F. Hlawatsch and G. Matz,
Linear time-frequency filters,
in Time-Frequency Signal Analysis and Processing: A Comprehensive Reference,
ed. B. Boashash, Oxford, UK: Elsevier, 2003,ch. 11.1, pp. 466–475.

11.1 LINEAR TIME-FREQUENCY FILTERS[0]

11.1.1 Time-Frequency Design of Linear, Time-Varying Filters

Linear, time-varying (LTV) filters are useful in many applications, especially for weighting, suppressing, or separating nonstationary signal components. The input-output relation of an LTV filter $\mathbf{H}$ with kernel (impulse response) $h(t,t')$ reads

$$y(t) \;=\; (\mathbf{H}x)(t) \;=\; \int_{-\infty}^{\infty} h(t,t')\,x(t')\,dt' \;. \tag{11.1.1}$$

The nonstationary nature of input signal $x(t)$, output signal $y(t)$, and LTV filter $\mathbf{H}$ suggests the use of time-frequency (TF) representations for analyzing, designing, and/or implementing LTV filters.

There are two fundamentally different approaches to a TF design of LTV filters, namely, the "explicit" and "implicit" design philosophies [9] [11]. Both are based on a prescribed TF weight function $M(t,f)$ that provides a TF specification of the desired filtering characteristic.

- **Explicit design:** The impulse response $h(t,t')$ of the LTV filter $\mathbf{H}$ is calculated (designed) such that a TF representation of $\mathbf{H}$ is equal to or best approximates the TF weight function $M(t,f)$. In this article, the TF representation of $\mathbf{H}$ will be chosen as the *generalized Weyl symbol* (see Article 4.7). An alternative explicit design of LTV filters using the *Wigner distribution of an LTV system* has been considered in [6]. The filtering itself is performed in the time domain according to (11.1.1).

- **Implicit design:** The LTV filter $\mathbf{H}$ is designed *implicitly* during the filtering, which is a three-step analysis-weighting-synthesis procedure. First (analysis step), a *linear* TF representation—such as the short-time Fourier transform—of the input signal $x(t)$ is calculated. Second (weighting step), this TF representation is multiplied by the TF weight function $M(t,f)$. Third (synthesis step), the output signal $y(t)$ is calculated in a linear manner from the TF function obtained in Step 2. Since all processing steps are linear, the overall procedure amounts to an LTV filter.

In this article, we will consider explicit TF filter designs based on the generalized Weyl symbol [5] [7] [9] [11] and implicit TF filter designs based on the short-time Fourier transform [1] [2] [8] [9] [11] [12] [14] and the Gabor transform [3] [4] [11]. In particular, we will show that the resulting filters tend to perform similarly if the TF weight function $M(t,f)$ is sufficiently smooth.

[0]Authors: **F. Hlawatsch and G. Matz**, Institute of Communications and Radio-Frequency Engineering, Vienna University of Technology, Gusshausstrasse 25/389, A-1040 Vienna, Austria (email: fhlawats@pop.tuwien.ac.at, g.matz@ieee.org, web: http://www.nt.tuwien.ac.at/dspgroup/time.html). Reviewers: M. Amin and D. Jones. This work was supported by FWF grant P11904-TEC.

11.1.2 Explicit Design—The Generalized Weyl Filter

The *generalized Weyl symbol* (GWS) of an LTV system $\mathbf{H}$ is defined as

$$L_{\mathbf{H}}^{(\alpha)}(t,f) \triangleq \int_{-\infty}^{\infty} h\Big(t+\Big(\frac{1}{2}-\alpha\Big)\tau, t-\Big(\frac{1}{2}+\alpha\Big)\tau\Big)\, e^{-j2\pi f\tau}\, d\tau\,, \tag{11.1.2}$$

where α is a real-valued parameter. The special cases $\alpha = 0$ and $\alpha = 1/2$ give the *Weyl symbol* and *Zadeh's time-varying transfer function*, respectively. For *underspread* LTV systems (i.e., LTV systems that produce only moderate TF displacements), the GWS can be interpreted as a TF transfer function describing the TF weighting produced by the system (see Article 4.7). Hence, a conceptually simple TF design of an LTV filter $\mathbf{H}$ from a prescribed TF weight function $M(t,f)$ is based on setting the filter's GWS equal to $M(t,f)$ [7] [9] [11],

$$L_{\mathbf{H}_{\mathrm{GWF}}}^{(\alpha)}(t,f) \equiv M(t,f)\,. \tag{11.1.3}$$

The impulse response of the LTV filter $\mathbf{H}_{\mathrm{GWF}}$ thus defined is obtained via the inverse of (11.1.2), i.e.,

$$h_{\mathrm{GWF}}(t,t') = \int_{-\infty}^{\infty} M\Big(\Big(\frac{1}{2}+\alpha\Big)t+\Big(\frac{1}{2}-\alpha\Big)t', f\Big)\, e^{j2\pi f(t-t')}\, df\,. \tag{11.1.4}$$

