

## Research Article

# Localized Mode DFT-S-OFDMA Implementation Using Frequency and Time Domain Interpolation

Ari Viholainen,<sup>1</sup> Tero Ihalainen,<sup>1</sup> Mika Rinne,<sup>2</sup> and Markku Renfors (EURASIP Member)<sup>1</sup>

<sup>1</sup>Department of Communications Engineering, Tampere University of Technology, P.O. Box 553, 33101 Tampere, Finland

<sup>2</sup>Nokia Research Center, P.O. Box 407, 00045 Helsinki, Finland

Correspondence should be addressed to Ari Viholainen, ari.viholainen@tut.fi

Received 3 September 2008; Revised 19 December 2008; Accepted 12 March 2009

Recommended by Ana Perez-Neira

This paper presents a novel method to generate a localized mode single-carrier frequency division multiple access (SC-FDMA) waveform. Instead of using DFT-spread OFDMA (DFT-S-OFDMA) processing, the new structure called SCiFI-FDMA relies on frequency and time domain interpolation followed by a user-specific frequency shift. SCiFI-FDMA can provide signal waveforms that are compatible to DFT-S-OFDMA. In addition, it provides any resolution of user bandwidth allocation for the uplink multiple access with comparable computational complexity, because the DFT is avoided. Therefore, SCiFI-FDMA allows a flexible choice of parameters appreciated in broadband mobile communications in the future.

Copyright © 2009 Ari Viholainen et al. 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

OFDMA is a multiple access technique that inherits many attractive features of the orthogonal frequency division multiplexing (OFDM) transmission [1, 2]. However, as a multicarrier signal waveform, it suffers from a high peak-to-average power ratio (PAPR) and hence leads to power inefficiency, which has serious consequences for an uplink transmission [3].

Recently single-carrier frequency division multiple access (SC-FDMA) transmission by DFT-spread OFDMA (DFT-S-OFDMA) has drawn increasing attention because it enables frequency-domain equalization (FDE), advanced receiver techniques, and low PAPR [4, 5]. Low PAPR is due to a serially modulated single-carrier block-transmission, where the dynamic range of the transmitted signal's instantaneous power is considerably smaller compared to multicarrier transmission on parallel subcarriers. The lower PAPR reduces the necessary back-off of the nonlinear power amplifier, required to keep the spectral regrowth and in-band distortion at a tolerable level. This property can be exploited to improve the cell coverage or to extend the battery active-time of the mobile terminal.

Alternative techniques to generate the SC-FDMA signal include the DFT-S-OFDMA [6, 7], the DFT-spread

generalized multicarrier (DFT-S-GMC) [8], and the interleaved frequency division multiple access (IFDMA) [9]. DFT-S-GMC is a frequency-domain technique, where the  $M$ -point IFFT is replaced by an  $M$ -band inverse filter bank transform. IFDMA utilizes time-domain generation of the signal waveform using block-wise symbol repetition and a user-specific phase rotation. Another interpretation is that IFDMA is equivalent to a distributed mode DFT-S-OFDMA with equidistant subcarrier mapping.

DFT-S-OFDMA is a very elegant technique to generate the uplink transmission for a specific user. However, the complexity of the DFT and the resulting resolution of the practical DFT sizes are issues for the implementation. Thus, a communications standard like the 3GPP E-UTRA [10] limits the supported subset of the DFT sizes  $K$ . For bandwidth allocation, the standard defines a resource block of 12 subcarriers (i.e.,  $K$  is a multiple of 12 with radix  $\{2, 3, 5\}$ ). The motivation of this paper is to present a generic method to generate SC-FDMA waveform in such a manner that all values of  $K$  are feasible yet with practical complexity. Here, the IFFT size  $M$  is assumed to be a power-of-two but the DFT size  $K$  may take all the values from 1 to  $\lfloor M/\gamma \rfloor$  with  $\gamma$  being a small positive integer. Compared to the reference design, this method enhances flexibility to allocate bandwidth and provides reduced number of multiplications

for large majority of  $K$  values. One drawback inherent in the proposed structure is an approximation error, which however can nicely be parametrized below the noise level.

The rest of this paper is organized as follows. In Section 2, the frequency-domain generation of a single-user signal based on DFT-S-OFDMA is reviewed. Also, the motivation for developing an alternative implementation solution for the localized mode is addressed. Section 3 describes an efficient implementation, where initial frequency-domain interpolation using a specific transform matrix is followed by fractional time-domain interpolation and user-specific frequency translation. Hereafter, this structure is referred to as SC-FDMA implementation using frequency and time domain interpolation (SCiFI-FDMA). The numerical analysis of DFT-S-OFDMA and SCiFI-FDMA are presented in Section 4. The computational complexity is analyzed in terms of the number of required multiplications and additions. Section 5 compares the DFT-S-OFDMA and SCiFI-FDMA techniques to generate an SC-FDMA signal in a single-user scenario and in a multi-user (multiple access) scenario. The performance is compared by evaluating the error vector magnitude (EVM) of the approximation error inherent in the SCiFI-FDMA transmitter due to fractional interpolation. Finally, conclusions are drawn in Section 6.

## 2. Localized Mode DFT-S-OFDMA

DFT-S-OFDMA is a frequency-domain precoding technique to generate the SC-FDMA signal waveform. In this paper, we focus on the properties of the DFT-IFFT processing shown in Figure 1. This structure enables both distributed mode and localized mode symbol mapping by simple change of the subcarrier allocation. Both transmission modes provide signals with an envelope of a single-carrier transmission. This is beneficial in order to minimize the in-band distortion and out-of-band emissions. In general, the localized mode is preferred in the practical systems due to various imperfections of the distributed mode. The distributed mode signal has been shown to be sensitive to the carrier frequency offsets (caused by Doppler effects and/or mismatch of the transmit-receive oscillators), phase noise, and imperfect power control [11]. The localized mode signal is far less sensitive to these imperfections.

*2.1. Signal Model.* In the signal model of Figure 1, the discrete Fourier transformed sequence is expressed as

$$A[k] = \sum_{n=0}^{K-1} x[n] e^{-j((2\pi/K)nk)}, \quad (1)$$

where  $k = 0, 1, \dots, K-1$  and  $x[n]$  is a length- $K$  sequence of symbols, commonly from a QAM alphabet. The subcarrier allocation specifies how the DFT-spread samples are mapped to the frequency bins of the IFFT, that is, whether distributed or localized transmission mode is used. The output of the IFFT is

$$y[m] = \frac{1}{M} \sum_{l=0}^{M-1} B[l] e^{j((2\pi/M)ml)}, \quad (2)$$

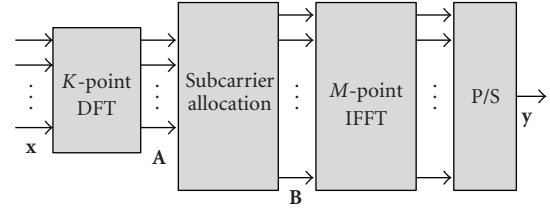


FIGURE 1: Frequency-domain realization of SC-FDMA signal.

where  $m = 0, 1, \dots, M-1$  and  $B[l]$  is the length- $M$  output sequence of the subcarrier allocation block.

