APPROXIMATION AND RESAMPLING OF TAPPED DELAY LINE CHANNEL MODELS WITH GUARANTEED CHANNEL PROPERTIES

Christian Mehlführer and Markus Rupp

Institute of Communications and Radio-Frequency Engineering, Vienna University of Technology
Gusshausstrasse 25/389, A-1040 Vienna, Austria
Email: {chmehl, mrupp}@nt.tuwien.ac.at

ABSTRACT

In this paper, we present a novel framework for resampling and complexity reduction of tapped delay line channel models. In contrast to related algorithms in this field, our framework guarantees that the total impulse response power, the mean delay, as well as the RMS delay spread of every input impulse response remains unchanged if a solution is found. The phase angles of the output impulse response are chosen such that the magnitude error of the channel realization in the frequency domain is minimized as well.

***Index Terms*—** channel modeling, channel emulation

1. INTRODUCTION

Real-time channel emulators are required for testing receiver implementations of communication systems. When building a real-time channel emulator working in the digital baseband domain only, usually two problems occur:

1. Since channel emulators should support a multitude of different channel models, the channel emulator's sampling frequency may not be the same or a multiple of the various channel model sampling frequencies. This requires either resampling of transmit and receive signals (which can be of very high complexity due to rational resampling factors), or resampling of impulse responses. The resampling of the impulse response can be performed offline in advance but results in spreading of a single tap energy to multiple taps. Even a small set of (causal) taps results in an infinite set of (non causal) taps by sinc interpolation. This increased number of taps requires a lot of additional computational power that would not be required if the channel emulator can work at the original channel model's sampling frequency.
2. The computational resources and the clock frequency of the channel emulator hardware define the maximum number of paths (taps), the maximum system bandwidth, and the maximum number of channels (e.g. several channels of one MIMO channel) that can be emulated in real-time. To utilize the available resources as best as possible, the number of taps considered by the emulator should be as small as possible while representing the original impulse response with high accuracy.

Because of these reasons, algorithms for reducing the number of taps of an impulse response are necessary. Of course, such algorithms have to preserve specific properties of the input impulse response that affect the performance of the communication system. For example, if the RMS delay spread of the channel is not preserved correctly this results in a different bit error floor due to delay dispersion [1, p. 230].

Previous work on path reduction techniques aim at minimizing either the errors in the impulse response power, the error in the delay spread, or the magnitude error in the frequency domain, e.g. [2]. However, they do not guarantee that these errors are reduced to zero. Also, several methods for converting the impulse response sampling time were studied in [3]. These methods preserve either the mean delay or the RMS delay spread.

In this paper, we present a novel framework for resampling *and* path reduction of impulse responses specified by a tapped-delay-line model. Our framework guarantees that the resulting impulse response preserves the properties *total power*, *mean delay*, and *RMS delay spread* at the same time for each impulse response. Furthermore, we select the phase angles of the resulting impulse response taps such that the frequency domain error of each channel realization is minimized as well. Our framework can also be applied to measured channel impulse responses to extract a given number of taps that have the three above guaranteed properties.

The paper is organized as follows. In Section 2 we present the framework of our path reduction algorithm. The detailed, mathematical descriptions of the individual processing parts are explained in Section 3. Implementation results are presented in Section 4 and our conclusions are drawn in Section 5.

2. THE FRAMEWORK

Our framework shown in Figure 1 takes an impulse response $a_n \in \mathbb{C}$ ($n = 0, \ldots, N_a - 1$) with sampling time T_i as input. The output is an impulse response (sampling time T_o) specified by the tap magnitudes d_m and the phase angles ϕ_m. The input impulse response has the following three channel properties:

1. Total power:
$$P = \sum_{n=0}^{N_a-1} |a_n|^2 \tag{1}$$
2. Mean delay:
$$\tau_\text{mean} = \frac{1}{P} \sum_{n=0}^{N_a-1} |a_n|^2 \, nT_\mathrm{i} \tag{2}$$
3. RMS delay spread:
$$\tau_\text{RMS} = \sqrt{\frac{1}{P} \sum_{n=0}^{N_a-1} |a_n|^2 \, (nT_\mathrm{i})^2 - \tau_\text{mean}^2} \tag{3}$$

After resampling and path reduction, we want these three properties to be preserved exactly. Our framework consists of four major parts,

This work has been funded by the Christian Doppler Laboratory for Design Methodology of Signal Processing Algorithms and the Austrian Research Centers GmbH.

