, c_{2i+1}), \mathrm{sum}(d_{2i}, d_{2i+1})\}. \quad (3)$$

At this point, we have the winners of the decisions of stage 1 and a new cost tree for stage 2 in Fig. 3, where the number of levels is reduced by one to $J - 1$. If the minimum of the cost of two blocks in a considered level is less than or equal to the minimum of the cost of the corresponding two transformed blocks, a time split is performed; otherwise, a frequency split is performed. The resulting time–frequency splits and the winning costs are put in the second stage. The same approach is applied (recursively) with the number of levels reduced by one from the previous stage until reaching the last stage (stage J). At this stage, there is only one level left resulting in time–frequency tilings with the minimal cost ($\mathrm{Cost}_{\min}$), which can be expressed as

$$\mathrm{Cost}_{\min} = \min\{\mathrm{sum}(h_0, h_1), \mathrm{sum}(i_0, i_1)\}. \quad (4)$$

Fig. 4 graphically summarizes the modified forward algorithm for the synthetic signal, where the number represents the cost of the coefficient block at any stage and level. The bold frame is associated with the blocks at the considered stage and level, whereas the dashed line indicates a performed split either in time or frequency. The pseudocode of the modified forward algorithm is illustrated in Algorithm 1. The function *GetCost* and *GetTrCost* are used to calculate the cost of two blocks in a considered level and the cost of the corresponding two transformed blocks, respectively. The function *PutCost* is used to store the winning cost and the function *PutWin* is used to store the winning split value, i.e., "0" for time split and "1" for frequency split.

Algorithm 1 Modified Forward Algorithm

```
1) for stage = 1 : J − 1 do
2)   if stage == 1 then
3)    lblock = block_length;
4)   else
5)    lblock = 1;
6)   end if
7)   for level = 1 : J − stage do
8)    for block = 1 : 2^(level−1) do
9)     time = GetCost(stage, level, block, lblock);
10)    freq = GetTrCost(stage, level + 1, block, lblock);
11)    if time <= freq then
12)     PutCost(stage + 1, level, block, lblock) = time;
13)     PutWin(stage + 1, level, block, lblock) = 0;
14)    else
15)     PutCost(stage + 1, level, block, lblock) = freq;
16)     PutWin(stage + 1, level, block, lblock) = 1;
17)    end if
18)   end for
19)  end for
20) end for
```

Algorithm 2 Modified Backward Algorithm

```
1) for stage = J − 1 : −1 : 1 do
2)   if stage ~= 1 then
3)    lblock = 1;
4)   else if stage == 1 then
5)    lblock = block_length;
6)   end if
7)   for level = 1 : J − stage do
8)    for block = 1 : 2^(level−1) do
9)     if GetWin(stage + 1, level, block, lblock) == 0 then
10)     ElimCoeff(stage, level + 1, lblock);
11)     ElimWin(stage, level + 1, block, lblock);
12)    else if GetWin(stage + 1, level, block, lblock) == 1
       then
13)     ElimCoeff(stage, level, block, lblock);
14)     ElimWin(stage, level, block, lblock);
15)    end if
16)   end for
17)  end for
18) end for
```

B. Modified Backward Algorithm

After the decisions on time–frequency splitting have been made for every level of the wavelet packet coefficients, the next step is to get rid of the irrelevant coefficients by the modified backward algorithm summarized in Algorithm 2. The function *GetWin* is used to retrieve the stored winning split values obtained by the modified forward algorithm. The function *ElimCoeff* and *ElimWin* are used to eliminate the irrelevant wavelet packet coefficients and the irrelevant cost value, respectively.