The filter $\mathbf{H}_{\mathrm{GWF}}$ in (11.1.4) is termed *generalized Weyl filter* [9]; it depends on the choice of the GWS parameter α used in (11.1.3). In particular, the choices $\alpha = 0$ and $\alpha = 1/2$ lead to the *Weyl filter* and *Zadeh filter*, respectively. For $\alpha = 0$, a real-valued weight function $M(t,f)$ will result in a self-adjoint [13] Weyl filter.

Dependence on α. The dependence of $\mathbf{H}_{\mathrm{GWF}}$ on α effectively disappears in the case of a *smooth* TF weight function $M(t,f)$ (yielding an *underspread* LTV system $\mathbf{H}_{\mathrm{GWF}}$ as discussed in Article 4.7; also note that smoothness of $M(t,f)$ is incompatible with a sharp TF cutoff). Let $\mathbf{H}_{\mathrm{GWF}}^{(1)}$ and $\mathbf{H}_{\mathrm{GWF}}^{(2)}$ be two generalized Weyl filters designed according to (11.1.4) with GWS parameter α_1 and α_2, respectively. Then, one can show that the difference $(\mathbf{H}_{\mathrm{GWF}}^{(1)}x)(t) - (\mathbf{H}_{\mathrm{GWF}}^{(2)}x)(t)$ of the output signals of these filters satisfies

$$\frac{\big\|\mathbf{H}_{\mathrm{GWF}}^{(1)}x - \mathbf{H}_{\mathrm{GWF}}^{(2)}x\big\|_2}{\|x\|_2} \le \epsilon_1 \triangleq |\alpha_1-\alpha_2|\left\|\frac{\partial^2 M}{\partial t\,\partial f}\right\|_2,$$

where $\|\cdot\|_2$ denotes the L_2 norm. Thus, it is seen that the generalized Weyl filter design is almost independent of α if ϵ_1 is small, i.e., if $M(t,f)$ is a smooth function.

TF projection filter. Formally, (11.1.3) can be viewed as the solution to the (unconstrained) minimization problem $\mathbf{H}_{\mathrm{GWF}} = \arg\min_{\mathbf{H}} \big\|M - L_{\mathbf{H}}^{(\alpha)}\big\|_2$. Solving this minimization problem under the side constraint that $\mathbf{H}$ be an orthogonal projection operator[1] yields the *TF projection filter* $\mathbf{H}_{\mathrm{P}}$ introduced in [5]. More specifically, let $u_k(t)$ and λ_k denote the eigenfunctions and eigenvalues, respectively, of

[1] An orthogonal projection operator is characterized by being self-adjoint ($\mathbf{H}^+ = \mathbf{H}$) and idempotent ($\mathbf{HH} = \mathbf{H}$) [13].

$(\mathbf{H}_{\mathrm{GWF}} + \mathbf{H}_{\mathrm{GWF}}^{+})/2$, where $\mathbf{H}_{\mathrm{GWF}}^{+}$ denotes the adjoint of $\mathbf{H}_{\mathrm{GWF}}$ [13]. Then, the impulse response of $\mathbf{H}_{\mathrm{P}}$ can be shown [5] to equal

$$h_{\mathrm{P}}(t,t') \;=\; \sum_{k\in\mathcal{I}} u_k(t)\, u_k^*(t')\,,$$

where $\mathcal{I}$ is the set of indices k for which $\lambda_k > 1/2$.

The TF projection filter is only able to pass or suppress signal components, with no other weights possible. It is advantageous in some situations since it is capable of realizing very sharp TF cutoff characteristics. However, compared to $\mathbf{H}_{\mathrm{GWF}}$, the calculation of $\mathbf{H}_{\mathrm{P}}$ requires the additional solution of an eigenproblem. An efficient online implementation of the TF projection filter is proposed in [10].

11.1.3 Implicit Design I—The STFT Filter

An *STFT filter* consists of the following three steps [1] [2] [9] [11] [12] [14]:

- *Analysis*: Calculation of the short-time Fourier transform (STFT) [12] [14] of the input signal $x(t)$,

$$F_x^{\gamma}(t,f) \;=\; \int_{-\infty}^{\infty} x(t')\,\gamma_{t,f}^*(t')\,dt'\,,$$

  where $\gamma_{t,f}(t') = \gamma(t'-t)\,e^{j2\pi f t'}$ with $\gamma(t)$ being an analysis window.