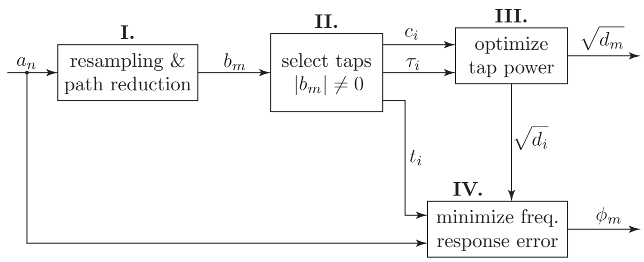

Fig. 1. Overview of the impulse response complexity reduction.

labeled I. to IV. in Figure 1. In Part I., the input impulse response is resampled from sampling time T_i to T_o using sinc interpolation. Additionally, in Part I., the number of non-zero taps is reduced to $N^{(\text{tap})}$ using an ad hoc path combining method [4]. This method generates an impulse response $b_m \in \mathbb{C}$ that roughly approximates the above three properties of a_n. In Part II., the energies of the taps b_m that are unequal to zero are stored in the variables $c_i \in \mathbb{R}$ ($i = 1, \ldots, N^{(\text{tap})}$) and the corresponding delay indices and delays in the variables t_i and τ_i, respectively. Part III. adjusts the energy values c_i to the energy values $d_i \in \mathbb{R}$ in order to guarantee that the total power, the mean delay, and the RMS delay spread are exactly the same as for the input impulse response. In Part IV., we select the phase angles ϕ_m to the corresponding power levels d_m, to achieve a minimum magnitude error in the frequency domain. All four parts of the framework are explained in detail in the next section.

3. DETAILS OF THE FRAMEWORK

Part I.: Resampling and Path Reduction

Given the impulse response a_n sampled with sampling time T_i, the following algorithm resamples a_n to the channel simulator's sampling time T_o and reduces the number of paths to $N^{(\text{tap})}$.

1. Resample the impulse response a_n to the channel simulator sampling time T_o using a sinc[1] interpolation filter of length $2M + 1$ (In our simulations we chose $M = 50$).

$$b_m = \frac{T_\text{o}}{T_\text{i}} \sum_{n=-M}^{M} a_n \text{sinc}\left(\frac{\pi}{T_\text{i}}(mT_\text{o} - nT_\text{i})\right) \tag{4}$$

2. Sum all tap values at negative time indices and add them to the tap b_0.

$$b_0 = \sum_{m \leq 0} b_m \tag{5}$$

   This step ensures that the resulting impulse response remains causal.

3. Construct the set $\mathcal{M}$ of all indices m, where the resampled impulse response is not zero, i.e. $b_m \neq 0$.

$$\mathcal{M} = \{m | b_m \neq 0\} \tag{6}$$

   If the number of non-zero taps is equal to the desired, maximum number of taps, i.e. $|\mathcal{M}| = N^{(\text{tap})}$, terminate the iteration.

4. Find the index $m^{(\text{min})}$ of the tap with the smallest energy unequal to zero.

$$m^{(\text{min})} = \arg\min_{m \in \mathcal{M}} |b_m|^2 \tag{7}$$

5. Find the nearest neighbor $m^{(\text{nb})}$ of $m^{(\text{min})}$ in $\mathcal{M}$. If two nearest neighbors exist, choose the one with smaller tap energy.

$$m^{(\text{nb})} = \arg\min_{m \in \mathcal{M}} |m - m^{(\text{min})}| \tag{8}$$

6. Add and update the tap values accordingly.

$$b_{m^{(\text{nb})}} \leftarrow b_{m^{(\text{nb})}} + b_{m^{(\text{min})}} \tag{9}$$

$$b_{m^{(\text{min})}} = 0 \tag{10}$$

7. Go to Step 3.

Note that after the sinc interpolation the number of non-zero taps is greatly increased. The combining in the subsequent path reduction algorithm tries to "concentrate" the energy of the taps near to their initial position (see also Figure 2). It should also be noted that the tap positions of the resulting impulse response depend on the actual input channel realization. This fact leads to a good approximation of the power delay profile and our three impulse response properties [4]. By the post processing performed in Parts II. to IV. of our framework, we adjust the tap energies and phases to improve the accuracy of the impulse response approximation.