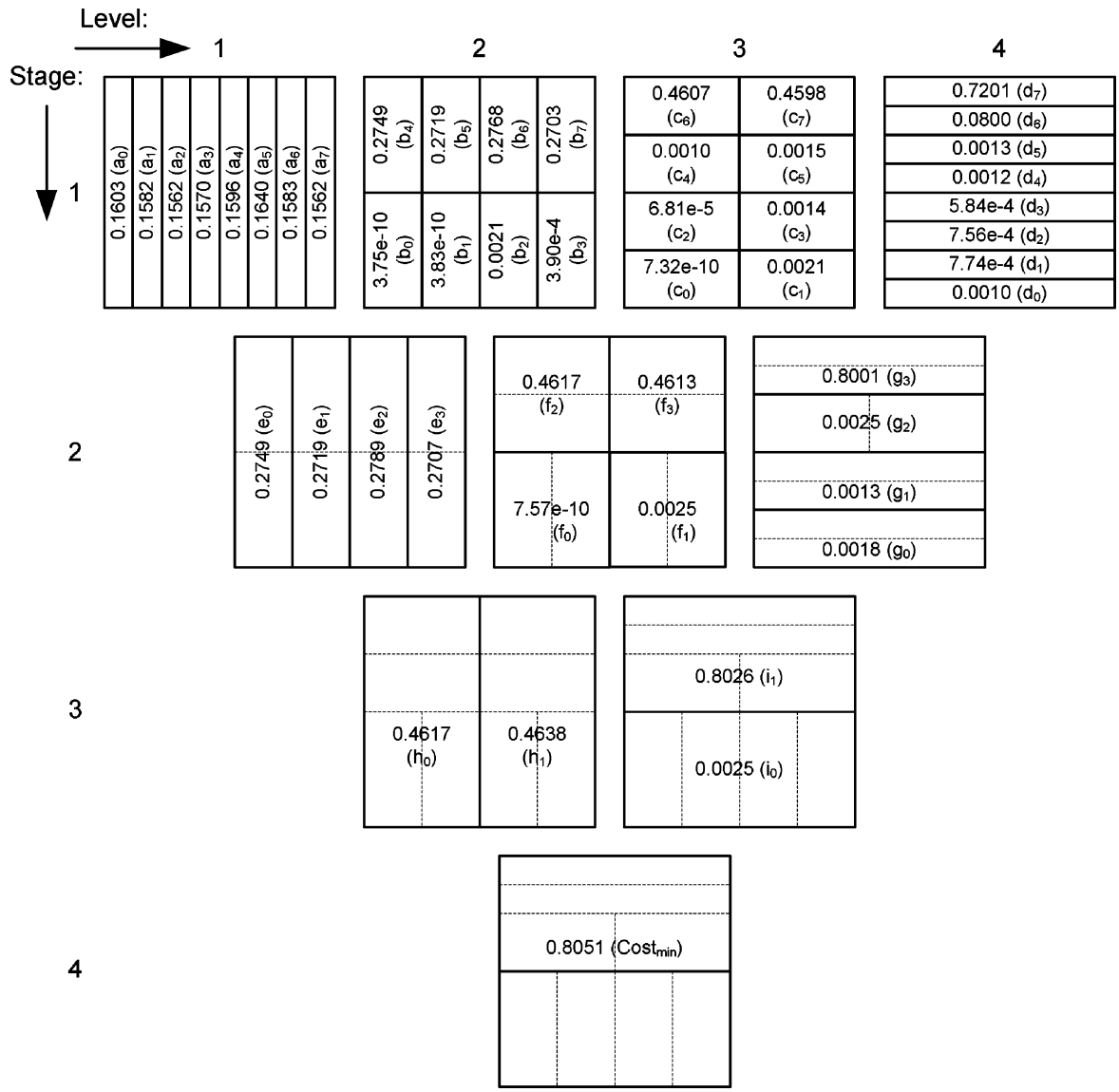


Fig. 4. Graphical representation of the modified forward algorithm for a 2048-sample synthetic signal composed of a high-frequency (5 kHz) sinusoid and a single impulse.

Fig. 5 illustrates the resulting time–frequency segmentation for the synthetic signal illustrated in Fig. 2. Consider (4), which is the last decision of the modified forward algorithm, where the outcome for this synthetic signal is known to be "frequency". This implies that $\mathrm{sum}(i_0, i_1) < \mathrm{sum}(h_0, h_1)$, and that the final cost is $\mathrm{sum}(i_0, i_1)$. Therefore, we can eliminate all of the e_i from stage 2 and all of the a_i from stage 1. Also, we can construct a first coarse time–frequency tiling as a horizontal line to represent the single frequency split as shown in Fig. 5(b), where the surviving trees are illustrated in the surviving and independent upper and lower branches, respectively.

At the next stage (stage 2), there are two decisions, which have to be performed, i.e., the decisions between the f_i and the g_i because the other decisions between the e_i and f_i are no longer of interest and have been anticipated in the previous step. Because the synthetic signal is dominated by the 5-kHz sinusoid in the high-frequency region, the upper branch is split in "frequency," i.e., $\mathrm{sum}(g_2, g_3) < \mathrm{sum}(f_2, f_3)$ and we can eliminate $b_4 \ldots b_7$. In addition, the low-frequency region is dominated by the single impulse and, therefore, the lower frequency branch is split in "time," i.e., $\mathrm{sum}(f_0, f_1) \leq \mathrm{sum}(g_0, g_1)$ and we can eliminate $d_0 \ldots d_3$. These eliminations result in the tilings of Fig. 5(c) with the respective trees. The algorithm proceeds further until reaching stage 1 and the remaining decisions affect only two blocks of coefficients. As a result, the tiling in the high-frequency region is further split in "frequency," where the 5-kHz sinusoid is located, while the tilings in the low-frequency regions are further split in "time," where the single impulse is dominant. Finally, the resulting time–frequency segmentation is illustrated in Fig. 5(d), where the tiling is split both in time and in frequency based on the characteristics of the analyzed signal.

C. Transient Estimation

After the optimal time–frequency tiling has been determined, the next step is to recover the transient component from the

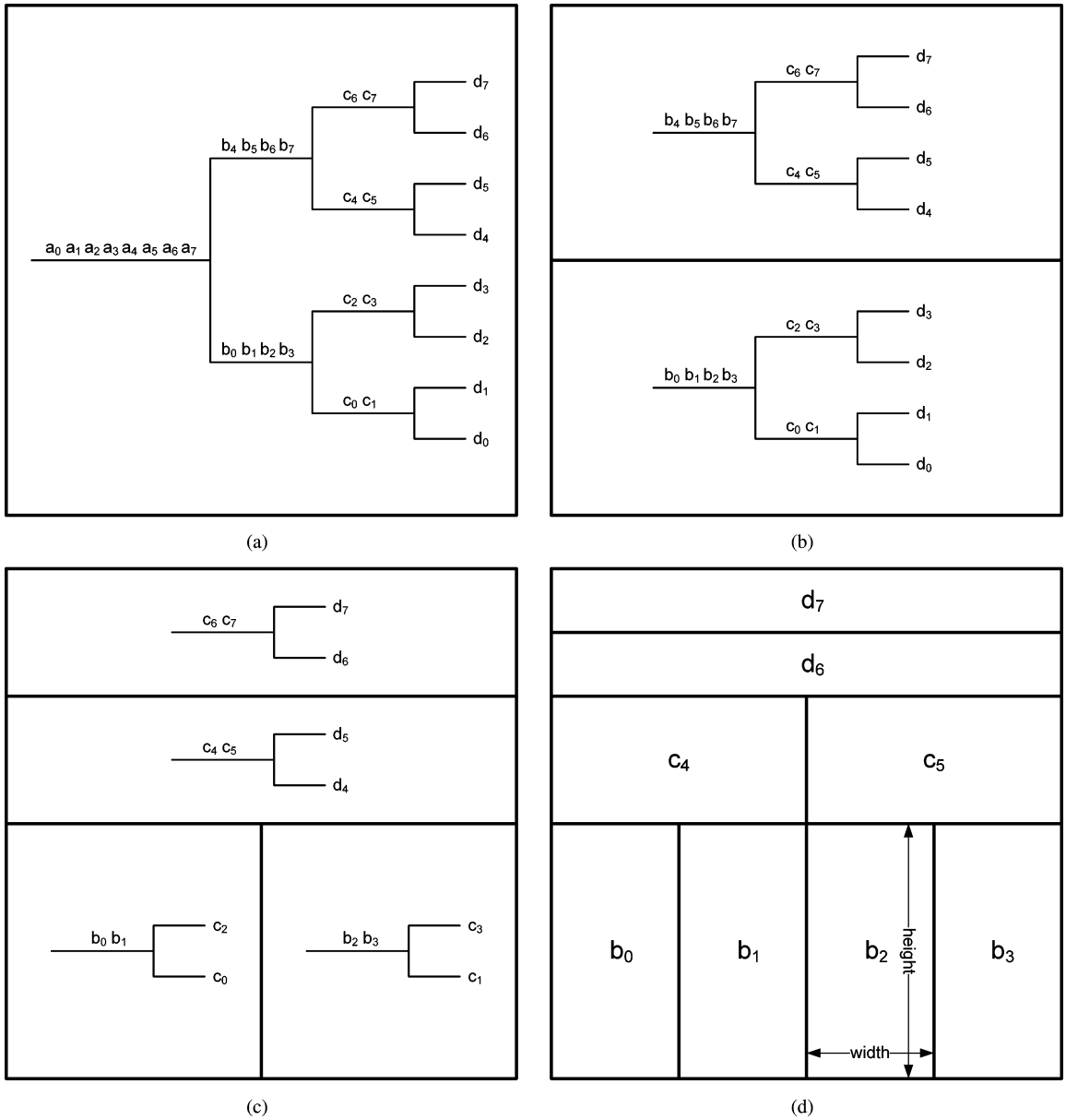


Fig. 5. Modified backward algorithm to determine the coefficient blocks of the minimum cost for a 2048-sample synthetic signal composed of a high-frequency (5 kHz) sinusoid and a single impulse located at sample index 1345.