In the localized mode, the DFT-spread samples are allocated to a set of contiguous frequency bins of the IFFT. This results in the following subcarrier allocation:

$$B[l] = \begin{cases} A[s], & \text{if } l = (s + \xi) \bmod M, \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

where  $s = 0, 1, \dots, K-1$ ,  $\xi$  is a user-specific subcarrier allocation offset ( $0 \leq \xi \leq M-1$ ), and the mod stands for modulo operation. The input-output relation of the localized mode DFT-S-OFDMA cannot be simplified as much as in the case of the distributed mode and this results in a higher implementation complexity.

*2.2. Implementation Complexity.* In the localized mode, the  $M$ -point IFFT can be efficiently implemented via the Split-Radix FFT algorithm requiring  $M(\log_2(M) - 3) + 4$  real multiplications and  $3M(\log_2(M) - 1) + 4$  real additions [12]. The main challenge is the  $K$ -point DFT with arbitrary values of  $K$ . The direct computation of the DFT requires  $3K^2$  real multiplications. This is because one complex multiplication can be calculated with three real multiplications and three real additions as shown in [12]. A more efficient implementation is possible, if  $K$  can be factorized into a small set of prime numbers. Based on this principle, the number of real multiplications required for the  $K$ -point DFT, can be reduced using the Cooley-Tukey algorithm to  $3K \sum_i k_i l_i$ , where  $K = \prod_i k_i^{l_i}$  [13]. Even more efficient techniques, such as Prime Factor and Winograd Fourier transform algorithms, have been reported in the literature [14]. However, these highly optimized algorithms are not necessarily practical to the SC-FDMA application due to complicated re-indexing of data and increased memory requirements. On the other hand, very low-complexity techniques for a limited set of highly composite DFT sizes up to 1024 have been studied in [15], where the resulting computational complexities have been given in a table format. Later on, these specific values are used for comparison and they are referred to as Murphy's method.

Figure 2 shows the number of required real multiplications when the DFT complexity is estimated using the Cooley-Tukey algorithm and Murphy's method. The different curves indicate the cases where the DFT size is a prime number, a number with a specific radix representation, and a specific value that is feasible in Murphy's method,

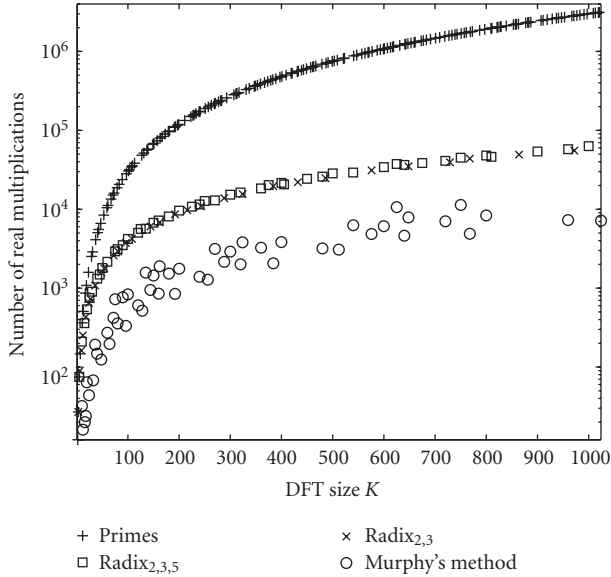


FIGURE 2: DFT complexity using the Cooley-Tukey algorithm and Murphy's method. Radix<sub>2,3,5</sub> means that  $K = 2^{l_1} 3^{l_2} 5^{l_3}$ , where  $l_1 \geq 0$ ,  $l_2 \geq 0$ , and  $l_3 > 0$ .

respectively. As can be seen, the number of required real multiplications is very high, for example, if the DFT size is a prime number. Murphy's method provides a low number of multiplications for a specific set of composite non power-of-two values of  $K$  with factors 2, 3, or 5. However, the number of feasible  $K$  values reported in [15] is only 45 while the DFT size ranges from 10 to 1024.

### 3. SCiFI-FDMA

The motivation for an alternative generic method to generate a localized mode SC-FDMA signal originates from the observations discussed in Section 2.2. It is difficult to find an efficient implementation for arbitrary values of  $K$ . Our idea is to apply a novel processing structure, called SCiFI-FDMA, shown in Figure 3. It is based on the operation by a specific  $M_1 \times K$  transform matrix  $\mathbf{Q}$  ( $M_1 = \gamma K$ , where  $\gamma \in \{2, 3\}$ ), a fractional time-domain interpolation, and a user-specific frequency shift.

The matrix multiplication  $\mathbf{z} = \mathbf{Q}\mathbf{x}$  can be considered as the frequency-domain interpolation by the  $\gamma$ -factor. This results in a limitation, where the  $K$  value can only be varied from 1 to  $\lfloor M/\gamma \rfloor$ . Here, the cases of  $\gamma = 2$  and  $\gamma = 3$  are studied, which may slightly limit the maximum band allocation. The overcoming of this limitation is considered as an additional future study.