Part II.: Determine the $N^{(\text{tap})}$ surviving paths and their delays

The set $\mathcal{M}$ now contains the $N^{(\text{tap})}$ indices t_i for which the energy of the impulse response b_m does not vanish. We define these indices as the vector $\mathbf{t} = [t_1, \ldots, t_{N^{(\text{tap})}}]^T$. We also define the vector $\mathbf{c} = [c_1, \ldots, c_{N^{(\text{tap})}}]^T$ that stores all corresponding energies $|b_m|^2$ that are larger than zero

$$c_i = |b_{t_i}|^2 \quad ; i = 1, \ldots, N^{(\text{tap})}, \tag{11}$$

and the corresponding tap delay vector $\boldsymbol{\tau} = [\tau_1, \ldots, \tau_{N^{(\text{tap})}}]^T$ with the elements $\tau_i = t_i T_\text{o}$ ($i = 1, \ldots, N^{(\text{tap})}$).

Part III.: Constrained Optimization of Tap Energies

Now we have found $N^{(\text{tap})}$ delays τ_i and the corresponding tap energies c_i. In the following we formulate a constraint optimization problem of the tap energies that guarantees an exact match of the total impulse response power, the mean delay, and the RMS delay spread. The optimization problem and the constraints (h_1, h_2, h_3, and $\mathbf{g}$) for the optimized tap energies $\mathbf{d} = [d_1, \ldots, d_{N^{(\text{tap})}}]$ are

$$\mathbf{d} = \arg\min_{\mathbf{d}} f(\mathbf{d}) = \arg\min_{\mathbf{d}} \sum_{i=1}^{N^{(\text{tap})}} (d_i - c_i)^2, \tag{12}$$

$$h_1(\mathbf{d}) = \sum_{i=1}^{N^{(\text{tap})}} d_i - P = 0, \tag{13}$$

$$h_2(\mathbf{d}) = \frac{\sum_{i=1}^{N^{(\text{tap})}} d_i \tau_i}{P} - \tau_\text{mean} = 0, \tag{14}$$

$$h_3(\mathbf{d}) = \frac{\sum_{i=1}^{N^{(\text{tap})}} d_i \tau_i^2}{P} - \tau_\text{mean}^2 - \tau_\text{RMS}^2 = 0, \tag{15}$$

$$\mathbf{g}(\mathbf{d}) = \mathbf{d} \geq \mathbf{0}. \tag{16}$$

In the last constraint, we mean that every component of $\mathbf{d}$ is larger or equal than zero. In other words, the power on every delay τ_i is not allowed to become negative. The Kuhn-Tucker conditions [5, p. 777] state that for a local minimum $\mathbf{d}^*$ there exists a Lagrange multiplier vector $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3]^T$ and a Lagrange multiplier vector $\boldsymbol{\mu} \leq 0$ such that

$$\mathbf{g}(\mathbf{d}^*)^T \boldsymbol{\mu} = 0$$

$$\nabla f(\mathbf{d}^*) + \nabla \mathbf{h}(\mathbf{d}^*)^T \boldsymbol{\lambda} + \nabla \mathbf{g}(\mathbf{d}^*)^T \boldsymbol{\mu} = 0. \tag{17}$$