resulting tiling. The tiling size of each coefficient block is measured relative to the full time period and whole frequency range of the signal, each interval is normalized to 1. All of the blocks of coefficients whose tiling height is more than or equal to the tiling width are characterized as transient tiles (b_0, b_1, b_2, and b_3) and vice versa for the non-transient tiles (c_4, c_5, d_6, and d_7) as illustrated in Fig. 5(d). All of the wavelet packet coefficients in the transient tiles, referred to as the significant wavelet packet coefficients, are retained, but those in the nontransient tiles, referred to as the insignificant wavelet packet coefficients, are set to zero based on the idea of transform coding [24]. Then, the transient component, $x_{\mathrm{tran}}(t)$, is simply estimated by the inverse wavelet packet transform of those significant wavelet packet coefficients. The nontransient component is obtained by the inverse wavelet packet transform of the insignificant wavelet packet coefficients or simply calculated by subtraction of the transient component from the original speech signal as

$$x_{\mathrm{nont}}(t) = x_{\mathrm{orig}}(t) - x_{\mathrm{tran}}(t). \tag{5}$$

Fig. 6(a) and (b) illustrates the resulting transient and nontransient components of the synthetic signal of Fig. 2. The transient component predominantly includes the single impulse and the nontransient component is dominated by the high-frequency sinusoid, fully matching our intuition about the desired decomposition of such signal.

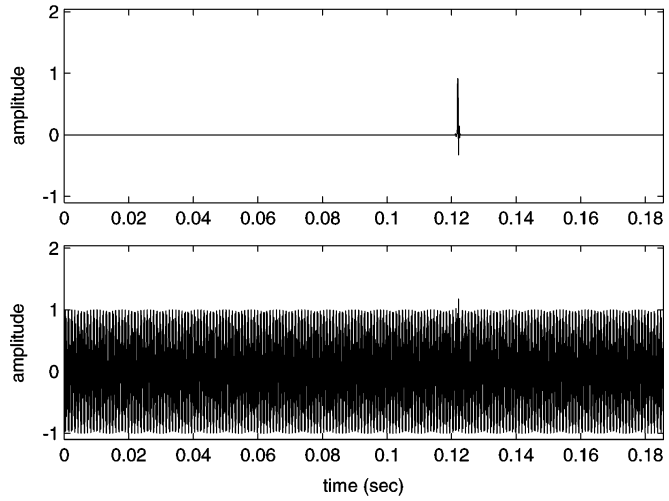


Fig. 6. (a) Transient component of the synthetic signal of Fig. 2. (b) Nontransient component.

III. Speech Decomposition Results

Three hundred monosyllabic CVC words proposed by House *et al.* [21] are decomposed using our joint time–frequency segmentation algorithm described in Section II. The nontransient component predominantly includes constant frequency information of vowel formants and consonant hubs. Hence, it includes most of the energy of the original speech. The transient component, on the other hand, includes comparatively little energy of the original speech. It emphasizes "edges" in time–frequency and includes transitions from consonants to vowels, transitions between and within vowels, and transitions at the end of vowels to consonants.

A. Results

Results are illustrated by the decomposition obtained on "bat" transcribed phonetically as /bæt/. These results are typical for all studied words. The word "bat" is chosen as an example of the decomposition results because it represents a relatively simple distinction between transient and nontransient components as illustrated in Fig. 7(a). More generally, it is composed of clear time–frequency edges, i.e., /b/ (arrow A1) and /t/ (arrow F1), visible as vertical ridges in the spectrogram, and vowel /æ/, which has fairly constant frequency information in the first (arrow B1), second (arrow C1), third (arrow D1), and fourth (arrow E1) formants, visible as horizontal ridges in the spectrogram. Consonants, transitions from consonants to vowels (arrow H1), and transitions at the end of vowels (arrow I1) should be included in the transient component. On the other hand, constant frequency information in vowels and consonant hub (arrow G1) should be assigned to the nontransient component.

The nontransient component computed by our algorithm, illustrated in Fig. 7(b), includes most of the energy (96.2%) of the speech signal. It predominantly includes the vowel /æ/ as desired and small parts of /t/ in the low-frequency region around 0.35 s (arrow A2). The transient component, shown in Fig. 7(c), includes 3.8% of the energy of the speech signal. It includes the release of the plosive /b/ (arrow A3), the transition from /b/ to vowel /æ/ (arrow B3), the transition at the end of vowel /æ/ (arrow C3), and most of the release of the plosive /t/ (arrow D3). It also includes the aspiration noise of /t/ visible as noise pattern in high-frequency regions (arrow E3).

B. Comparison of Transient Speech Component and Modified Speech From Various Algorithms

If the time and frequency tilings of the joint time–frequency segmentation algorithm are automatically adapted to the characteristics of the speech signal itself, it should provide more effective identification of the transient component compared to an algorithm that is restricted to fixed time–frequency tilings [8] or to a fixed number of time-varying bandpass filters [7]. To investigate this hypothesis, the transient component and the resulting modified speech obtained by our joint time–frequency segmentation algorithm, Y's algorithm, and T's algorithm are compared. Results for the transient component of the monosyllabic CVC word "bat" obtained by [7] and [8] are illustrated in Fig. 7(d) and (e), respectively. Results of the modified speech of the same word obtained by our algorithm, Y's algorithm, and T's algorithm are illustrated in Fig. 7(f), (g), and (h), respectively.

The transient component identified by Y's algorithm includes most of /b/ (arrow A4), most of /t/ (arrow G4), and most of the release of /t/ (arrow H4). It also includes the transition from /b/ to vowel /æ/ (arrow B4) and the transition at the end of vowel /æ/ (arrow F4) similar to our algorithm. However, the transient component appears to retain a significant amount of constant formant information, which is expected to be included in the tonal component [8]. Specifically, it includes parts of the constant frequency information of the second formant (arrow C4) and most of the constant frequency information of the third (arrow D4), and fourth formants (arrow E4), respectively. In addition, as clearly visible in Fig. 7(d), this algorithm cannot capture the transient component frequencies below 700 Hz [8].