The transform matrix  $\mathbf{Q}$  is constructed by allocating the output of the  $K$ -point DFT into the first  $\lceil (K+1)/2 \rceil$  and the last  $\lfloor (K-1)/2 \rfloor$  (here  $\lceil \cdot \rceil$  and  $\lfloor \cdot \rfloor$  denote ceil and floor operations, resp.) bins of the  $M_1$ -point IDFT. This bin allocation results in a very simplified transform matrix leading to computational savings. The time-domain interpolation increases the number of samples from  $M_1$  to the final block length of  $M$  samples. This calls for a

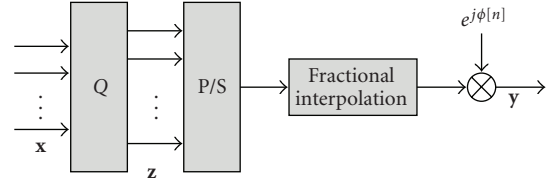


FIGURE 3: An alternative structure (SCiFI-FDMA) for the realization of the localized mode SC-FDMA signal.

fractional interpolation unless  $M_1$  is a power-of-two. Due to the proposed bin allocation, the spectrum of a user is centered around the zero-frequency (exactly around zero for odd values of  $K$  and only half a bin to the right of zero for even values of  $K$ ) resulting in efficient interpolation. The baseband signal format enables fractional interpolation with real-valued anti-imaging filter. The last processing element implements the user-specific frequency shift

$$e^{j\phi[n]} = \frac{K}{M} \exp\left(j \frac{2\pi n}{M} \left(\left\lfloor \frac{K-1}{2} \right\rfloor + \xi\right)\right), \quad (4)$$

where  $n = 0, 1, \dots, M-1$  and  $\xi$  is the subcarrier allocation offset. This frequency shift translates the user spectrum to its scheduled user-specific position in the system band.

**3.1. Transform Matrix.** The transform matrix  $\mathbf{Q}$  defines the input-output relation of the  $K$ -point DFT and the  $M_1$ -point IDFT for the given zero-centered bin allocation. This  $M_1 \times K$  matrix can be expressed as

$$\mathbf{Q} = \gamma \mathbf{W}_{\text{IDFT}} \mathbf{P} \mathbf{W}_{\text{DFT}}, \quad (5)$$

where the elements of the  $u$ th row and the  $v$ th column in the DFT and IDFT matrices are defined as

$$\begin{aligned} [\mathbf{W}_{\text{DFT}}]_{u,v} &= e^{-j(2\pi/K)uv}, \quad u, v = 0, 1, \dots, K-1, \\ [\mathbf{W}_{\text{IDFT}}]_{u,v} &= \frac{1}{M_1} e^{j(2\pi/M_1)uv}, \quad u, v = 0, 1, \dots, M_1-1, \end{aligned} \quad (6)$$

respectively. Moreover,  $\mathbf{P}$  denotes an  $M_1 \times K$  expansion and permutation matrix that controls the selection of active bins.

The transform matrix  $\mathbf{Q}$  has an efficient structure when matrix  $\mathbf{P}$  is selected according to

$$\mathbf{P} = \begin{bmatrix} \mathbf{e}_1^T \\ \vdots \\ \mathbf{e}_n^T \\ \text{---} \\ \mathbf{0}_{(M_1-K) \times K} \\ \text{---} \\ \mathbf{e}_{n+1}^T \\ \vdots \\ \mathbf{e}_K^T \end{bmatrix}, \quad (7)$$

where  $\mathbf{0}$  is the  $(M_1 - K) \times K$  zero matrix,  $n = \lceil (K + 1)/2 \rceil$ , and  $\mathbf{e}_i = [0 \cdots 0 1 0 \cdots 0]^T$  denotes the  $i$ th  $K \times 1$  natural basis vector (with one as the  $i$ th element).

With the given IDFT bin allocation, the  $\mathbf{Q}$  matrix consists of  $\gamma$  interlaced  $K \times K$  circulant matrices. Submatrices  $\mathbf{R}_i$ , for  $i = 0, 1, \dots, \gamma - 1$ , can be extracted from the  $\mathbf{Q}$  matrix by picking up every  $\gamma$ th row (from top to down) with offsets of  $i$  rows, that is,

$$[\mathbf{R}_i]_{u,v} = [\mathbf{Q}]_{\gamma u+i,v}, \quad u, v = 0, 1, \dots, K - 1. \quad (8)$$

Furthermore, each of these circulant matrices is fully characterized by its first column vector  $\mathbf{r}_i = \mathbf{R}_i \mathbf{e}_1$ . The other column vectors are obtained as rotations of  $\mathbf{r}_i$  [16]. As an example, the structure of a  $3K \times K$  ( $\gamma = 3$ ) transform matrix is shown below:

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ a_0 & a_1 & \cdots & a_{K-2} & a_{K-1} \\ b_0 & b_1 & \cdots & b_{K-2} & b_{K-1} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 1 & \cdots & 0 & 0 \\ a_{K-1} & a_0 & \cdots & a_{K-3} & a_{K-2} \\ b_{K-1} & b_0 & \cdots & b_{K-3} & b_{K-2} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ & & \ddots & & \ddots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 & 1 \\ a_1 & a_2 & \cdots & a_{K-1} & a_0 \\ b_1 & b_2 & \cdots & b_{K-1} & b_0 \end{bmatrix}. \quad (9)$$

The coefficients on the second row of  $\mathbf{Q}$  matrix can be expressed as  $a_v = [R_1]_{0,v}$  for  $v = 0, 1, \dots, K - 1$ , whereas the coefficients on the third row of  $\mathbf{Q}$  matrix are  $b_v = [R_2]_{0,v}$ . The rest of the coefficients are determined by cyclic rotations according to the definition of a circular matrix.

**3.1.1. Efficient Implementation of Matrix Multiplication.** Let us now consider the transformed vector  $\mathbf{z}$ . It comprises of  $\gamma$  interlaced subvectors  $\mathbf{z}_i$  (meaning that  $z(\gamma u + i) = z_i(u)$ ), each of which is the outcome of the matrix multiplication  $\mathbf{z}_i = \mathbf{R}_i \mathbf{x}$ . The observation that a matrix multiplication  $\mathbf{z}_i = \mathbf{R}_i \mathbf{x}$  can be identified as a circular (cyclic) convolution between  $\mathbf{r}_i$  and  $\mathbf{x}$  leads to an efficient frequency-domain realization. The circular convolution is the IDFT of the product of the DFTs of two length- $K$  vectors [17], that is,

$$\mathbf{z}_i = \mathbf{r}_i \odot \mathbf{x} = \text{IDFT}(\text{DFT}(\mathbf{r}_i) \cdot \text{DFT}(\mathbf{x})). \quad (10)$$

However, we target to replace the IDFT (DFT) by the IFFT (FFT) because of the reduced complexity. If  $K$  is a power-of-two then this follows straightforwardly. For other values of  $K$ , vectors  $\mathbf{r}_i$  and  $\mathbf{x}$  are zero-padded (by adding trailing zeros) to form the length- $K_1$  vectors  $\tilde{\mathbf{r}}_i$  and  $\tilde{\mathbf{x}}$ , where  $K_1$  is the smallest

power-of-two value greater than  $2K - 1$ . (Another option is to use length- $K_2$  DFT, where  $K_2$  is the smallest integer value greater than  $2K - 1$ , when it can be implemented more efficiently using the Cooley-Tukey algorithm or Murphy's method). Now,

$$\tilde{\mathbf{z}}_i = \text{IFFT}(\text{FFT}(\tilde{\mathbf{r}}_i) \cdot \text{FFT}(\tilde{\mathbf{x}})) \quad (11)$$

represents actually linear convolution instead of circular convolution. Fortunately, there is a relationship between circular convolution and linear convolution as mentioned in [17]. Therefore, we can obtain circular convolution by using the following equation:

$$\mathbf{z}_i(l) = \tilde{\mathbf{z}}_i(l) + \tilde{\mathbf{z}}_i(l + K), \quad l = 0, 1, \dots, K - 1. \quad (12)$$