[1] Here, we define the sinc function as $\text{sinc}\,(x) = \frac{\sin(x)}{x}$

We can therefore calculate our minimum $\mathbf{d}^*$ from Equation (17) by forming the Lagrangian

$$\begin{aligned} L &= \sum_{i=1}^{N^{(\text{tap})}} (d_i - c_i)^2 + \\ &+ \lambda_1 \left(\sum_{i=1}^{N^{(\text{tap})}} d_i - P \right) + \\ &+ \lambda_2 \left(\frac{\sum_{i=1}^{N^{(\text{tap})}} d_i \tau_i}{P} - \tau_{\text{mean}} \right) + \\ &+ \lambda_3 \left(\frac{\sum_{i=1}^{N^{(\text{tap})}} d_i \tau_i^2}{P} - \tau_{\text{mean}}^2 - \tau_{\text{RMS}}^2 \right) + \\ &+ \sum_{i=1}^{N^{(\text{tap})}} \mu_i d_i \end{aligned}$$

and its derivation with respect to d_i

$$\frac{\partial L}{\partial d_i} = 2\,(d_i - c_i) + \lambda_1 + \lambda_2 \frac{\tau_i}{P} + \lambda_3 \frac{\tau_i^2}{P} + \mu_i = 0 \qquad (18)$$

$$\rightarrow \quad d_i = c_i - \frac{\lambda_1}{2} - \frac{\lambda_2}{2}\frac{\tau_i}{P} - \frac{\lambda_3}{2}\frac{\tau_i^2}{P} - \mu_i. \qquad (19)$$

We now have to determine if the constraint for d_i is inactive ($d_i \geq 0$ and $\mu_i = 0$) or active ($d_i = 0$). If we define the set of inactive indices i as $\mathcal{I}$ and the number of inactive indices as $|\mathcal{I}|$ we can calculate

$$d_i = \begin{cases} c_i - \frac{\lambda_1}{2} - \frac{\lambda_2}{2}\frac{\tau_i}{P} - \frac{\lambda_3}{2}\frac{\tau_i^2}{P} & ; i \in \mathcal{I} \\ 0 & ; i \notin \mathcal{I} \end{cases}. \qquad (20)$$

If we use Equation (20) in our conditions for the resulting impulse response (Equations (13), (14), and (15)) we obtain for the Lagrange multipliers λ_i the following set of equations

$$\begin{aligned} &P = \sum_{i\in\mathcal{I}} x_i - \frac{|\mathcal{I}|}{2}\lambda_1 - \frac{\sum_{i\in\mathcal{I}} \tau_i}{2P}\lambda_2 - \frac{\sum_{i\in\mathcal{I}} \tau_i^2}{2P}\lambda_3, \\ &P\tau_{\text{mean}} = \sum_{i\in\mathcal{I}} x_i\tau_i - \frac{\sum_{i\in\mathcal{I}} \tau_i}{2}\lambda_1 - \frac{\sum_{i\in\mathcal{I}} \tau_i^2}{2P}\lambda_2 - \frac{\sum_{i\in\mathcal{I}} \tau_i^3}{2P}\lambda_3, \\ &P\left(\tau_{\text{RMS}}^2 + \tau_{\text{mean}}^2\right) = \\ &= \sum_{i\in\mathcal{I}} c_i\tau_i^2 - \frac{\sum_{i\in\mathcal{I}} \tau_i^2}{2}\lambda_1 - \frac{\sum_{i\in\mathcal{I}} \tau_i^3}{2P}\lambda_2 - \frac{\sum_{i\in\mathcal{I}} \tau_i^4}{2P}\lambda_3. \quad (21) \end{aligned}$$

Using a vector-matrix notation this system of linear equations can be rewritten as

$$\mathbf{K}\boldsymbol{\lambda} = \mathbf{y} \qquad (22)$$

with

$$\mathbf{K} = \sum_{i\in\mathcal{I}} \begin{bmatrix} 1 & \tau_i & \tau_i^2 \\ \tau_i & \tau_i^2 & \tau_i^3 \\ \tau_i^2 & \tau_i^3 & \tau_i^4 \end{bmatrix} = \sum_{i\in\mathcal{I}} \mathbf{K}_i, \qquad (23)$$

$$\boldsymbol{\lambda} = [P\lambda_1, \lambda_2, \lambda_3]^T, \qquad (24)$$

and

$$\mathbf{y} = 2P \begin{bmatrix} \sum_{i\in\mathcal{I}} c_i - P \\ \sum_{i\in\mathcal{I}} c_i\tau_i - P\tau_{\text{mean}} \\ \sum_{i\in\mathcal{I}} c_i\tau_i^2 - P\left(\tau_{\text{RMS}}^2 + \tau_{\text{mean}}^2\right) \end{bmatrix}. \qquad (25)$$

