Postprint of: Niedźwiecki M., Sobociński A., Generalized adaptive notch filters with frequency debiasing for tracking of polynomial phase systems, Automatica, Vol. 43, Iss. 1 (2007), pp. 128-134, DOI: 10.1016/j.automatica.2006.08.004

Generalized Adaptive Notch Filters With Frequency Debiasing for Tracking of Polynomial Phase Systems *

Maciej Niedźwiecki [a] Adam Sobociński [a]

[a] *Faculty of Electronics, Telecommunications and Computer Science, Department of Automatic Control, Gdańsk University of Technology, ul. Narutowicza 11/12, Gdańsk , Poland, e-mail: maciekn@eti.pg.gda.pl, adsob@eti.pg.gda.pl*

Abstract

Generalized adaptive notch filters are used for identification/tracking of quasi-periodically varying dynamic systems and can be considered an extension, to the system case, of classical adaptive notch filters. For general patterns of frequency variation the generalized adaptive notch filtering algorithms yield biased frequency estimates. We show that when system frequencies change slowly in a smooth way, the estimation bias can be substantially reduced by means of post-filtering of the frequency estimates. The modified (debiased) algorithm has better tracking capabilities than the original algorithm.

Key words: system identification, time-varying processes, frequency estimation

1 Introduction

The term "generalized adaptive notch filter" (GANF) was coined in (Niedźwiecki & Kaczmarek, 2004) and denotes an adaptive filtering algorithm capable of identification/tracking of quasi-periodically varying systems. Complex-valued quasi-periodically varying systems are governed by

$$y(t) = \sum_{l=1}^{n} \theta_l(t)\varphi_l(t) + v(t) = \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\theta}(t) + v(t) \qquad (1)$$

where $t = 1, 2, \ldots$ denotes the normalized discrete time, $y(t)$ denotes the system output, $\boldsymbol{\varphi}(t) = [\varphi_1(t), \ldots, \varphi_n(t)]^{\mathrm{T}}$ is the regression vector, $v(t)$ is an additive noise and $\boldsymbol{\theta}(t) = [\theta_1(t), \ldots, \theta_n(t)]^{\mathrm{T}}$ denotes the vector of time varying coefficients, modeled as weighted sums of complex exponentials

$$\theta_l(t) = \sum_{i=1}^{k} a_{li}(t) e^{j\sum_{s=1}^{t}\omega_i(s)}, \quad l = 1, \ldots, n \qquad (2)$$

Since both the complex amplitudes $a_{li}(t)$ and the angular frequencies $\omega_i(t)$ in (2) are assumed to vary slowly with time, the system described by (1) - (2) changes in a periodic-like, but not exactly periodic manner.
Denote by $\boldsymbol{\alpha}_i(t) = [a_{1i}(t), \ldots, a_{ni}(t)]^{\mathrm{T}}$ the vector of system coefficients associated with a particular frequency ω_i and let $\boldsymbol{\beta}_i(t) = f_i(t)\boldsymbol{\alpha}_i(t)$, where $f_i(t) = e^{j\sum_{s=1}^{t}\omega_i(s)}$.

* This work was supported by MNiSW under Grant N514 011 31/3091.

Using the short-hand notation introduced above, system equation (1) can be rewritten in the form

$$y(t) = \sum_{i=1}^{k} \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\beta}_i(t) + v(t) \qquad (3)$$

When the sequence of regression vectors $\{\boldsymbol{\varphi}(t)\}$ is wide-sense stationary and persistently exciting, with known covariance matrix $\boldsymbol{\Phi} = \mathrm{E}[\boldsymbol{\varphi}^*(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)] > 0$, the normalized steady state single-frequency ($k = 1$) version of the GANF algorithm presented in (Niedźwiecki & Kaczmarek, 2006a) can be written down in the form