Figure 4 summarizes the efficient implementation of the matrix multiplication  $\mathbf{z} = \mathbf{Q}\mathbf{x}$ . The first branch is simpler than the others because the first submatrix  $\mathbf{R}_0$  is an identity matrix. For other branches, the input vector is first zero-padded to a length of  $K_1$  samples and transformed to the frequency domain. Then, the resulting vector is multiplied element-wise by a length- $K_1$  vector  $\Phi_i$ , which can be obtained by the FFT of  $\tilde{\mathbf{r}}_i$ , and the result is transformed back to the time domain. The L2C-block denotes the conversion from linear convolution to circular convolution. Finally, the vectors  $\mathbf{z}_i$  are interlaced to form the output vector  $\mathbf{z}$ . This is done by using upsamplers and a delay-chain like in a typical polyphase implementation.

It should be pointed out that the coefficients of the vectors  $\Phi_i$ , where  $i = 1, 2, \dots, \gamma - 1$ , can be pre-calculated for each desired combination of  $K$  and  $\gamma$  parameters and stored in a look-up table for run-time access.

**3.2. Fractional Interpolation.** Fractional interpolation can be performed straightforwardly by using a cascade of upsampling by  $L$ , image-rejection/anti-alias filtering, and downsampling by  $D$  [18]. However, this approach is not the most practical one because it requires a large amount of unnecessary calculations since only every  $D$ th sample is finally preserved. A more efficient technique for fractional interpolation is to use polynomial-based interpolation filters, that is, filters having a piece-wise polynomial impulse response.

The modified Farrow structure, shown in Figure 5, provides an attractive realization for polynomial-based interpolation filters [19]. The input signal is filtered with  $M_p + 1$  parallel real-valued symmetric/antisymmetric FIR filters of order  $N$  (with coefficients  $c_{m,n}$ , where  $m = 0, 1, \dots, M_p$  and  $n = 0, 1, \dots, N$ ), to obtain the output samples of the branch filters  $C_m(z)$ . The interpolated sample value is obtained by multiplying output samples by so-called basis multipliers  $(\alpha(\mu))^m = (2\mu - 1)^m$ , for  $m = 0, 1, \dots, M_p$ , and summing all up. Here,  $M_p$  defines the polynomial-order of the interpolation filter and  $\alpha(\mu)$  is the continuous-valued parameter, so-called fractional interval, that controls the time difference between the desired time instant for the interpolated output sample and the previous time instant where the discrete-time input sample exists.

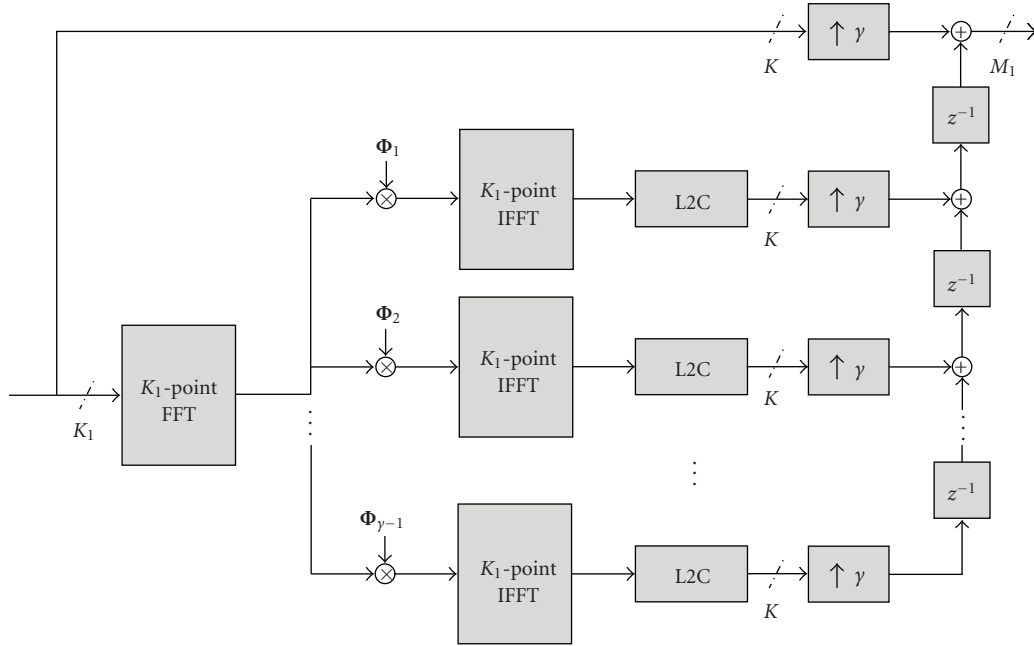


FIGURE 4: Efficient implementation of the matrix multiplication  $\mathbf{z} = \mathbf{Q}\mathbf{x}$  is based on the cyclic convolution in the frequency domain.

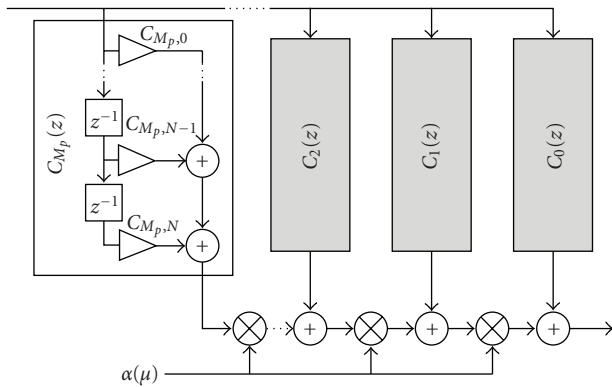


FIGURE 5: Modified Farrow structure.