The transient component identified by T's algorithm includes /b/ (arrow A5), /t/ (arrow F5), and most of the release of /t/ (arrow G5) similar to our algorithm and Y's algorithm. The algorithm removes constant formant frequency information more effectively than Y's algorithm illustrated as "holes" in the resulting transient component (arrow B5, C5, D5, and E5). However, as stated earlier, the vowel /æ/ is composed of fairly constant frequency information and should not be included in the transient component.

The transient component is used to improve speech intelligibility, i.e., the transient component is selectively amplified and recombined with the original speech, with the total energy adjusted to be equal to the energy of the original signal based on the idea of [7] and [8]. The optimal transient amplification factor of 12 has been selected in a similar manner as in [7], [8] based on informal listening tests among the amplification factors in the range between 1 and 15. A too small amplification factor results only in a small improvement of speech intelligibility while a large value results in a too strong emphasis of consonants and transitions in speech, leading to unnatural sounding speech and an implicit attenuation of the vowel sounds.

The modified speech from our algorithm emphasizes the edge /b/ (arrow A6), the transition into the vowel (arrow B6) and at

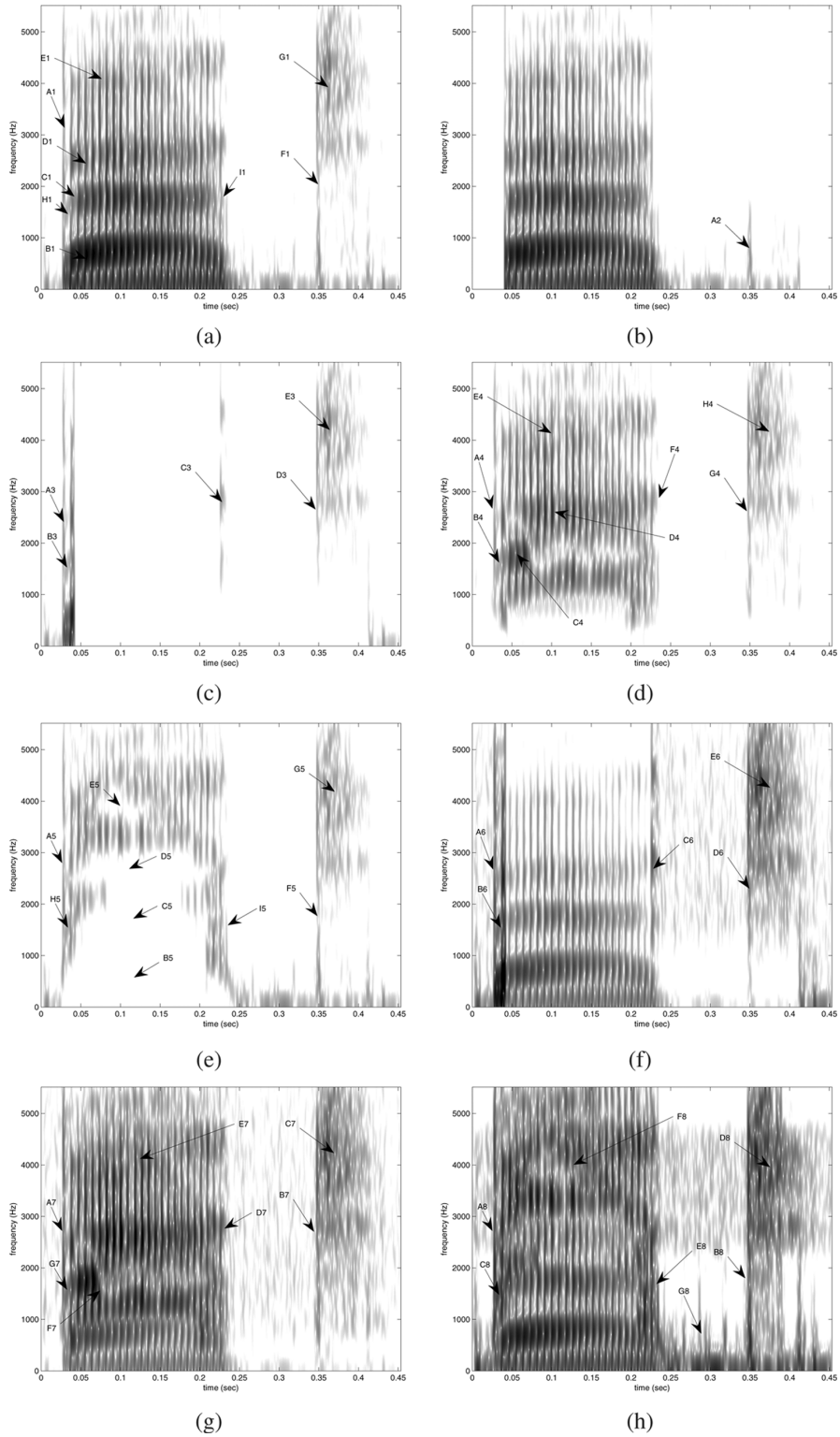


Fig. 7. Speech decomposition results and the modified speech from various algorithms. (a) Original speech of the word "bat." (b) The nontransient component from our algorithm. (c) The transient component from our algorithm. (d) The transient component from Y's algorithm. (e) The transient component from T's algorithm. (f) The modified speech from our algorithm. (g) The modified speech from Y's algorithm. (h) The modified speech from T's algorithm.

the end of the vowel /æ/ (arrow C6), the edge /t/ (D6), and the release of /t/ (arrow E6). It well preserves the formant structure of the vowel /æ/. The modified speech from Y's algorithm and T's algorithm also emphasize the edge /b/ (arrow A7 and A8), the transitions into the vowel (G7 and C8) and at the end of the vowel /æ/ (arrow D7 and E8), the edge /t/ (arrow B7 and B8) and the release of /t/ (arrow C7 and D8) similar to our algorithm. However, the modified speech from Y's algorithm emphasizes the high-frequency region (arrow E7) of the vowel with some distortions of the formant structure especially in the

low-frequency region (arrow F7), whereas the modified speech from T's algorithm introduces distortions in the formant structure (arrow F8). It also includes decomposition artifacts probably generated during the transient decomposition (arrow G8).

IV. Experimental Setup: Modified Rhyme Test Protocol

The objective of this experiment is first to investigate whether the transient component can improve the intelligibility of speech in background noise. Second, to formally compare the intelligibility of the modified speech generated from Y's algorithm and T's algorithm to our joint time–frequency segmentation algorithm (joint TF). The test protocol is a modified version of the word monitoring task of Mackersie *et al.* [10] using 300 monosyllabic CVC rhyming words proposed by House *et al.* [21].