$$\begin{aligned}
\varepsilon(t) &= y(t) - e^{j\widehat{\omega}(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}(t-1) \\
\widehat{\boldsymbol{\beta}}(t) &= e^{j\widehat{\omega}(t)}\widehat{\boldsymbol{\beta}}(t-1) + \mu\boldsymbol{\Phi}^{-1}\boldsymbol{\varphi}^*(t)\varepsilon(t) \\
g(t) &= \mathrm{Im}\left[\frac{\varepsilon^*(t)e^{j\widehat{\omega}(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}(t-1)}{\widehat{\boldsymbol{\beta}}^{\mathrm{H}}(t-1)\boldsymbol{\Phi}\widehat{\boldsymbol{\beta}}(t-1)}\right] \\
\widehat{\omega}(t+1) &= \widehat{\omega}(t) - \gamma g(t) \\
\widehat{\boldsymbol{\theta}}(t) &= \widehat{\boldsymbol{\beta}}(t)
\end{aligned} \qquad (4)$$

Tracking properties of this algorithm are determined by two user-dependent tuning coefficients: the adaptation gain $0 < \mu \ll 1$, which controls the rate of amplitude adaptation, and another adaptation gain $0 < \gamma \ll \mu$, which decides upon the rate of frequency adaptation.
The multiple frequency GANF algorithm can be obtained in a pretty straightforward way by combining k single-frequency identification subalgorithms, given by (4), into an appropriately designed parallel structure - see (Niedźwiecki & Kaczmarek, 2004) and Section 2.3.

In a special case where $n=1$ and $\varphi(t)=1, \forall t$, the model (1) - (2) becomes a description of a noisy nonstationary multifrequency signal $y(t)=\sum_{i=1}^{k} a_i(t) e^{j \sum_{s=1}^{t} \omega_i(s)} + v(t)$. In this case generalized adaptive notch filters turn into "ordinary" adaptive notch filters (ANF) - the algorithms used for extraction or elimination of sinusoidal signals buried in noise - see (Tichavský & Händel, 1995), (Tichavský & Nehorai, 1997) and the references therein. Adaptive notch filters have many applications such as cancellation of sinusoidal interferences or adaptive line enhancement (Pei & Tseng, 1994). Generalized adaptive notch filters can be applied to equalization of rapidly fading telecommunication channels (Tsatsanis & Giannakis, 1996), (Bakkoury et al., 2000).

2 Modified GANF algorithm

2.1 Tracking properties of GANF

Tracking properties of the GANF algorithm (4) can be analyzed using the approximating linear filter (ALF) technique proposed in (Tichavský & Händel, 1995). Denote by $\Delta\widehat{\omega}(t)=\widehat{\omega}(t)-\omega(t)$ the frequency estimation error, and let $w(t)=\omega(t)-\omega(t-1)$ stand for the true one-step frequency change.

Consider a quasi-periodically varying system with a single frequency mode ($k=1$), governed by

$$\boldsymbol{\beta}(t)=e^{j\omega(t)}\boldsymbol{\beta}(t-1) \tag{5}$$

Let $b^2=\boldsymbol{\beta}^{\mathrm{H}}(t)\boldsymbol{\Phi}\boldsymbol{\beta}(t)=\boldsymbol{\beta}_o^{\mathrm{H}}\boldsymbol{\Phi}\boldsymbol{\beta}_o$ and

$$z(t)=\mathrm{Im}\left[\frac{\boldsymbol{\beta}^{\mathrm{H}}(t)\boldsymbol{\varphi}^*(t)v(t)}{b^2}\right]$$

It is easy to check that $\{z(t)\}$ is a real-valued white noise with variance $\sigma_z^2=\sigma_v^2/(2b^2)$

If the sequence of regression vectors $\{\boldsymbol{\varphi}(t)\}$, independent of $\{v(t)\}$ and $\{w(t)\}$, is wide-sense stationary and persistently exciting. then the frequency estimation errors yielded by the GANF algorithm (4), applied to the system governed by (5), can be approximately described by the following linear equation (Niedźwiecki & Kaczmarek, 2006a)

$$\Delta\widehat{\omega}(t)\cong E_1(q^{-1})z(t)+E_2(q^{-1})w(t) \tag{6}$$

where q^{-1} denotes the backward shift operator,

$$\begin{aligned} E_1(q^{-1}) &= \frac{(1-\delta)(1-q^{-1})q^{-1}}{1-(\lambda+\delta)q^{-1}+\lambda q^{-2}} \\ E_2(q^{-1}) &= -\frac{1-\lambda q^{-1}}{1-(\lambda+\delta)q^{-1}+\lambda q^{-2}} \end{aligned} \tag{7}$$

and $\lambda=1-\mu$, $\delta=1-\gamma$.