**3.2.1. Simplified Version of Modified Farrow Structure.** The parameters of the modified Farrow structure could be tuned for different pairs of  $M_1$  and  $M$  in order to obtain the best tradeoff between performance and complexity. However, we have observed that the polynomial-order of two,  $M_p = 2$ , provides sufficiently good performance with arbitrary values of  $M_1$  even if the length of the branch filters is kept relatively short, that is,  $N + 1 \leq 10$ . Naturally, the branch filter coefficients have to be pre-optimized and stored in a look-up table for every pair of  $M_1$  and  $M$  for run-time execution.

In order to guarantee interpolation where the incoming values are preserved and new values are generated between the original ones, the piece-wise polynomial impulse response should form a Nyquist filter. This Nyquist property results in an additional simplification in the modified Farrow

structure because now different branch filters have a special relationship. If the  $m$ th branch filter is written as

$$C_m(z) = \sum_{n=0}^N c_{m,n} z^{-n}, \quad (13)$$

the filter coefficients are expressed as follows:

$$c_{2,n} = c_{2,N-n},$$

$$c_{1,n} = \begin{cases} 0.5, & \text{if } n = \frac{N-1}{2}, \\ -0.5, & \text{if } n = \frac{N+1}{2}, \\ 0, & \text{otherwise,} \end{cases} \quad (14)$$

$$c_{0,n} = \begin{cases} 0.5 - c_{2,n}, & \text{if } n = \frac{N-1}{2} \text{ or } n = \frac{N+1}{2}, \\ -c_{2,n}, & \text{otherwise.} \end{cases}$$

Basically, there are only  $(N + 1)/2$  coefficients,  $c_{2,n}$  for  $n = 0, 1, \dots, (N - 1)/2$ , to be optimized. Moreover, it is quite easy to optimize these coefficients in such a manner that they directly have a sum-of-power-of-two representation. This means that a multiplication can be replaced by simple shifts and additions. The actual optimization of the filter coefficients is out of the scope of this paper, however, an interested reader can find more details in [20, 21] and references therein.

## 4. Computational Complexity

The computational complexity of DFT-S-OFDMA and SciFI-FDMA are evaluated by calculating the number of real



multiplications ( $\theta$ ) that is required to compute length- $M$  complex-valued output sequence. The number of real additions ( $\phi$ ) is also roughly estimated although multiplications are dominant with regard to the complexity. DFT-S-OFDMA consists of the  $K$ -point DFT, subcarrier mapping, and the  $M$ -point IFFT. The subcarrier mapping does not require any multiplications, so the complexity results from the transform blocks. The total number of real multiplications for the DFT-S-OFDMA structure (assuming the Cooley-Tukey and the Split-Radix algorithms) is

$$\theta_R = \theta_{\text{DFT}} + \theta_{\text{IFFT}}, \quad (15)$$

where

$$\theta_{\text{DFT}} = \left( K \sum_i k_i l_i \right) \cdot 3, \quad K = \prod_i k_i^{l_i}, \quad (16)$$

$$\theta_{\text{IFFT}} = M(\log_2(M) - 3) + 4.$$

The total number of real multiplications for SCiFI-FDMA is the sum of the multiplications in each processing block; the matrix multiplication ( $\theta_{\text{MM}}$ ), the modified Farrow structure ( $\theta_{\text{FA}}$ ), and the user-specific frequency shift ( $\theta_{\text{FS}}$ ):

$$\theta_S = \theta_{\text{MM}} + \theta_{\text{FA}} + \theta_{\text{FS}}, \quad (17)$$

where

$$\theta_{\text{MM}} = \gamma(K_1(\log_2(K_1) - 3) + 4) + (\gamma - 1)K_1 \cdot 3,$$

$$\theta_{\text{FA}} = M_1 \frac{N+1}{2} \cdot 2 + 2M \cdot 2, \quad (18)$$

$$\theta_{\text{FS}} = M \cdot 3.$$

The efficient implementation of the matrix multiplication consists of one FFT and  $\gamma - 1$  IFFTs. In addition,  $\gamma - 1$  element-wise length- $K_1$  vector multiplications are required. The modified Farrow structure only consists of  $(N + 1)/2$  different branch filter coefficients (if trivial multiplications by 0.5 are omitted) and base multipliers  $\alpha(\mu)$ . The frequency shift requires  $M$  complex multiplications.

In the case of DFT-S-OFDMA, the number of real additions is calculated using

$$\phi_R = \phi_{\text{DFT}} + \phi_{\text{IFFT}}, \quad (19)$$

where

$$\phi_{\text{DFT}} = \left( K \sum_i k_i l_i \right) \cdot 3, \quad K = \prod_i k_i^{l_i}, \quad (20)$$

$$\phi_{\text{IFFT}} = 3M(\log_2(M) - 1) + 4.$$

For SCiFI-FDMA

$$\phi_S = \phi_{\text{MM}} + \phi_{\text{FA}} + \phi_{\text{FS}}, \quad (21)$$

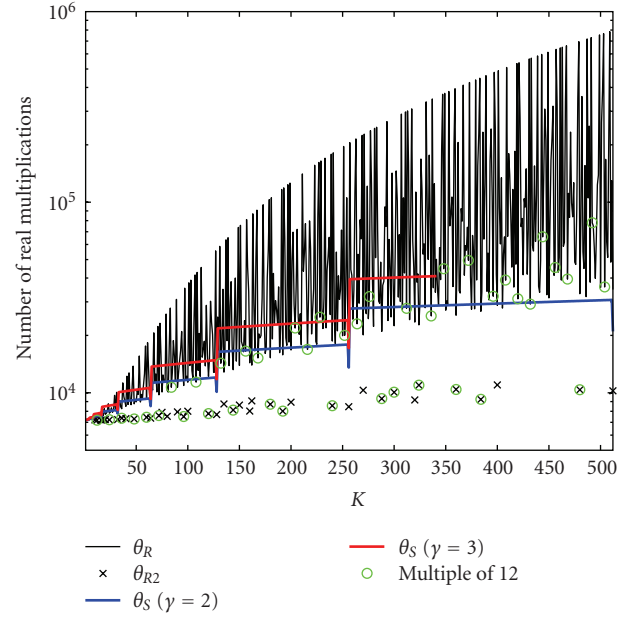


FIGURE 6: Number of real multiplications for DFT-S-OFDMA ( $\theta_R$  and  $\theta_{R2}$ ) and SCiFI-FDMA ( $\theta_S$  ( $\gamma = 2$ ) and  $\theta_S$  ( $\gamma = 3$ )).