Equation (23) can only be solved exactly if the matrix $\mathbf{K}$ is regular. Since $\mathbf{K}$ is composed of a sum of rank one matrices $\mathbf{K}_i$, at least three matrices $\mathbf{K}_i$ have to be accumulated to obtain a full rank matrix $\mathbf{K}$. We can therefore only calculate $\boldsymbol{\lambda}$ for sets $\mathcal{I}$ where at least three constraints for d_i are inactive ($d_i > 0$). This is also intuitively clear because if we want to fulfill our three conditions (13)-(15) we have to adjust at least three tap energies.

The Lagrange vector $\boldsymbol{\lambda}$ can now be calculated for different choices of $\mathcal{I}$ and, with the knowledge of $\boldsymbol{\lambda}$, we can calculate the d_i from Equation (19). A solution d_i for a specific set $\mathcal{I}$ is valid if all $d_i \geq 0$ ($i = 1, \ldots, N^{(\text{tap})}$). Among the many valid solutions for d_i, we select the solution that minimizes the magnitude error of the frequency response, as explained in Part IV. of our framework. Note that it may happen—especially for small $N^{(\text{tap})}$—that the minimization yields no valid solution at all. In such a case, either the number of emulated taps has to be increased, or one of the three constraints has to be abandoned.

The $N^{(\text{tap})}$ energies d_i can be mapped to d_m, specified at the output sampling time T_o, by using

$$d_m = \begin{cases} d_i & ; m = t_i \\ 0 & ; \text{otherwise} \end{cases}. \qquad (26)$$

Part IV.: Select phases to minimize frequency response error

Once the tap delays τ_i have been found and the tap energies d_i have been adjusted, we can optimize the phase angles ϕ_i at these tap delays for a minimum mean square error in the frequency response. Let us assume that the relation between input and output sampling rate can be expressed by the rational number $\frac{T_\text{o}}{T_\text{i}} = \frac{N_\text{o}}{N_\text{i}}$. We can thus oversample the input impulse response by a factor of N_i and the output impulse response by a factor of N_o to obtain the common sampling time $T_\text{s} = \frac{T_\text{i}}{N_\text{i}} = \frac{T_\text{o}}{N_\text{o}}$. The oversampling is simply performed by inserting zeros between the known tap values. If the maximum number of samples required for representing both, the input and the output impulse response is $N_\text{max} = N_\text{i}N_\text{o}K$, we can calculate the discrete Fourier transform of the two impulse responses as $\mathbf{a}^{(\text{f})} = \mathbf{D}_a\mathbf{a}$ and $\mathbf{h}^{(\text{f})} = \mathbf{D}_h\mathbf{h}$ with the matrices $\mathbf{D}_a$ and $\mathbf{D}_h$ denoting the discrete Fourier transform matrices whose elements are defined as

$$(\mathbf{D}_a)_{k,n} = e^{-j\frac{2\pi}{N_\text{o}K}kn}\; ; k = 1, \ldots, N_\text{o}K;\; n = 1, \ldots, N_\text{i}K \quad (27)$$

$$(\mathbf{D}_h)_{k,i} = e^{-j\frac{2\pi}{N_\text{i}K}kt_i}\; ; k = 1, \ldots, N_\text{i}K;\; i = 1, \ldots, N^{(\text{tap})}. \quad (28)$$

and the vector $\mathbf{h}$ as $\mathbf{h} = \left[\sqrt{d_1}e^{j\phi_1}, \ldots, \sqrt{d_{N^{(\text{tap})}}}e^{j\phi_{N^{(\text{tap})}}}\right]^T$. Note that the $t_i = \frac{\tau_i}{T_\text{o}}$ (see description in Part II.) are the time indices that correspond to the non-vanishing energy taps c_i.