Suppose that the instantaneous frequency changes linearly with time, that is $w(t)=\omega(t)-\omega(t-1)=\delta_\omega, \forall t$. By taking expectations of both sides of (6) one arrives at $\mathrm{E}[\Delta\widehat{\omega}(t)]=-\delta_\omega\mu/\gamma$, which shows that the frequency estimates are in this case biased. This is a typical situation – parameter estimates yielded by causal adaptive filters usually lag behind the true signal/system parameters (Niedźwiecki, 2000).

We will show that when the system frequency changes slowly in a smooth way, the frequency bias introduced by the GANF algorithm can be significantly reduced by means of post-filtering of the frequency estimates.

2.2 Frequency debiasing

Derivation of the correction scheme will be based on assumption that the true frequency trajectory $\{\omega(t)\}$ can be locally (i.e. in sufficiently short time intervals) approximated by a polynomial model

$$\omega(t)\cong\sum_{i=0}^{m} c_i t^i \tag{8}$$

Note that this is a reasonable hypothesis in mobile radio channel applications, where $\omega(t)$ is a Doppler shift along a specific path of signal arrival and its variation is caused by the vehicle's speed changes.

Since under (8) the phase $\phi(t)=\sum_{s=1}^{t}\omega(s)$ is also a polynomial of t, in the signal processing case the model (8) is usually referred to as polynomial phase. Even though there is a number of algorithms capable of tracking polynomial phase signals – see e.g. (Tichavský & Händel, 1997a) and (Tichavský & Händel, 1997b) – none of them can be easily extended to the system case. To the best of our knowledge the approach described below is the first attempt to solve the parameter tracking problem for polynomial phase systems. As a byproduct of the system-oriented analysis, we will also obtain a novel tracking algorithm for polynomial phase signals.

Note that the ALF equation (6) can be rewritten in the form

$$\widehat{\omega}(t)\cong F_1(q^{-1})z(t)+F_2(q^{-1})\omega(t) \tag{9}$$

where $F_1(q^{-1})=E_1(q^{-1})$ and

$$F_2(q^{-1})=\frac{(1-\delta)q^{-1}}{1-(\lambda+\delta)q^{-1}+\lambda q^{-2}} \tag{10}$$

It is straightforward to check that the nominal (low-frequency) delay introduced by the filter $F_2(e^{j\xi})=A_2(\xi)e^{j\phi_2(\xi)}$, where ξ denotes the standard Fourier-domain frequency variable, is equal to

$$\tau=-\lim_{\xi\mapsto 0}\frac{d\phi_2(\xi)}{d\xi}=\frac{\mu}{\gamma} \tag{11}$$

Then, according to (8), (9) and (11), it holds that

$$\mathrm{E}[\widehat{\omega}(t)|\omega(s), s\le t]\cong F_2(q^{-1})\omega(t)\cong\omega(t-\tau)=\mathbf{f}^{\mathrm{T}}(t)\mathbf{c}$$

where $\mathbf{f}(t)=[1, t-\tau, \ldots, (t-\tau)^m]^{\mathrm{T}}$ and $\mathbf{c}=[c_0, c_1, \ldots, c_m]^{\mathrm{T}}$ is the vector of unknown coefficients.

The local estimate of $\mathbf{c}$ can be obtained by applying the method of exponentially weighted least squares

$$\widehat{\mathbf{c}}(t) = \arg\min_{\mathbf{c}} \sum_{i=0}^{t-1} \eta^i [\widehat{\omega}(t-i) - \mathbf{f}^{\mathrm{T}}(t-i)\mathbf{c}]^2$$

$$= \left[\sum_{i=0}^{t-1} \eta^i \mathbf{f}(t-i)\mathbf{f}^{\mathrm{T}}(t-i)\right]^{-1} \sum_{i=0}^{t-1} \eta^i \mathbf{f}(t-i)\widehat{\omega}(t-i) \quad (12)$$