The test protocol was performed at the Signal Processing and Speech Communication Laboratory (SPSC Lab.), Graz University of Technology, Austria, by 16 volunteer subjects (15 male and 1 female). All of the subjects have negative otologic histories, i.e., they do not have histories of hearing loss, direct injury to the ears, undergone surgery for hearing, working or recreational activities in noisy environment, etc. Moreover, they have at least one ear of hearing sensitivity of 15-dB hearing level (HL) or better by conventional audiometry (250–8 kHz). All of the volunteer subjects have at least eight years of learning English. Among them, 11 subjects have German, two subjects have Polish, and the remaining three subjects have Ladin, Urdu, and Thai as their native languages, respectively. The maximum, minimum, and average age of the subjects are 33, 21, and 27 years, respectively.

Fifty sets of rhyming monosyllabic CVC words (six words per set) were recorded by a male English native speaker at the Department of Communication Science and Disorders, University of Pittsburgh, as we used the same speech data as in [7] and [8]. Among them, 25 sets differ in their initial consonants and 25 sets differ in their final consonants. Additional words of the same speaker were recorded for training purposes. Each word is normalized to a unit root mean-square amplitude. The speech-weighted background noise is presented for 1.83 s and is windowed by a Tukey window for smooth onset and offset, where the rise and fall time is set equally to 0.25 s [7], [8]. In each trial, subjects heard up to six acoustic stimuli in a row (but with 0.25 s of pause in between the words) corrupted by one level of background noise chosen randomly from six SNR levels (0, −6, −12, −18, −24, and −30 dB), where each SNR is defined by the power amplitude ratio of the speech signal and noise over the entire word [7]. The target word appears as text on the computer screen and remains visible until termination of the trial.

Subjects have to identify which of the six stimuli is the target word. They hear each stimulus only once and have to press the "SUBMIT" button as soon as they have recognized a stimulus as the target word. Then, the trial is terminated and the next trial is presented. If they think that the stimulus just heard is not the target word, they have to press the "NEXT" button to hear the next stimulus. The whole experiment is composed of one training session for a total of 12 trials and three test sessions (100 trials each) for a total of 300 trials, i.e., 50 sets (one set is composed of six rhyming words) of rhyming words are repeated six times for each SNR condition for each subject. The target words are randomly chosen from those 300 rhyming words. Once a chosen target word is presented, it is removed from the future selection pool. Therefore, the same rhyming word does not occur as target word more than once. A 5-min break is provided to the subjects after every 100 trials.

TABLE I
Numbers of Trials (the Original Speech and the Modified Speech of Various Algorithms) Across Six SNR Levels for Test I

Speech type	SNR (dB)					
	−30	−24	−18	−12	−6	0
Original speech	13	13	13	12	12	12
Modified speech (Joint TF)	13	13	13	12	12	12
Modified speech of [7]	13	13	13	12	12	12
Modified speech of [8]	13	13	13	12	12	12

TABLE II
Numbers of Trials (the Original Speech and the Modified Speech of Various Algorithms) Across Six SNR Levels for Test II

Speech type	SNR (dB)					
	−30	−24	−18	−12	−6	0
Original speech	12	12	12	13	13	13
Modified speech (Joint TF)	12	12	12	13	13	13
Modified speech of [7]	12	12	12	13	13	13
Modified speech of [8]	12	12	12	13	13	13

Eight subjects performed the scenario Test I and the remaining eight subjects performed Test II so as to achieve a perfectly symmetric distribution of the 300 trials across the four types of speech material and the six SNR conditions. The number of trials across SNR levels and algorithms for Test I and Test II are summarized in Tables I and II, respectively.

V. Psychoacoustic Test Results

The average percentage of correct responses for the original speech and the average percentage of correct responses for the modified speech generated from our algorithm, Y's algorithm, and T's algorithm at each SNR level are calculated by the subjects' correct responses divided by the total number of stimuli. The results are illustrated in Fig. 8. Variance bars are omitted to improve readability. Means, standard deviations (SDs), and 95% confidence intervals (CIs) of the paired-sample difference between the modified speech of the joint TF and the original speech at each SNR level are illustrated and summarized in Fig. 9 and Table III, respectively.

The results in Fig. 9 and Table III show that the modified speech of the joint time–frequency segmentation algorithm is recognized better than the original speech at all SNR levels with minimum improvement of 2.68% at −6 dB and maximum improvement of 27.28% at −18 dB. The modified speech significantly improves speech intelligibility in background noise in five of six SNR levels, i.e., 9.58% at 0 dB, 18.59% at −12 dB, 27.28% at −18 dB, 21.51% at −24 dB, and 26.96% at −30 dB, respectively. Specifically, at these SNR levels, the 95% CI differences do not include the value zero.

The paired sample differences between the modified speech of our algorithm and of Y's algorithm are illustrated and summarized in Fig. 10 and Table IV, respectively. The results in

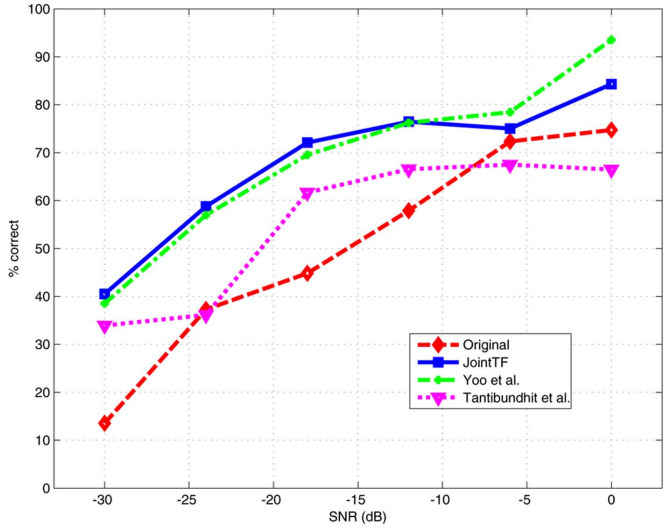


Fig. 8. Average percentage of correct responses for original speech (dashed line), modified speech from joint time–frequency segmentation algorithm (solid line), modified speech from Y's algorithm (dash-dotted line), and modified speech from T's algorithm (dotted line).