where

$$\begin{aligned} \phi_{\text{MM}} &= \gamma(3K_1(\log_2(K_1) - 1) + 4) + (\gamma - 1)K_1 \cdot 3 \\ &\quad + (\gamma - 1)(K - 1) \cdot 2, \end{aligned} \quad (22)$$

$$\phi_{\text{FA}} = M_1 N \cdot 2 + M_1 3 \cdot 2 + M_p M \cdot 2,$$

$$\phi_{\text{FS}} = M \cdot 3.$$

The following set of parameters is used for the numerical complexity analysis:  $M = 1024$ ,  $K = 1, 2, \dots, \lfloor M/\gamma \rfloor$  with  $\gamma \in \{2, 3\}$ ,  $K_1$  is the next power-of-two value greater than  $2K - 1$ ,  $M_1 = \gamma K$ ,  $M_p = 2$ , and  $N = 5$ . The effect of the  $\gamma$ -factor on the performance will be discussed in Section 5. Figures 6 and 7 show the number of required real multiplications and additions as a function of increasing  $K$ , respectively. The complexities of DFT-S-OFDMA ( $\theta_R$ ,  $\phi_R$ ) and SCiFI-FDMA ( $\theta_S$ ,  $\phi_S$ ) are calculated using (15)–(22). In addition, a number of discrete points ( $\theta_{R2}$ ,  $\phi_{R2}$ ) indicate the DFT-S-OFDMA complexity when the DFT part is estimated using Murphy's table given in [15]. As for the  $\theta_R$  and  $\theta_{R2}$ , the points that correspond to multiples of 12 are indicated by circle markers.

As can be seen, the complexity of DFT-S-OFDMA is a strongly fluctuating function of  $K$  when the performance over the whole range of  $K = 1, 2, \dots, \lfloor M/\gamma \rfloor$  is considered. Clearly, there are tempting values of  $K$  that yield low complexity, whereas, for example, prime values of  $K$  result in overwhelming complexity. On the other hand, SCiFI-FDMA provides a solution that adds flexibility by allowing moderate and smooth complexity over the whole range of  $K$  values. If  $K$  is a power-of-two, then the term  $\theta_{\text{MM}}$  in (18) is evaluated by substituting  $K$  for  $K_1$ . This results in the downward pointing spikes in Figure 6. The number of

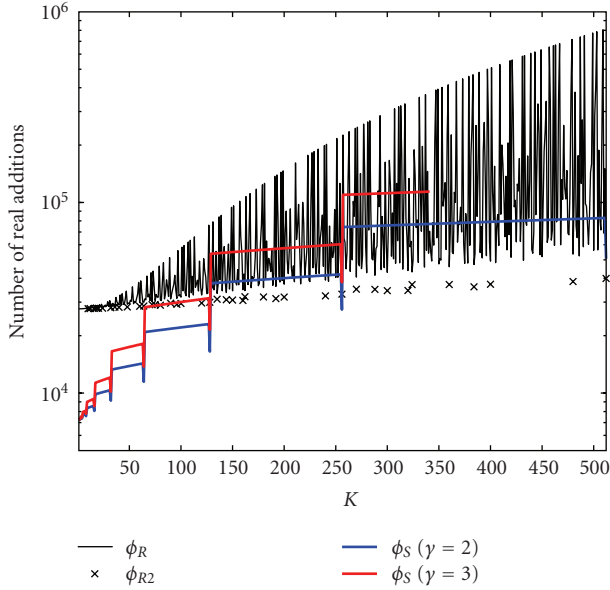


FIGURE 7: Number of real additions for DFT-S-OFDMA ( $\phi_R$  and  $\phi_{R2}$ ) and SCiFi-FDMA ( $\phi_S$  ( $\gamma = 2$ ) and  $\phi_S$  ( $\gamma = 3$ )).

additions can be quite high for larger values of  $K$  but this is typically not considered as a problem, because adders are less costly to implement than multipliers.

Table 1 compares the number of real multiplications of DFT-S-OFDMA and SCiFi-FDMA for three sets of  $K$  values. The first set (multiples of 12 with radix  $\{2, 3, 5\}$ ) clearly favors the DFT-S-OFDMA implementation. As for the second set (other multiples of 12), the relative performance depends on the  $K$  value considered and the difference in complexity fluctuates (+/-) for the benefit of either structure. The third set of arbitrarily chosen points shows the potential of the SCiFi-FDMA structure. In general, a small set of points that favors either DFT-S-OFDMA or SCiFi-FDMA can easily be chosen. Therefore, it is necessary to consider the performance over the full range of  $K = 1, 2, \dots, \lfloor M/\gamma \rfloor$ . The SCiFi-FDMA structure is shown to provide lower complexity for 52% ( $\gamma = 3$ ) and 81% ( $\gamma = 2$ ) of cases over the full set of  $K$  values. When the branch filter coefficients of the modified Farrow structure have a sum-of-power-of-two representation these numbers increase to 63% ( $\gamma = 3$ ) and 89% ( $\gamma = 2$ ), respectively.

Regarding the memory consumption of a practical implementation, it should be noted that the following components can be stored in a memory for the run-time access for each value of  $K$ :

- (i)  $\gamma - 1$  vectors  $\Phi_i$  of length- $K_1$ ,
- (ii)  $(N + 1)/2$  pre-optimized Farrow coefficients  $c_{2,m}$ .

## 5. Performance Evaluation

In this section, we compare the DFT-S-OFDMA and SCiFi-FDMA techniques for implementing an equivalent

TABLE 1: Number of required multiplications for specific values of  $K$  for DFT-S-OFDMA ( $\theta_{R/R2}$ ) and SCiFi-FDMA ( $\theta_S$  ( $\gamma = 2$ )).

	$K$	$\theta_{R/R2}$	$\theta_S$	$\theta_S - \theta_{R/R2}$
Set 1:	270	10316	27824	17508
	288	9320	28040	18720
	324	10964	28472	17508
	384	9232	29192	19960
Set 2:	276	32012	27896	-4116
	312	27764	28328	564
	336	25316	28616	3300
	396	32120	29336	-2784
Set 3:	282	51164	27968	-23196
	302	145790	28208	-117582
	321	113102	28436	-84666
	344	57740	28712	-29028

SC-FDMA uplink transmission. The comparison is performed by evaluating the approximation error introduced by the SCiFi-FDMA transmitter processing. The impact of the approximation error is studied both in a single-user case and in a multi-user case.