where $0 < \eta < 1$ denotes a forgetting constant, introduced to decrease the influence of old data on the current parameter estimates. Exponential forgetting is a standard technique allowing one to track slow changes of the estimated coefficients – in our case the possible slow changes in $\mathbf{c}$. Replacing $c_0, \ldots, c_m$ in (8) with $\widehat{c}_0(t), \ldots, \widehat{c}_m(t)$, respectively, and compensating the estimation delay, one arrives at the following estimator

$$\bar{\omega}(t) = \mathbf{f}^{\mathrm{T}}(t+\tau)\widehat{\mathbf{c}}(t) \quad (13)$$

which belongs to the class of exponentially weighted basis function (EWBF) estimators, described in (Niedźwiecki, 2000).
The recursive algorithm for computation of $\widehat{\mathbf{c}}(t)$ has the form

$$\begin{aligned} \mathbf{f}(t) &= \mathbf{A}\mathbf{f}(t-1) \\ \widehat{\varepsilon}(t) &= \widehat{\omega}(t) - \mathbf{f}^{\mathrm{T}}(t)\widehat{\mathbf{c}}(t-1) \\ \mathbf{k}(t) &= \frac{\mathbf{G}(t-1)\mathbf{f}(t)}{\eta + \mathbf{f}^{\mathrm{T}}(t)\mathbf{G}(t-1)\mathbf{f}(t)} \\ \mathbf{G}(t) &= \frac{1}{\eta}\left[\mathbf{G}(t-1) - \mathbf{k}(t)\mathbf{f}^{\mathrm{T}}(t)\mathbf{G}(t-1)\right] \\ \widehat{\mathbf{c}}(t) &= \widehat{\mathbf{c}}(t-1) + \mathbf{k}(t)\widehat{\varepsilon}(t) \end{aligned} \quad (14)$$

where $\mathbf{G}(t) = \left[\sum_{i=0}^{t-1} \eta^i \mathbf{f}(t-i)\mathbf{f}^{\mathrm{T}}(t-i)\right]^{-1}$ and

$$\mathbf{A} = \begin{bmatrix} 1 & & & \\ \binom{1}{1} & \ddots & 0 & \\ \vdots & & & \\ \binom{m}{m} & \ldots & \binom{m}{1} & 1 \end{bmatrix}$$

The fact that the elements of $\mathbf{f}(t)$ are not bounded for $t \mapsto \infty$ may cause numerical problems. The numerically safe algorithm, which is free of the drawback mentioned above, can be obtained by rewriting (14) in a different system of coordinates: $\widetilde{\mathbf{c}}(t) = (\mathbf{A}^{\mathrm{T}})^{t-\tau}\widehat{\mathbf{c}}(t)$, $\widetilde{\mathbf{k}}(t) = (\mathbf{A}^{\mathrm{T}})^{t-\tau}\mathbf{k}(t)$, $\widetilde{\mathbf{G}}(t-1) = (\mathbf{A}^{\mathrm{T}})^{t-\tau}\mathbf{G}(t-1)\mathbf{A}^{t-\tau}$. Using these substitutions one can rewrite (13) and (14) in the following equivalent form

$$\begin{aligned} \widehat{\varepsilon}(t) &= \widehat{\omega}(t) - \mathbf{f}^{\mathrm{T}}(\tau+1)\widetilde{\mathbf{c}}(t-1) \\ \widetilde{\mathbf{k}}(t) &= \frac{\widetilde{\mathbf{G}}(t-1)\mathbf{f}(\tau)}{\eta + \mathbf{f}^{\mathrm{T}}(\tau)\widetilde{\mathbf{G}}(t-1)\mathbf{f}(\tau)} \\ \widetilde{\mathbf{G}}(t) &= \frac{\mathbf{A}^{\mathrm{T}}}{\eta}\left[\widetilde{\mathbf{G}}(t-1) - \widetilde{\mathbf{k}}(t)\mathbf{f}^{\mathrm{T}}(\tau)\widetilde{\mathbf{G}}(t-1)\right]\mathbf{A} \\ \widetilde{\mathbf{c}}(t) &= \mathbf{A}^{\mathrm{T}}\widetilde{\mathbf{c}}(t-1) + \widetilde{\mathbf{k}}(t)\widehat{\varepsilon}(t) \\ \bar{\omega}(t) &= \mathbf{f}^{\mathrm{T}}(2\tau)\widetilde{\mathbf{c}}(t) \end{aligned} \quad (15)$$