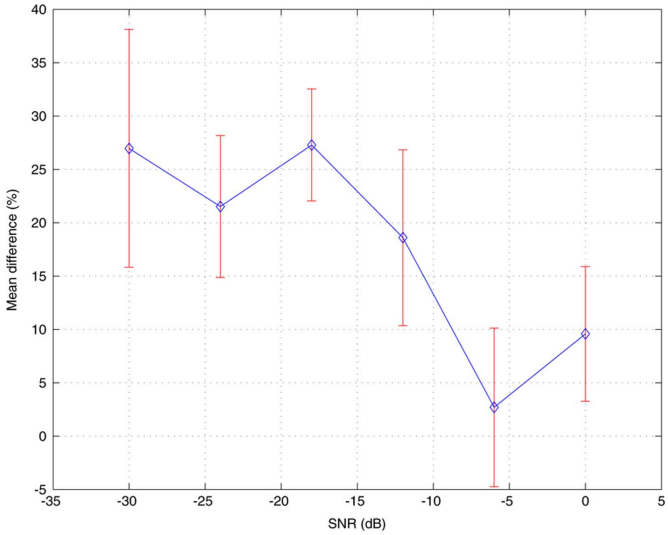


Fig. 9. Mean difference between the modified speech of the joint time–frequency segmentation algorithm and the original speech with 95% confidence interval.

TABLE III
PAIRED SAMPLE DIFFERENCES (BETWEEN THE MODIFIED SPEECH OF JOINT TIME–FREQUENCY SEGMENTATION ALGORITHM AND THE ORIGINAL SPEECH) WITH THEIR MEANS, STANDARD DEVIATIONS (SDS), AND 95% CONFIDENCE INTERVALS (CIS)

SNR	Mean difference	SD	95% CI
−30dB	26.96	20.90	15.82 to 38.10
−24dB	21.51	12.48	14.87 to 28.16
−18dB	27.28	9.86	22.03 to 32.54
−12dB	18.59	15.48	10.34 to 26.84
−6dB	2.68	13.95	−4.75 to 10.12
0dB	9.58	11.85	3.26 to 15.89

Fig. 10 and Table IV show that the modified speech of our algorithm is recognized slightly better than Y's algorithm at low SNR levels, i.e., 0.24% at −12 dB, 2.56% at −18 dB, 1.80% at −24 dB, and 1.96% at −30 dB, respectively. However, the modified speech of our algorithm is recognized worse than Y's algorithm at high SNR levels, i.e., 9.25% at 0 dB and 3.41% at −6 dB, respectively. There is no significant difference between the recognition rates at all SNR levels.

TABLE IV
PAIRED SAMPLE DIFFERENCES (BETWEEN THE MODIFIED SPEECH OF JOINT TIME–FREQUENCY SEGMENTATION ALGORITHM AND OF Y'S ALGORITHM) WITH THEIR MEANS, STANDARD DEVIATIONS (SDS), AND 95% CONFIDENCE INTERVALS (CIS)

SNR	Mean difference	SD	95% CI
−30dB	1.96	21.44	−9.46 to 13.39
−24dB	1.80	19.29	−8.47 to 12.08
−18dB	2.56	15.89	−5.90 to 11.03
−12dB	0.24	17.11	−8.88 to 9.36
−6dB	-3.41	12.37	−10.00 to 3.19
0dB	-9.25	15.00	−17.25 to −1.26

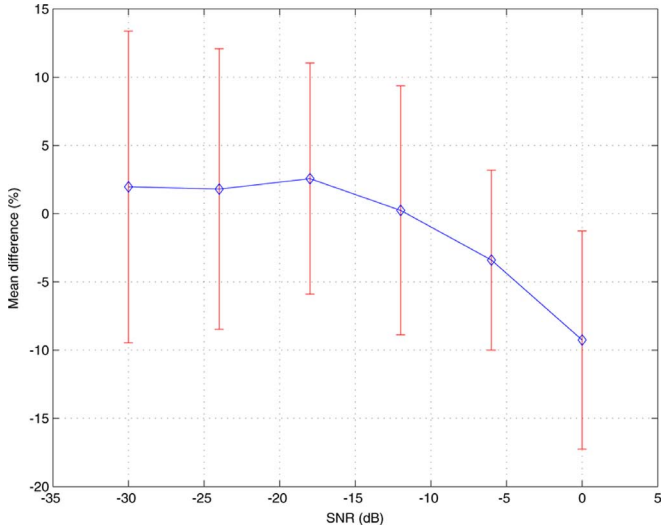


Fig. 10. Mean difference between the modified speech using the joint time–frequency segmentation algorithm and Y's algorithm with 95% confidence interval.

Finally, the paired sample differences between the modified speech of our algorithm and of T's algorithm are shown and summarized in Fig. 11 and Table V, respectively. The results in Fig. 11 and Table V indicate that the modified speech of our algorithm is recognized better than the modified speech of T's algorithm at all SNR levels with minimum improvement of 6.57% at −30 dB and maximum improvement of 22.64% at −24 dB. The modified speech of our algorithm significantly improves speech intelligibility in background noise in four of six SNR levels, i.e., 17.83% at 0 dB, 9.94% at −12 dB, 10.46% at −18 dB, and 22.64% at −24 dB, respectively.

VI. DISCUSSION

The transient component identified by Y's algorithm retains a substantial amount of energy of the formants even for sustained vowel sounds especially in the high-frequency regions. T's algorithm removes most of the energy during the steady-state vowel better than Y's algorithm resulting in some "holes" in the transient component. However, the transient component suffers from pre-echo distortion effects due to the use of the

TABLE V
PAIRED SAMPLE DIFFERENCES (BETWEEN THE MODIFIED SPEECH OF JOINT TIME–FREQUENCY SEGMENTATION ALGORITHM AND OF T'S ALGORITHM) WITH THEIR MEANS, STANDARD DEVIATIONS (SDS), AND 95% CONFIDENCE INTERVALS (CIS)

SNR	Mean difference	SD	95% CI
−30dB	6.57	19.50	−3.82 to 16.96
−24dB	22.64	23.17	10.29 to 34.99
−18dB	10.46	13.63	3.20 to 17.72
−12dB	9.94	10.78	4.19 to 15.68
−6dB	7.53	17.63	−1.86 to 16.93
0dB	17.83	26.36	3.78 to 31.87

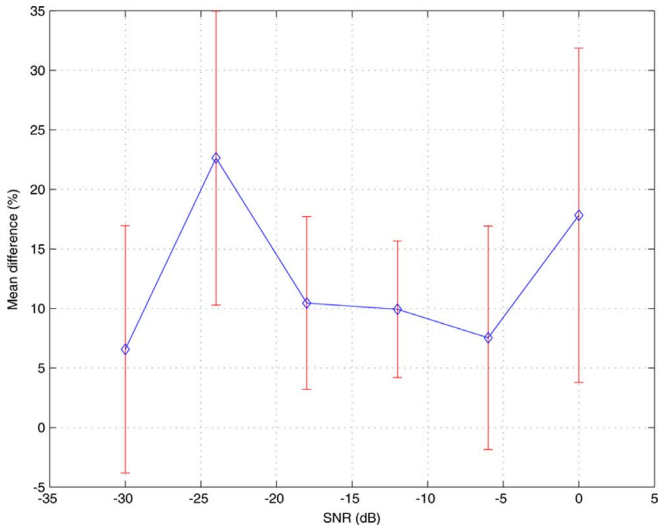


Fig. 11. Mean difference between the modified speech using the joint time–frequency segmentation algorithm and T's algorithm with 95% confidence interval.