*5.1. Single-User Case.* We begin the analysis by considering the approximation error in a single-user case. The relevant SCiFi-FDMA parameters for this numerical example are as follows:

$$\begin{aligned} M &= 1024, & K &= 101, & \xi &= 65, \\ N &= 5, & M_p &= 2, & \gamma &\in \{2, 3\}. \end{aligned} \quad (23)$$

The influence of the approximation error is analyzed through the detection of a SCiFi-FDMA synthesized uplink signal at the receiver side of the link. Here, the signal transmission is assumed to be ideal in a sense that the effects of channel distortion and additive noise are not considered and perfect time synchronization is assumed. Moreover, the receiver processing is based on the reference structure (DFT-S-OFDMA) consisting of the  $M$ -point FFT, subcarrier selection, and the  $K$ -point IDFT. Therefore, the potential non-idealities of the SCiFi-FDMA processing form the only source of errors in the considered example. Figures 8 and 9 show the received signal constellation obtained using the SCiFi-FDMA transmitter with initial frequency-domain upsampling factor of  $\gamma = 2$  and  $\gamma = 3$ , respectively. In the case of DFT-S-OFDMA, the received symbol estimates coincide with the ideal constellation points, whereas for SCiFi-FDMA they disperse slightly around the ideal points due to the approximation error introduced by the time-domain fractional interpolation.

The error vector magnitude (EVM) is a well-defined and widely adopted metric to measure the signal quality/purity.

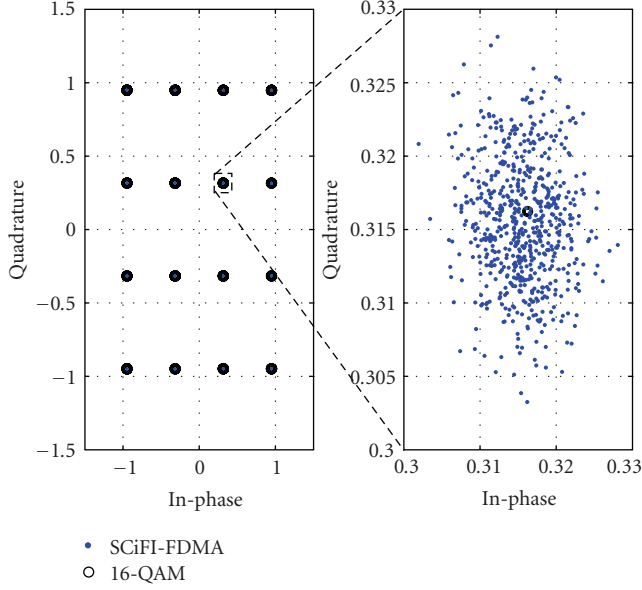


FIGURE 8: Dispersion of the received signal constellation due to fractional interpolation ( $\gamma = 2$ ).

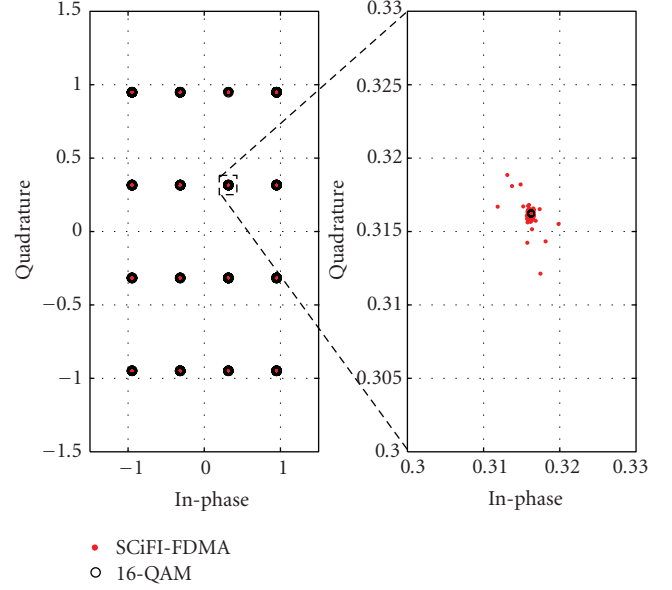


FIGURE 9: Dispersion of the received signal constellation due to fractional interpolation ( $\gamma = 3$ ).

In order to estimate the average signal distortion due to the approximation error, the EVM is evaluated according to [22]:

$$\text{EVM} = \sqrt{\frac{E\{|\tilde{x}[n] - x[n]|^2\}}{E\{|x[n]|^2\}}} \approx \sqrt{\frac{1/N_s \sum_{n=1}^{N_s} |\tilde{x}[n] - x[n]|^2}{1/N_s \sum_{n=1}^{N_s} |x[n]|^2}}, \quad (24)$$

where  $E\{\cdot\}$ ,  $\tilde{x}[n]$ ,  $x[n]$ , and  $N_s$  denote the expectation of ensemble averages, the actual (measured) and the ideal symbols, and the length of the symbol sequence, respectively. Moreover, the mean-squared error is normalized by the average power of the ideal signal. It should be emphasized that the level of EVM can be controlled by adjusting the SCiFI-FDMA parameters  $\gamma$ ,  $M_p$ , and  $N$ . This allows different complexity-performance trade-offs in the actual system design. Figure 10 shows the evaluated (average) EVM for SCiFI-FDMA with varying DFT size  $K$ . It can be observed that the resulting EVM of SCiFI-FDMA is below  $-40$  dB or  $-52$  dB, over the whole range of  $K$  values, for  $\gamma = 2$  and  $\gamma = 3$ , respectively. Therefore, as the estimated level of EVM is well below the level of thermal noise encountered in practise, the BER performance would dominantly be determined by the SNR operation point instead of the signal dispersion by the fractional interpolation.

**5.2. Multi-User Case.** In a multi-user case, the other users can be considered as possible sources of multiple access interference (MAI) due to non-ideal spectral nulls of SCiFI-FDMA synthesized signals. MAI degrades the detection performance of a specific uplink signal, thus it was numerically estimated from the received compound signal at the base station receiver. A multiple access reception of ten simultaneous uplink users with consecutive allocations in the signal band (with neighboring, non-overlapping frequency

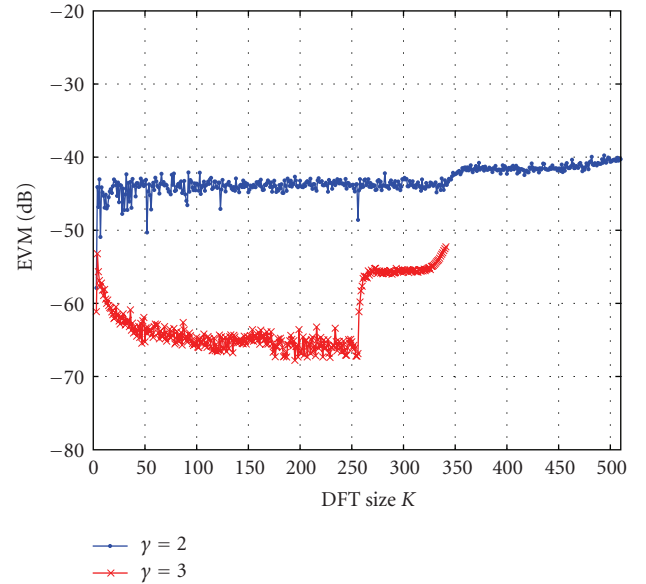


FIGURE 10: EVM over a range of the DFT size  $K$ .

bins) was considered. Furthermore, the uplink transmission was assumed to be ideal both in timing and power control.