Note that $\mathbf{f}^{\mathrm{T}}(\tau) = [1, 0, \ldots, 0]$, $\mathbf{f}^{\mathrm{T}}(\tau+1) = [1, 1, \ldots, 1]$ and $\mathbf{f}^{\mathrm{T}}(2\tau) = [1, \tau, \ldots, \tau^m]$.
One can show that the relationship between $\bar{\omega}(t)$ and $\widehat{\omega}(t)$, given by (12) and (13), is asymptotically time-invariant, i.e. in steady state it holds that

$$\bar{\omega}(t) = H_{m,\tau,\eta}(q^{-1})\widehat{\omega}(t) \quad (16)$$

where $H_{m,\tau,\eta}(q^{-1})$ is a stationary rational filter. In particular (see Appendix)

$$H_{1,\tau,\eta}(q^{-1}) = \frac{(1-\eta)(a_0 + a_1 q^{-1})}{(1-\eta q^{-1})^2} \quad (17)$$

where $a_0 = 1 + \eta + (1-\eta)\tau$, $a_1 = -2\eta - (1-\eta)\tau$, and

$$H_{2,\tau,\eta}(q^{-1}) = \frac{(1-\eta)(a_0 + a_1 q^{-1} + a_2 q^{-2})}{2(1-\eta q^{-1})^3} \quad (18)$$

where $a_0 = 2(\eta^2+\eta+1) + 3(1+\eta)(1-\eta)\tau + (1-\eta)^2\tau^2$, $a_1 = -6\eta(1+\eta) - 4(1+2\eta)(1-\eta)\tau - 2(1-\eta)^2\tau^2$ and $a_2 = 6\eta^2 + (1+5\eta)(1-\eta)\tau + (1-\eta)^2\tau^2$.
While $F_2(q^{-1})$ is a "lag" filter (it delays low-frequency signal components by τ sampling intervals), the polynomial approximation filters $H_{m,\tau,\eta}(q^{-1})$, $m \geq 1$ are typical "lead" filters - at low frequencies they introduce a positive time shift, i.e. time advance, equal to τ (it is easy to check this property for $m = 1, 2$). This confirms that bias reduction is achieved primarily by means of delay compensation.
The ALF approximation of the debiased algorithm can be obtained by combining (9) with (16), which results in

$$\bar{\omega}(t) \cong K_1(q^{-1})z(t) + K_2(q^{-1})\omega(t) \quad (19)$$

where $K_1(q^{-1}) = F_1(q^{-1})H_{m,\tau,\eta}(q^{-1})$ and $K_2(q^{-1}) = F_2(q^{-1})H_{m,\tau,\eta}(q^{-1})$.
Similarly as it was done in (Tichavský & Händel, 1995), the ALF model can be used to determine the relaxing time of the algorithm, as well as to optimize its design parameters under various specific tracking scenarios.
It is worth noticing that the filter (15) closely resembles the signal processing tool known as Savitzky-Golay smoother (Orfanidis, 1996). In both cases signal estimation is based on polynomial approximation. The differ-

ence lies in estimation details. The Savitzky-Golay approach incorporates the method of sliding window least squares. The resulting filter is noncausal and nonrecursive. The approach presented above exploits the method of exponentially weighted least squares and results in a filter that is causal and recursive.

2.3 Three-step algorithm

In order to improve tracking capability of the generalized adaptive notch filter (4), one can run another algorithm, of the same form, which incorporates the debiased frequency estimates $\bar{\omega}(t)$. Instead of presenting the single-frequency system-oriented debiased version of the algorithm (4), we will turn directly to the multiple frequencies case ($k > 1$).