MDCT [14]. Moreover, the transient component includes decomposition artifacts, which probably occur during the tonal and transient decomposition [8]. We conclude that the adaptive tiling both in time and frequency of our algorithm provides the best estimate of the transient component compared to the algorithm using three time varying bandpass filters [7] and the algorithm using a fixed time–frequency tiling in the MDCT and the wavelet transform [8].

Word recognition rates of our modified speech in speech weighted background noise are better than those of the original speech at all SNR levels. Among them, for five of six SNR levels except for −6 dB, the modified speech significantly improves speech intelligibility compared to the original speech. The insignificant difference between the modified and the original speech at −6 dB mainly comes from the variability of two subjects, i.e., at this level, they perceived the original speech better then the modified speech 23.08% and 30.77%, respectively. However, the overall results show conclusively that the emphasis of the transient component significantly improves the intelligibility of the speech perception in noise. This suggests that the transient component in speech is important for speech enhancement supporting [7] and [8].

We formally compared the intelligibility of the modified speech identified by our algorithm, Y's algorithm, and T's algorithm. Word recognition rates of our modified speech are better than recognition rates of the modified speech of Y's algorithm at low SNR levels (SNR below −12 dB). Y's algorithm incorporates high-pass filtering at 700 Hz, which has been shown to increase the intelligibility of speech in noise [25], [26]. We presume that the higher recognition rates at 0 and −6 dB of Y's algorithm are due to the effect of increasing the relative power of formant frequency information in high-frequency regions [8]. However, we do not perform a formal experiment to investigate the role of high-frequency regions in the transient component because we expect to achieve a similar result as in [8], where the authors mentioned that the emphasis of high-frequency regions of the transient component increases speech intelligibility [8]. In this paper, we investigate the role of both low- and high-frequency transient activity for improving speech intelligibility without a broad high-frequency emphasis.

Word recognition rates of our modified speech are better than the results of T's algorithm at all SNR levels. We believe that the superior performance of our algorithm is due to the use of the joint time–frequency segmentation. Further, our algorithm does not incorporate a transform coding scheme like the MDCT so as to avoid the pre-echo distortions artifacts. We conclude that our improvements over T's algorithm are at least 7%.

VII. CONCLUSION AND FUTURE WORK

We have developed an algorithm, the joint time–frequency segmentation algorithm, where the tiling is adapted both in time and frequency to the characteristics of the speech signal. Therefore, the Markov modeling is not needed in order to achieve a good quality of the transient speech estimation. The transient information in speech is obtained by using all of the blocks of wavelet packet coefficients, whose tiling heights are larger than or equal to the tiling widths. Although there is no accepted quantitative definition of the transient component of speech [7], [8], we expect the transient component to include abrupt temporal changes, such as onsets and offsets associated with consonants, e.g., /b/, and /t/ in "bat" (/bæt/) and releases of the consonant excluding consonant hubs. In addition, the transient component is expected to include transitions from the consonant to the vowel, within the vowel, and at the end of the vowel to the consonant.

We have used a relative ratio of the tiling height and the tiling width to extract the significant wavelet packet coefficients for the transient component. Yet, we do not consider the use of a tiling ratio as a direct amplification factor to the blocks of wavelet packet coefficients. We suggest that this technique may allow to develop a new dynamic speech enhancement algorithm, where each wavelet packet coefficient is multiplied by a factor depending on the characteristics identified by the tiling ratio. Although in this work we are solely interested to investigate the role of the transient component to enhance speech intelligibility in background noise, we suggest that knowing the noise properties can be useful for enhancing our algorithm.

Our algorithm is an offline algorithm that needs to be applied to one complete block of speech data at a time. We are currently working on the possibility to turn it into an online algorithm, which is capable of processing running speech without blocking artifacts.

The English language is a stress-timed language and the transient component has been shown to be important to increase speech intelligibility in background noise [7], [8], [19]. Other

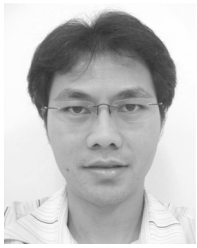
languages have different characteristics of vowel and consonant sounds including differences in speech perception, e.g., tonal languages. We assume that these differences affect the use of our algorithm. We currently investigate the use of our algorithm to enhance speech intelligibility in background noise for tonal languages such as Chinese and Thai.

ACKNOWLEDGMENT

The authors would like to thank S. Yoo for his speech materials and all of the subjects who participated in the modified rhyme test protocol.