- *Weighting*: Multiplication of the STFT by the TF weight function $M(t,f)$, i.e., calculation of $M(t,f)\,F_x^{\gamma}(t,f)$.

- *Synthesis*: The output signal $y(t)$ is obtained via an inverse STFT [12] [14],

$$y(t) \;=\; \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \left[M(t',f')\,F_x^{\gamma}(t',f')\right] g_{t',f'}(t)\,dt'df'\,.$$

  Here, $g_{t,f}(t') = g(t'-t)\,e^{j2\pi f t'}$ where $g(t)$ is a synthesis window that is usually assumed to satisfy $\int_{-\infty}^{\infty} g(t)\,\gamma^*(t)\,dt = 1$ (this guarantees perfect reconstruction for $M(t,f)\equiv 1$).

These steps implement an LTV filter—hereafter denoted $\mathbf{H}_{\gamma,g}$—that depends on the TF weight function $M(t,f)$ and the windows $\gamma(t)$ and $g(t)$.

Multiwindow STFT filter. An extension of the STFT filter $\mathbf{H}_{\gamma,g}$ is the *multiwindow STFT filter* [8] [11]

$$\mathbf{H}_N \;\triangleq\; \sum_{i=1}^{N} \eta_i\,\mathbf{H}_{\gamma^{(i)},g^{(i)}}\,, \qquad \text{with} \quad \sum_{i=1}^{N}\eta_i = 1\,,\;\; \eta_i\in\mathbb{R}\,.$$

This is a linear combination of N STFT filters $\mathbf{H}_{\gamma^{(i)},g^{(i)}}$ with the *same* TF weight function $M(t,f)$ but different analysis windows $\gamma^{(i)}(t)$ and different synthesis windows $g^{(i)}(t)$. Note that the STFT filter $\mathbf{H}_{\gamma,g}$ is a special case with $N=1$. Using a larger number N of STFT filters yields increased flexibility of design at the expense

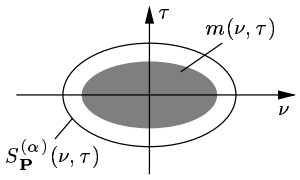

Figure 11.1: Illustration of the case where the effective support of $S_{\mathbf{P}}^{(\alpha)}(\nu,\tau)$ covers the effective support of $m(\nu,\tau)$.

of increased computational complexity. The impulse response of the multiwindow STFT filter $\mathbf{H}_N$ can be calculated as

$$h_N(t,\tilde{t}) \;=\; \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} M(t',f')\,p(t-t',\tilde{t}-t')\,e^{j2\pi f'(t-\tilde{t})}\,dt'\,df',$$

with $p(t,t') = \sum_{i=1}^{N} \eta_i\, g^{(i)}(t)\,\gamma^{(i)*}(t')$. Furthermore, the GWS of $\mathbf{H}_N$ is obtained as

$$L_{\mathbf{H}_N}^{(\alpha)}(t,f) \;=\; M(t,f) \mathbin{**} L_{\mathbf{P}}^{(\alpha)}(t,f)\,, \tag{11.1.5}$$

where $**$ denotes two-dimensional convolution and $\mathbf{P}$ is the LTV system with impulse response $p(t,t')$.

Comparison with generalized Weyl filter. Comparing (11.1.5) with (11.1.3), we see that the multiwindow STFT filter $\mathbf{H}_N$ using TF weight function $M(t,f)$ is equivalent to a generalized Weyl filter using the modified TF weight function $\widetilde{M}(t,f) = M(t,f) \mathbin{**} L_{\mathbf{P}}^{(\alpha)}(t,f)$. For nonnegative coefficients η_i, $\widetilde{M}(t,f)$ will be a smoothed version of $M(t,f)$. However, for $\eta_i = (-1)^i$ and $N \to \infty$, it is possible to have $L_{\mathbf{H}_N}^{(\alpha)}(t,f) \to M(t,f)$ and thus $\mathbf{H}_N \to \mathbf{H}_{\mathrm{GWF}}$, i.e., the multiwindow STFT filter approaches the generalized Weyl filter using the TF weight function $M(t,f)$. It can be shown that the difference $(\mathbf{H}_N x)(t) - (\mathbf{H}_{\mathrm{GWF}} x)(t)$ of the output signals of $\mathbf{H}_N$ and $\mathbf{H}_{\mathrm{GWF}}$ (both based on the same TF weight function $M(t,f)$) satisfies

$$\frac{\left\|\mathbf{H}_N x - \mathbf{H}_{\mathrm{GWF}} x\right\|_2}{\|x\|_2} \;\le\; \epsilon_2 \;\triangleq\; \left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \left|m(\nu,\tau)\right|^2 \left|1 - S_{\mathbf{P}}^{(\alpha)}(\nu,\tau)\right|^2 d\nu\,d\tau\right]^{1/2}. \tag{11.1.6}$$