From the single-user detection point of view, the other uplink users can be seen as additive noise sources. In order to estimate the variance of MAI, the mean-squared error (MSE) can be estimated, at the frequency bins allocated to a selected user being detected at a given time, while there are transmissions on the rest of the frequency bins allocated for the rest of the uplink users. The average MSE was estimated over a set of one hundred random MA profiles (each with a randomly picked sequence of the DFT sizes and modulation



orders of  $\{4, 16, 64\}$  QAM alphabets for all uplink users). Moreover, all the considered MA profiles were full bandwidth scenarios, that is, the DFT sizes allocated for the ten users summed up to the bandwidth of the IFFT size  $M$ . As a result, the level of the additive MAI was estimated to be  $-42$  dB and  $-52$  dB for the design with  $\gamma = 2$  and  $\gamma = 3$ , respectively.

## 6. Conclusions

In this paper, SCiFI-FDMA was proposed as a potential implementation structure for the wideband uplink transmission in a future communication system. SCiFI-FDMA is based on frequency and time domain interpolation and a user-specific frequency shift. It was shown that the SCiFI-FDMA structure is able to generate signal waveforms comparable to those obtained with DFT-S-OFDMA. The main advantages of SCiFI-FDMA are its enhanced flexibility to the generic choice of allocated bandwidth per user and its competitive computational complexity.

In this paper, the performance was analyzed using experimentally chosen parameter values to satisfy the expected requirements of a communication system. Naturally, the parameters can further be fine-tuned and filters re-optimized depending on the targeted performance. Based on its characteristics, the SCiFI-FDMA offers attractive trade-offs for the synthesis of SC-FDMA waveforms.

## Acknowledgments

This research was supported by Nokia (project Waveform Analysis for Cellular Systems). Moreover, Tero Ihalainen would like to thank Tampere Graduate School in Information Science and Engineering (TiSE) for financial support during this research.

## References

- [1] 3GPP TR 25.814 V7.0.0, "Physical layer aspects for evolved UTRA," (Release 7), June 2006.
- [2] R. van Nee and R. Prasad, *OFDM for Wireless Multimedia Communications*, Artech House, London, UK, 2000.
- [3] H. G. Myung, J. Lim, and D. J. Goodman, "Single carrier FDMA for uplink wireless transmission," *IEEE Vehicular Technology Magazine*, vol. 1, no. 3, pp. 30–38, 2006.
- [4] H. Sari, G. Karam, and I. Jeanclaude, "Transmission techniques for digital terrestrial TV broadcasting," *IEEE Communications Magazine*, vol. 33, no. 2, pp. 100–109, 1995.
- [5] D. Falconer, S. L. Ariyavisitakul, A. Benyamin-Seeyar, and B. Eidson, "Frequency domain equalization for single-carrier broadband wireless systems," *IEEE Communications Magazine*, vol. 40, no. 4, pp. 58–66, 2002.
- [6] D. Galda and H. Rohling, "A low complexity transmitter structure for OFDM-FDMA uplink systems," in *Proceedings of the 55th IEEE Vehicular Technology Conference (VTC '02)*, vol. 4, pp. 1737–1741, Birmingham, Ala, USA, May 2002.
- [7] R. Dinis, D. Falconer, C. T. Lam, and M. Sabbaghian, "A multiple access scheme for the uplink of broadband wireless systems," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '04)*, vol. 6, pp. 3808–3812, Dallas, Tex, USA, November-December 2004.
- [8] X. Zhang, M. Li, H. Hu, H. Wang, B. Zhou, and X. You, "DFT spread generalized multi-carrier scheme for broadband mobile communications," in *Proceedings of the 17th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '06)*, pp. 1–5, Helsinki, Finland, September 2006.
- [9] T. Frank, A. Klein, E. Costa, and E. Schulz, "Robustness of IFDMA as air interface candidate for future high rate mobile radio systems," *Advances in Radio Science*, vol. 3, pp. 265–270, 2005.
- [10] 3GPP TSG-RAN WG1 #47, "R1-063127: DFT size for uplink transmissions," November 2006.
- [11] T. Pollet, M. Van Bladel, and M. Moeneclaey, "BER sensitivity of OFDM systems to carrier frequency offset and Wiener phase noise," *IEEE Transactions on Communications*, vol. 43, no. 234, pp. 191–193, 1995.
- [12] H. S. Malvar, *Signal Processing with Lapped Transforms*, Artech House, Boston, Mass, USA, 1992.
- [13] R. E. Blahut, *Fast Algorithms for Digital Signal Processing*, Addison-Wesley, Reading, Mass, USA, 2nd edition, 1987.
- [14] S. K. Mitra and J. F. Kaiser, *Handbook for Digital Signal Processing*, John Wiley & Sons, New York, NY, USA, 1993.
- [15] C. D. Murphy, "Low-complexity FFT structures for OFDM transceivers," *IEEE Transactions on Communications*, vol. 50, no. 12, pp. 1878–1881, 2002.
- [16] G. H. Golub and C. F. van Loan, *Matrix Computations*, The Johns Hopkins University Press, London, UK, 3rd edition, 1996.
- [17] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1989.
- [18] R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1983.
- [19] J. Vesma and T. Saramäki, "Interpolation filters with arbitrary frequency response for all-digital receivers," in *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS '96)*, vol. 2, pp. 568–571, Atlanta, Ga, USA, May 1996.
- [20] J. Vesma, *Optimization and applications of polynomial-based interpolation filters*, Doctoral dissertation, Tampere University of Technology, Tampere, Finland, May 1999.
- [21] J. Vesma, "A frequency-domain approach to polynomial-based interpolation and the farrow structure," *IEEE Transactions on Circuits and Systems II*, vol. 47, no. 3, pp. 206–209, 2000.
- [22] Q. Gu, *RF System Design of Transceivers for Wireless Communications*, Springer, New York, NY, USA, 2005.