The unknown phases ϕ_i can now be determined by minimizing the mean square error in the frequency response. We obtain with $\mathbf{D}_a^H\mathbf{D}_a = N_iK\mathbf{I}$ and $\mathbf{D}_h^H\mathbf{D}_h = N_iK\mathbf{I}$

$$\begin{aligned} \left\|\mathbf{a}^{(\text{f})} - \mathbf{h}^{(\text{f})}\right\|_2^2 &= \|\mathbf{D}_a\mathbf{a} - \mathbf{D}_h\mathbf{h}\|_2^2 = \\ &= \mathbf{a}^H\mathbf{a} - 2\text{Re}\left\{\mathbf{h}^H\mathbf{D}_h^H\mathbf{D}_a\mathbf{a}\right\} + \mathbf{h}^H\mathbf{h}. \quad (29) \end{aligned}$$

A close look at Equation (29) reveals the following facts

- The expression $\mathbf{a}^H\mathbf{a}$ is the energy of the input impulse response and has to be considered as constant.
- The expression $\mathbf{h}^H\mathbf{h}$ is the energy of the output impulse response and was adjusted in Part III. to be the same as the input impulse response energy. Therefore, this term is also constant.

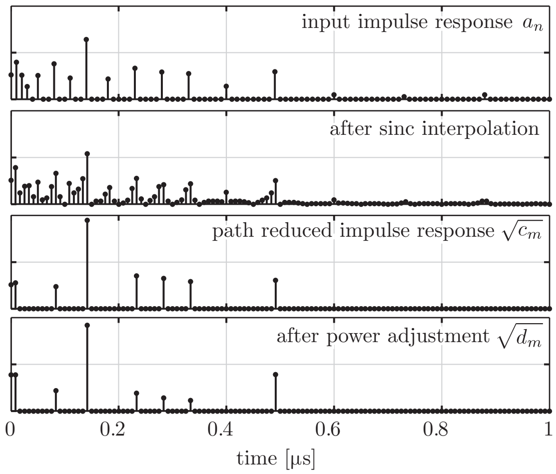

Fig. 2. Processing of one impulse response.

- The minimum of Equation (29) is therefore obtained when the real part of the expression $\mathbf{h}^H\mathbf{D}_h^H\mathbf{D}_a\mathbf{a}$ is maximized. This is achieved by choosing the phase angles ϕ_i of the vector $\mathbf{h}$ equal to the phases of the vector $\mathbf{D}_h^H\mathbf{D}_a\mathbf{a}$:

$$\phi_m = \begin{cases} \arg\left(\mathbf{D}_h^H\mathbf{D}_a\mathbf{a}\right)_i & ; m = t_i \\ 0 & ; \text{otherwise} \end{cases} . \tag{30}$$

The result of Equation (30) is somewhat intuitive since it means that the phase angles at the new sampling rate (and thus at the new positions) can be calculated as follows. Firstly the original impulse response is transformed to the frequency domain. Secondly, the frequency response is transformed back to the time domain at exactly the positions found by the path reduction algorithm. The phases obtained by this method can now be used to form the final impulse response.

Equation (29) is also a condition for selecting a specific solution d_i among the many valid solutions found for different choices of $\mathcal{I}$ in Part III.: The frequency response error is minimized, if the solution d_i that maximizes $\mathbf{h}^H\mathbf{D}_h^H\mathbf{D}_a\mathbf{a}$, is chosen.

4. IMPLEMENTATION RESULTS

The resampling, path reduction, and power optimization of one exemplary impulse response is illustrated in Figure 2. The impulse response a_n (first row of Figure 2) is generated according to the I-Metra channel model scenario "F" [6] at 100 MHz sampling rate. The impulse response after resampling to a 120 MHz channel emulator sampling frequency and after sinc interpolation is shown in the second row of Figure 2. The impulse response after path reduction to $N^{(\text{tap})} = 8$, is shown in the third row. We can see that the distribution of the tap energies approximately remains the same but is accumulated in only $N^{(\text{tap})}$ taps. Finally, the last row illustrates the power adjustment (Part III. of our framework) of the tap energies. The properties total power, mean delay, and RMS delay spread of the initial impulse response are perfectly preserved.