Here, $m(\nu,\tau) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} M(t,f)\,e^{-j2\pi(\nu t - \tau f)}\,dt\,df$ and $S_{\mathbf{P}}^{(\alpha)}(\nu,\tau)$ is the generalized spreading function of $\mathbf{P}$ (see Article 4.7) The constant ϵ_2 is related to the operator $\mathbf{P}$ that characterizes the effect of the windows $\gamma^{(i)}(t)$, $g^{(i)}(t)$. in particular, ϵ_2 will be small if the effective support of $S_{\mathbf{P}}^{(\alpha)}(\nu,\tau)$ covers the effective support of $m(\nu,\tau)$, so that $\left|1 - S_{\mathbf{P}}^{(\alpha)}(\nu,\tau)\right|^2 \approx 0$ on the support of $m(\nu,\tau)$ (see Fig. 11.1). This is favored by a *smooth* TF weight function $M(t,f)$. Here, $m(\nu,\tau)$ is well concentrated about the origin and thus its effective support can easily be covered by $S_{\mathbf{P}}^{(\alpha)}(\nu,\tau)$, even using a small N. Hence, for a smooth TF weight function $M(t,f)$, the generalized Weyl filter can easily be approximated by the (multiwindow) STFT filter. This will be verified experimentally in Section 11.1.6.

11.1.4 Implicit Design II—The Gabor Filter

The Gabor transform is the STFT evaluated on a TF lattice (nT, kF) with $n, k \in \mathbb{Z}$ [4]. A *Gabor filter* (see [3] [11] and Article 11.2) consists of the following steps:

- *Analysis*: Calculation of the Gabor coefficients of the input signal $x(t)$ [4],

$$c_{n,k} = \int_{-\infty}^{\infty} x(t)\, \gamma_{n,k}^*(t)\, dt\,,$$

  where $\gamma_{n,k}(t) = \gamma(t - nT)\, e^{j2\pi kFt}$ with $\gamma(t)$ being a suitable analysis window.

- *Weighting*: Multiplication of the Gabor coefficients by the weights $M_{n,k} = M(nT, kF)$, i.e., calculation of $M_{n,k}\, c_{n,k}$.

- *Synthesis*: The output signal $y(t)$ is obtained via Gabor synthesis [4],

$$y(t) = \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} M_{n,k}\, c_{n,k}\, g_{n,k}(t)\,,$$

  where $g_{n,k}(t) = g(t - nT)\, e^{j2\pi kFt}$ with $g(t)$ being a suitable synthesis window.

This scheme implements an LTV filter that will be denoted $\widetilde{\mathbf{H}}_{\gamma,g}$. The windows $\gamma(t)$ and $g(t)$ are usually assumed to satisfy the perfect-reconstruction (biorthogonality) condition $\int_{-\infty}^{\infty} g(t)\, \gamma^*\left(t - \frac{n}{F}\right) e^{-j2\pi kt/T}\, dt = \delta_n\, \delta_k$, which presupposes critical sampling ($TF = 1$) or oversampling ($TF < 1$).

Multiwindow Gabor filter. An extension of the Gabor filter $\widetilde{\mathbf{H}}_{\gamma,g}$ is the *multiwindow Gabor filter* [11]

$$\widetilde{\mathbf{H}}_N \triangleq \sum_{i=1}^{N} \eta_i\, \widetilde{\mathbf{H}}_{\gamma^{(i)},g^{(i)}}\,, \qquad \text{with} \quad \sum_{i=1}^{N} \eta_i = 1\,, \quad \eta_i \in \mathbb{R}\,,$$

i.e., a linear combination of Gabor filters $\widetilde{\mathbf{H}}_{\gamma^{(i)},g^{(i)}}$ with the same TF weights $M_{n,k}$ but different analysis windows $\gamma^{(i)}(t)$ and different synthesis windows $g^{(i)}(t)$. The Gabor filter $\widetilde{\mathbf{H}}_{\gamma,g}$ is reobtained with $N = 1$. Using a larger number N of Gabor filters allows to reduce the TF sampling density TF (cf. [15]). The impulse response of the multiwindow Gabor filter $\widetilde{\mathbf{H}}_N$ is given by