REFERENCES

[1] P. C. Loizou, *Speech Enhancement: Theory and Practice*. Boca Raton, FL: CRC, 2007.
[2] Y. Ephraim and H. Trees, "A signal subspace approach for speech enhancement," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 3, no. 4, pp. 251–266, Jul. 1995.
[3] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 2, pp. 113–120, Apr. 1979.
[4] J. Lim and A. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proc. IEEE*, vol. 67, no. 12, pp. 1586–1604, Dec. 1979.
[5] B. Sauert and P. Vary, "Near end listening enhancement: Speech intelligibility improvement in noisy environments," in *Proc. ICASSP*, 2006, pp. 493–496.
[6] V. Hazen and A. Simpson, "The effect of cue-enhancement on the intelligibility of nonsense word and sentence materials presented in noise," *Speech Commun.*, vol. 24, no. 12, pp. 211–226, 1998.
[7] S. Yoo, J. Boston, A. El-Jaroudi, C. Li, J. Durrant, K. Kovacyk, and S. Shaiman, "Speech signal modification to increase intelligibility in noisy environments," *J. Acoust. Soc. Amer.*, vol. 122, no. 2, pp. 1138–1149, 2007.
[8] C. Tantibundhit, J. Boston, C. Li, J. Durrant, S. Shaiman, K. Kovacyk, and A. El-Jaroudi, "New signal decomposition method based speech enhancement," *Signal Process.*, vol. 87, no. 11, pp. 2607–2628, 2007.
[9] W. Strange, J. Jenkins, and T. Johnson, "Dynamic specification of coarticulated vowels," *J. Acoust. Soc. Amer.*, vol. 74, no. 3, pp. 695–705, 1983.
[10] C. Mackersie, A. C. Neuman, and H. Levitt, "A comparison of response time and word recognition measures using a word-monitoring and closed-set identification task," *Ear Hear.*, vol. 20, no. 2, pp. 140–148, 1999.
[11] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.
[12] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk, "Wavelet-based statistical signal processing using hidden Markov models," *IEEE Trans. Signal Process.*, vol. 46, no. 4, pp. 886–902, Apr. 1998.
[13] J. B. Durand, P. Gonçalvès, and Y. Guédon, "Computational methods for hidden Markov tree models—An application to wavelet trees," *IEEE Trans. Signal Process.*, vol. 52, no. 9, pp. 2551–2560, Sep. 2004.
[14] T. Painter, "Perceptual coding of digital audio," *Proc. IEEE*, vol. 88, no. 4, pp. 451–513, Apr. 2000.
[15] C. Herley, Z. X. Xiong, K. Ramchandran, and M. T. Orchard, "Joint space-frequency segmentation using balanced wavelet packet trees for least-cost image representation," *IEEE Trans. Image Process.*, vol. 6, no. 9, pp. 1213–1230, Sep. 1997.
[16] C. M. Thiele and L. F. Villemoes, "A fast algorithm for adapted time–frequency tilings," *J. Appl. Comput. Harmonic Anal.*, vol. 3, no. 2, pp. 91–99, 1996.
[17] C. Tantibundhit and G. Kubin, "Joint time–frequency segmentation for transient decomposition," in *Proc. Interspeech*, 2008, pp. 2502–2505.
[18] R. Coifman and M. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Inf. Theory*, vol. 38, no. 2, pp. 713–718, 1992.
[19] C. Tantibundhit, F. Pernkopf, and G. Kubin, "Speech enhancement based on time–frequency segmentation," in *Proc. ICASSP*, 2009, pp. 4673–4676.
[20] S. Mallat, *A Wavelet Tour of Signal Processing*. San Diego, CA: Academic, 1998.
[21] A. S. House, C. E. Williams, H. M. L. Hecker, and K. D. Kryter, "Articulation-testing methods: Consonantal differentiation with a closed-response set," *J. Acoust. Soc. Amer.*, vol. 37, no. 1, pp. 158–166, 1965.
[22] R. Learned, "Wavelet Packet Based Transient Signal Classification," M.S. thesis, Dept. of Elect. Eng., Mass. Inst. of Technol., Cambridge, 1992.
[23] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.
[24] L. Daudet and B. Torrésani, "Hybrid representation for audiophonic signal encoding," *Signal Process.*, vol. 82, no. 11, pp. 1595–1617, 2002.
[25] N. R. French and J. C. Steinberg, "Factors governing the intelligibility of speech sounds," *J. Acoust. Soc. Amer.*, vol. 19, no. 1, pp. 90–119, 1947.
[26] R. J. Niederjohn and J. H. Grotelueschen, "The enhancement of speech intelligibility in high noise levels by high-pass filtering followed by rapid amplitude compression," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-24, no. 4, pp. 202–207, Aug. 1976.



Charturong Tantibundhit (S'05–M'08) received the B.E. degree in electrical engineering from Kasetsart University, Bangkok, Thailand, in 1996 and the M.S. degree in information science and the Ph.D. degree in electrical engineering from University of Pittsburgh, Pittsburgh, PA, in 2001 and 2006, respectively.

From 1996 to 1998, he was a Project Engineer and an Electrical Engineer at the Siam Cement Group (SCG), Bangkok. From 2003 to 2005, he was a Graduate Student Researcher at the Department of Electrical and Computer Engineering, University of Pittsburgh. Since 2006, he has been with Thammasat University, Pathumthani, Thailand, where he is currently a Lecturer at the Department of Electrical and Computer Engineering. From 2007 to 2008, he was a Postdoctoral Researcher at the Signal Processing and Speech Communication Laboratory (SPSC), Graz University of Technology, Graz, Austria. His research interests include speech enhancement, time–frequency analysis, computer vision, and statistical pattern recognition.

Dr. Tantibundhit received the IEEE ICASSP Student Paper Contest Winners in 2006 and received the ASEA-UNINET Postdoctoral Scholarship Award, Austria, in 2007.

Franz Pernkopf (M'05) received the M.Sc. (Dipl. Ing.) degree in electrical engineering from Graz University of Technology, Graz, Austria, in summer 1999 and the Ph.D. degree from the University of Leoben, Loeben, Austria, in 2002.

He was a Research Associate in the Department of Electrical Engineering, University of Washington, Seattle, from 2004 to 2006. Currently, he is Assistant Professor at the Signal Processing and Speech Communication Laboratory (SPSC), Graz University of Technology. His research interests include machine learning, Bayesian networks, feature selection, finite mixture models, vision, speech, and statistical pattern recognition.

Dr. Pernkopf was awarded the Erwin Schrödinger Fellowship in 2002.

Gernot Kubin (M'84) was born in Vienna, Austria, on June 24, 1960. He received the Dipl.-Ing. and Dr. Techn. (*sub auspiciis praesidentis*) degrees in electrical engineering from Vienna University of Technology (TU Vienna) in 1982 and 1990, respectively.

He has been a Professor of nonlinear signal processing and Head of the Signal Processing and Speech Communication Laboratory (SPSC), Graz University of Technology, Graz, Austria, since September 2000. Earlier international appointments include CERN, Geneva, Switzerland, in 1980; TU Vienna from 1983 until 2000; Erwin Schrödinger Fellowship at Philips Research Laboratories, Eindhoven, The Netherlands, in 1985; AT&T Bell Labs., Murray Hill, NJ, in 1992, 1993, and 1995; KTH Stockholm, Sweden, in 1998; Vienna Telecommunications Research Centre (FTW) from 1999 to present (Key Researcher and Member of the Board); Global IP Sound, Sweden, and USA (Scientific Consultant), in 2000 and 2001; Christian Doppler Laboratory for Nonlinear Signal Processing from 2002 to present (Founding Director). He has authored or coauthored over 80 peer-reviewed publications and three patents.

Dr. Kubin is a Member of the Board of Austrian Acoustics Association and Vice Chair for the European COST Action 277, nonlinear speech processing.