The magnitudes of the frequency responses corresponding to the input and output channel impulse responses of Figure 2 are shown in Figure 3. Although the magnitude error in the frequency domain is minimized by Part IV. of our framework, the accuracy of the approximation is not very high. This is due to the fact that only a limited number $N^{(\text{tap})}$ of phase angles (and not the corresponding tap energies) are adjusted for minimizing the magnitude error in the frequency domain. If a small error in the frequency domain is desired, this can be achieved by:

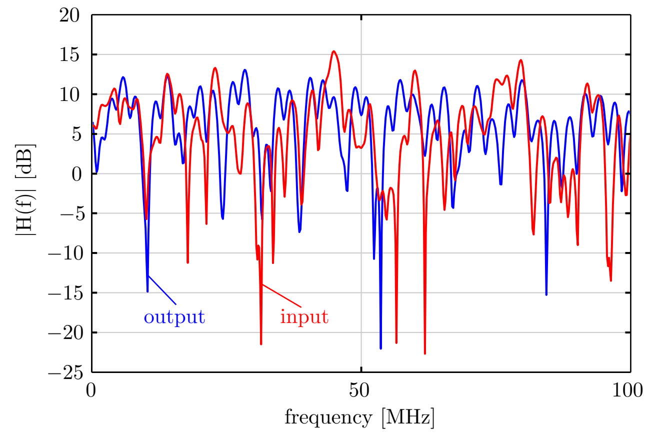

Fig. 3. Fourier transformations of the original and the resampled impulse responses.

1. Increasing the number of taps in the final impulse response. Unfortunately, this also increases the complexity of the channel emulation.
2. By using the tap energies *and* the phase angles in the minimization, the error could be reduced but the preservation of total power, mean delay, and RMS delay spread could not be guaranteed anymore.

Thus, both time domain errors and frequency response errors cannot be made arbitrarily small at the same time. In future work the tradeoff between frequency and time domain error is worthwhile to be investigated.

5. CONCLUSIONS

A novel framework for resampling and path reduction of given impulse responses was introduced. In contrast to previous work in this field, our framework guarantees that simultaneously, the impulse response power, the mean delay, and the RMS delay spread are preserved exactly. The phase angles of the output impulse response are selected such that the magnitude error in the frequency domain is minimized. Since the minimization of the frequency domain error is accomplished after the power adjustment and only by adjusting the phase angles of the taps, the approximation can only be rough. Future work should therefore aim at minimizing the frequency domain error further while still preserving the above properties.

References

[1] A. F. Molisch, *Wireless Communications*, 1st ed. John Wiley & Sons, 2005.

[2] J. Kolu, T. Jamsa, and A. Hulkkonen, "Real time simulation of measured radio channels," in *Proc. IEEE Vehicular Technology Conference (VTC 2003-Fall)*, vol. 1, Oct. 2003, pp. 183–187.

[3] J. Silva, N. Souto, A. Rodrigues, F. Cercas, and A. Correia, "Conversion of reference tapped delay line channel models to discrete time channel models [UMTS]," in *Proc. IEEE Vehicular Technology Conference (VTC 2003-Fall)*, vol. 1, Oct. 2003, pp. 128–132.

[4] C. Mehlführer, F. Kaltenberger, M. Rupp, and G. Humer, "Low-complexity MIMO channel simulation by reducing the number of paths," in *Proc. ITG/IEEE Workshop on Smart Antennas (WSA 2007)*, Vienna, Austria, Feb. 2007.

[5] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal processing*, 1st ed. Prentice Hall, 2000.

[6] J. P. Kermoal, L. Schumacher, K. I. Pedersen, P. E. Mogensen, and F. Frederiksen, "A stochastic MIMO radio channel model with experimental vaildation," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 6, pp. 1211–1226, Aug. 2002.