$$\tilde{h}_N(t, t') = \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} M_{n,k}\, p(t - nT, t' - nT)\, e^{j2\pi kF(t-t')}\,,$$

with $p(t, t') = \sum_{i=1}^{N} \eta_i\, g^{(i)}(t) \gamma^{(i)*}(t')$, and the GWS of $\widetilde{\mathbf{H}}_N$ is

$$L_{\widetilde{\mathbf{H}}_N}^{(\alpha)}(t, f) = \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} M_{n,k}\, L_{\mathbf{P}}^{(\alpha)}(t - nT, f - kF)\,,$$

where $\mathbf{P}$ is the LTV system with impulse response $p(t, t')$.

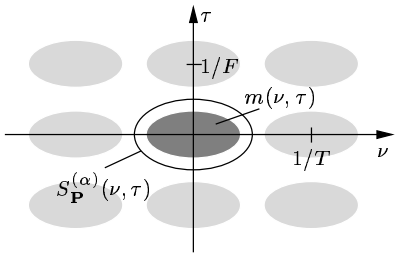

Figure 11.2: Illustration of windowing and aliasing effects involved in the Gabor filter design. The dark gray ellipse indicates the effective support of $m(\nu,\tau)$. The light gray ellipses indicate the effective support of $m\big(\nu-\frac{n}{F},\tau-\frac{k}{T}\big)$ for $(n,k)\neq(0,0)$.

Comparison with generalized Weyl filter. We next analyze how close the multiwindow Gabor filter $\widetilde{\mathbf{H}}_N$ (using TF weights $M_{n,k}=M(nT,kF)$) is to the generalized Weyl filter $\mathbf{H}_{\mathrm{GWF}}$ (using TF weight function $M(t,f)$). One can show that

$$\frac{\big\|\widetilde{\mathbf{H}}_N x-\mathbf{H}_{\mathrm{GWF}}x\big\|_2}{\|x\|_2}\;\leq\;\epsilon_2+\epsilon_3\,,$$

where ϵ_2 was given in (11.1.6) and ϵ_3 is defined as

$$\epsilon_3\;\triangleq\;\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\big|S_{\mathbf{P}}^{(\alpha)}(\nu,\tau)\big|^2\left|\sum_{\substack{n\neq 0\\ k\neq 0}} m\Big(\nu-\frac{n}{F},\tau-\frac{k}{T}\Big)\right|^2 d\nu\,d\tau\right]^{1/2}.$$

As in the case of the STFT filter, the term ϵ_2 is related to the operator $\mathbf{P}$ that describes the effect of the windows $\gamma^{(i)}(t)$, $g^{(i)}(t)$. If $M(t,f)$ is smooth so that $m(\nu,\tau)$ is well concentrated about the origin, a suitable choice of $\mathbf{P}$ allows to cover the effective support of $m(\nu,\tau)$ by the effective support of $S_{\mathbf{P}}^{(\alpha)}(\nu,\tau)$, which results in a small value of ϵ_2 (cf. our discussion in Section 11.1.3). The additional term ϵ_3 is mainly due to potential aliasing errors which are caused by the sampling $M_{n,k}=M(nT,kF)$ that distinguishes the Gabor filter from the STFT filter. For ϵ_3 to be small, it is necessary that the effective support of $S_{\mathbf{P}}^{(\alpha)}(\nu,\tau)$ does not overlap with the periodic repetitions $m\big(\nu-\frac{n}{F},\tau-\frac{k}{T}\big)$ of $m(\nu,\tau)$ (see Fig. 11.2). This can be achieved if (i) $m(\nu,\tau)$ is well concentrated about the origin and thus $m\big(\nu-\frac{n}{F},\tau-\frac{k}{T}\big)$ is well localized about $\big(\frac{n}{F},\frac{k}{T}\big)$ and (ii) T and F are small enough to ensure that the periodic repetitions $m\big(\nu-\frac{n}{F},\tau-\frac{k}{T}\big)$ are sufficiently separated. For $m(\nu,\tau)$ well concentrated, the latter condition can be met even for $TF>1$. Hence, we conclude that for a smooth $M(t,f)$, the generalized Weyl filter can be accurately approximated by the (multiwindow) Gabor filter.

11.1.5 The Discrete-Time Case

While our discussion of TF filters has so far been placed in a continuous-time framework, practical implementation of these filters calls for a discrete-time formulation.

The input-output relation of a *discrete-time* LTV system $\mathbf{H}$ reads

$$y[n] \;=\; (\mathbf{H}x)[n] \;=\; \sum_{n'=-\infty}^{\infty} h[n,n']\,x[n']\,,$$

with $h[n,n']$ being the impulse response of $\mathbf{H}$. The GWS with arbitrary α is not easily reformulated in a discrete-time setting. However, for $\alpha = 0$ (Weyl symbol) and $\alpha = 1/2$ (Zadeh's time-varying transfer function), which are the main cases of practical interest, discrete-time formulations are given by

$$L_{\mathbf{H}}^{(0)}(n,\theta) \;=\; 2\sum_{m=-\infty}^{\infty} h[n+m,n-m]\,e^{-j4\pi\theta m}\,,$$

$$L_{\mathbf{H}}^{(1/2)}(n,\theta) \;=\; \sum_{m=-\infty}^{\infty} h[n,n-m]\,e^{-j2\pi\theta m}\,.$$

Here, θ denotes normalized frequency. We note that in order for $L_{\mathbf{H}}^{(0)}(n,\theta)$ to be in one-to-one correspondence to $h[n,n']$, $\mathbf{H}$ has to be a *halfband system*, i.e., an LTV system that accepts input signal components only within a specified halfband (e.g., $\theta \in [-1/4,1/4)$) and maps them to a halfband output signal [11].

The TF system representations $L_{\mathbf{H}}^{(0)}(n,\theta)$ and $L_{\mathbf{H}}^{(1/2)}(n,\theta)$ can be used to design discrete-time LTV filters from a TF weight function $M(n,\theta)$ via an explicit filter design (cf. Section 11.1.2) [11]. The *discrete-time Zadeh filter* ($\alpha = 1/2$) is defined by setting $L_{\mathbf{H}}^{(1/2)}(n,\theta) = M(n,\theta)$; its impulse response is obtained as

$$h[n,n'] \;=\; \int_{-1/2}^{1/2} M(n,\theta)\,e^{j2\pi\theta(n-n')}\,d\theta\,.$$

In a similar manner, the *discrete-time Weyl filter* ($\alpha = 0$) is obtained as

$$h[n+m,n-m] \;=\; \int_{-1/4}^{1/4} M(n,\theta)\,e^{j2\pi\theta m}\,d\theta\,. \tag{11.1.7}$$

Since $L_{\mathbf{H}}^{(0)}(n,\theta)$ is meaningful only for halfband systems, $M(n,\theta)$ here is specified on the halfband $[-1/4,1/4)$. According to (11.1.7), the impulse response $h[n_1,n_2]$ of the discrete-time Weyl filter is obtained only for n_1+n_2 even (since $n_1+n_2 = n+m+n-m = 2n$). If we assume $\mathbf{H}$ to be a halfband system, $h[n_1,n_2]$ is completely specified by these samples. The missing samples (for n_1+n_2 odd) could be obtained by interpolation; however, this is not necessary since the entire filtering can be performed using only the even-indexed samples [11]. We note that in some cases (especially for "chirpy" TF weight functions), the Weyl filter design results in better filtering performance than the Zadeh filter design [11].

Discrete-time versions of the implicit filter design methods from Sections 11.1.3 and 11.1.4 can be obtained in a straightforward manner; see [1] [11] [12] [14] for discrete-time STFT filters and [3] [11] for discrete-time Gabor filters.

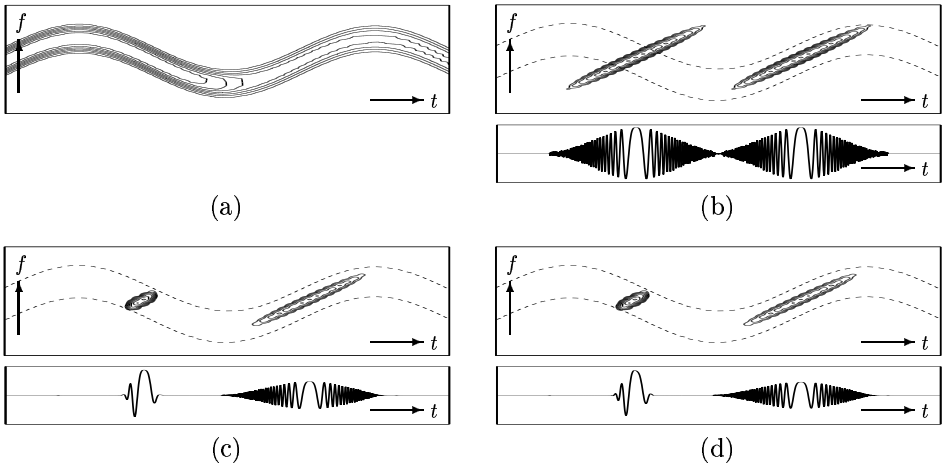

Figure 11.3: Comparison of explicit and implicit TF filter designs: (a) Specified TF weight function $M(t,f)$, (b)–(d) spectrogram (top) and real part (bottom) of (b) input signal $x(t)$, (c) output signal $y(t)$ obtained with Zadeh filter, and (d) output signal $y(t)$ obtained with STFT filter. The dashed lines in the spectrograms indicate the TF pass region. The time duration is 2048 samples; the (normalized) frequency interval shown is $[-1/2, 1/2)$.

11.1.6 Simulation Results

Our first simulation example, shown in Fig. 11.3, compares the performance of the Zadeh filter $\mathbf{H}_{\mathrm{GWF}}$ (generalized Weyl filter with $\alpha = 1/2$) and the STFT filter $\mathbf{H}_{\gamma,g}$. The TF weight function $M(t,f)$ (see Fig. 11.3(a)) models a bandpass filter with sinusoidally time-varying center frequency and time-varying gain. The gain is 1 in the first (earlier) half and 1/2 in the second (later) half, with a roll-off in between. The two filters were applied to an input signal $x(t)$ consisting of two chirps (see Fig. 11.3(b)). The resulting output signals (shown in Figs. 11.3(c),(d)) are seen to conform to the specified TF weighting. Furthermore, they are effectively identical (we obtained $\|\mathbf{H}_{\gamma,g}x - \mathbf{H}_{\mathrm{GWF}}x\|_2 / \|\mathbf{H}_{\mathrm{GWF}}x\|_2 = 0.047$), which is due to the smoothness of $M(t,f)$ and confirms our approximation in (11.1.6).

The application of a multiwindow Gabor filter $\widetilde{\mathbf{H}}_N$ to speech enhancement (denoising) is considered in Fig. 11.4. The speech signal $s(t)$ and its noisy version $x(t) = s(t) + n(t)$ (where $n(t)$ is white noise with an SNR of 0 dB) are shown in Fig. 11.4(a),(b). The multiwindow Gabor filter has $N = 5$ branches and lattice parameters $T = 5.8\,\mathrm{ms}$, $F = 172.25\,\mathrm{Hz}$. The analysis/synthesis windows $\gamma^{(i)}(t) = g^{(i)}(t)$ and the branch weights η_i were chosen as discussed in [11, Section 4.6.5]. The weights $M_{n,k}$ were computed from the multiwindow Gabor coefficients $c_{n,k}^{(i)} = \int_{-\infty}^{\infty} x(t)\,\gamma_{n,k}^{(i)*}(t)\,dt$ in a signal-adaptive, online manner that does not require knowledge about the clean speech signal or its statistics [11, Section 4.7.3]. The filter output is shown in Fig. 11.4(c); the SNR improvement is 4.92 dB.

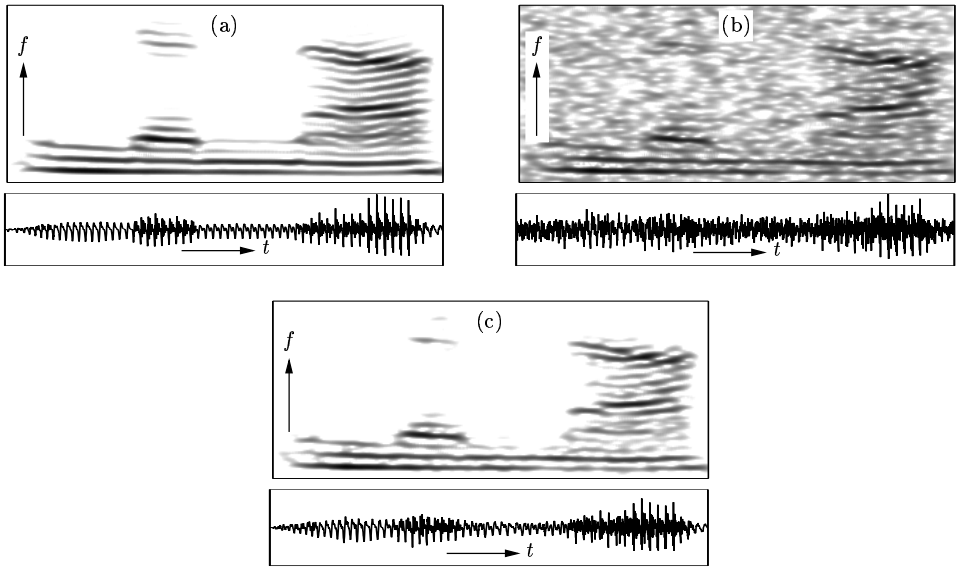

Figure 11.4: Speech enhancement using a multiwindow Gabor filter $\widetilde{\mathbf{H}}_N$. The figure shows the smoothed pseudo-Wigner distribution (top) and the time-domain signal (bottom) of (a) the clean speech, (b) the noisy speech (input of $\widetilde{\mathbf{H}}_N$), and (c) the enhanced speech (output of $\widetilde{\mathbf{H}}_N$). The time duration is 4096 samples; the (normalized) frequency interval shown is $[0, 1/2)$.

11.1.7 Summary and Conclusions

We have discussed "explicit" and "implicit" time-frequency (TF) designs of linear, time-varying filters. These design methods are useful for filtering nonstationary signals if the filter characteristic can be specified in the TF domain via a TF weight function. All filters discussed (except the TF projection filter) tend to perform similarly if the TF weight function is sufficiently smooth. In the opposite case, however, different designs may result in filters that perform very differently [11].

We finally note that the application of TF filtering to nonstationary signal estimation and detection is considered in Article 12.4. Other TF approaches to time-varying filtering are described in Articles 11.2–11.4.

References

[1] R. Bourdier, J. F. Allard, and K. Trumpf, "Effective frequency response and signal replica generation for filtering algorithms using multiplicative modifications of the STFT," *Signal Processing*, vol. 15, pp. 193–201, Sept. 1988.

[2] I. Daubechies, "Time-frequency localization operators: A geometric phase space approach," *IEEE Trans. Inf. Theory*, vol. 34, pp. 605–612, July 1988.

[3] S. Farkash and S. Raz, "Linear systems in Gabor time-frequency space," *IEEE Trans. Signal Processing*, vol. 42, no. 3, pp. 611–617, 1994.

[4] H. G. Feichtinger and T. Strohmer, eds., *Gabor Analysis and Algorithms: Theory and Applications*. Boston (MA): Birkhäuser, 1998.

[5] F. Hlawatsch, *Time-Frequency Analysis and Synthesis of Linear Signal Spaces: Time-Frequency Filters, Signal Detection and Estimation, and Range-Doppler Estimation*. Boston (MA): Kluwer, 1998.

[6] F. Hlawatsch and G. Matz, "Quadratic time-frequency analysis of linear time-varying systems," in *Wavelet Transforms and Time-Frequency Signal Analysis* (L. Debnath, ed.), ch. 9, pp. 235–287, Boston (MA): Birkhäuser, 2001.

[7] W. Kozek, "Time-frequency signal processing based on the Wigner-Weyl framework," *Signal Processing*, vol. 29, pp. 77–92, Oct. 1992.

[8] W. Kozek, H. G. Feichtinger, and J. Scharinger, "Matched multiwindow methods for the estimation and filtering of nonstationary processes," in *Proc. IEEE ISCAS-96*, (Atlanta, GA), pp. 509–512, May 1996.

[9] W. Kozek and F. Hlawatsch, "A comparative study of linear and nonlinear time-frequency filters," in *Proc. IEEE-SP Int. Sympos. Time-Frequency Time-Scale Analysis*, (Victoria, Canada), pp. 163–166, Oct. 1992.

[10] G. Matz and F. Hlawatsch, "Time-frequency projection filters: Online implementation, subspace tracking, and application to interference excision," in *Proc. IEEE ICASSP-02*, Orlando (FL), May 2002.

[11] G. Matz and F. Hlawatsch, "On-line algorithms for time-frequency signal processing," in *Applications in Time-Frequency Signal Processing* (A. Papandreou-Suppappola, ed.), Boca Raton (FL): CRC Press, 2002.

[12] S. H. Nawab and T. F. Quatieri, "Short-time Fourier transform," in *Advanced Topics in Signal Processing* (J. S. Lim and A. V. Oppenheim, eds.), ch. 6, pp. 289–337, Englewood Cliffs (NJ): Prentice Hall, 1988.

[13] A. W. Naylor and G. R. Sell, *Linear Operator Theory in Engineering and Science*. New York: Springer, 2nd ed., 1982.

[14] M. R. Portnoff, "Time-frequency representation of digital signals and systems based on short-time Fourier analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 28, pp. 55–69, Feb. 1980.

[15] Y. Y. Zeevi, M. Zibulski, and M. Porat, "Multi-window Gabor schemes in signal and image representations," in *Gabor Analysis and Algorithms: Theory and Applications* (H. G. Feichtinger and T. Strohmer, eds.), ch. 12, pp. 381–407, Boston (MA): Birkhäuser, 1998.