Denote by $y_i(t) = \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\beta}_i(t) + v(t)$ the output of the ith subsystem of (3), i.e. subsystem associated with the frequency ω_i. If the signals $y_1(t), \dots, y_k(t)$ were available, one could design k independent GANF algorithms each of which would take care of a particular subsystem. Since $\boldsymbol{\theta}(t) = \sum_{i=1}^{k} \boldsymbol{\beta}_i(t)$, the final estimation result could be easily obtained by combining the partial estimates $\widehat{\boldsymbol{\theta}}(t) = \sum_{i=1}^{k} \widehat{\boldsymbol{\beta}}_i(t)$. Even though the signals $y_i(t)$ are not available, one can easily estimate them using the formula

$$\widehat{y}_i(t) = y(t) - \sum_{\substack{m=1 \\ m \neq i}}^{k} \widehat{y}_m(t|t-1)$$

where $\widehat{y}_i(t|t-1) = e^{j\widehat{\omega}_i(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1)$ denotes the predicted value of $y_i(t)$, yielded by the estimation algorithm designed to track parameters of the ith subsystem. Note that after replacing $y_i(t)$ with $\widehat{y}_i(t)$ one obtains $\varepsilon_1(t) = \ldots = \varepsilon_k(t) = y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t) \sum_{i=1}^{k} e^{j\widehat{\omega}_i(t)}\widehat{\boldsymbol{\beta}}_i(t-1) = \varepsilon(t)$ i.e. all subalgorithms are in fact driven by the same "global" prediction error $\varepsilon(t)$.

From the system-analytic point of view, the distributed estimation scheme described above is a parallel structure made up of k identical (from the functional viewpoint) blocks. Each block tracks a particular frequency component of the parameter vector $\boldsymbol{\theta}(t)$. The resulting parallel-form algorithm is summarized below. To add some extra design flexibility, we have equipped each subalgorithm with independently assigned adaptation gains μ_i and γ_i.

pilot filter:

$$\varepsilon(t) = y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t) \sum_{i=1}^{k} e^{j\widehat{\omega}_i(t)}\widehat{\boldsymbol{\beta}}_i(t-1)$$

$$\widehat{\boldsymbol{\beta}}_i(t) = e^{j\widehat{\omega}_i(t)}\widehat{\boldsymbol{\beta}}_i(t-1) + \mu_i \boldsymbol{\Phi}^{-1}\boldsymbol{\varphi}^*(t)\varepsilon(t)$$

$$g_i(t) = \mathrm{Im}\left[\frac{\varepsilon^*(t)e^{j\widehat{\omega}_i(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1)}{\widehat{\boldsymbol{\beta}}_i^{\mathrm{H}}(t-1)\boldsymbol{\Phi}\widehat{\boldsymbol{\beta}}_i(t-1)}\right]$$

$$\widehat{\omega}_i(t+1) = \widehat{\omega}_i(t) - \gamma_i g_i(t)$$

$$i = 1, \dots, k$$

$$\widehat{\boldsymbol{\theta}}(t) = \sum_{i=1}^{k} \widehat{\boldsymbol{\beta}}_i(t) \tag{20}$$

correction filter:

$$\bar{\omega}_i(t) = H_{m,\tau_i,\eta}(q^{-1})\widehat{\omega}_i(t) \tag{21}$$

$$i = 1, \dots, k$$

frequency-guided filter:

$$\bar{\varepsilon}(t) = y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t) \sum_{i=1}^{k} e^{j\bar{\omega}_i(t)}\bar{\boldsymbol{\beta}}_i(t-1)$$

$$\bar{\boldsymbol{\beta}}_i(t) = e^{j\bar{\omega}_i(t)}\bar{\boldsymbol{\beta}}_i(t-1) + \mu_i \boldsymbol{\Phi}^{-1}\boldsymbol{\varphi}^*(t)\bar{\varepsilon}(t)$$

$$i = 1, \dots, k$$

$$\bar{\boldsymbol{\theta}}(t) = \sum_{i=1}^{k} \bar{\boldsymbol{\beta}}_i(t) \tag{22}$$

where $\tau_i = \mu_i/\gamma_i, i = 1, \dots, k$.

Note that the frequency-guided filter does not estimate system frequencies on its own.

When the matrix $\boldsymbol{\Phi}$ is not known, or when it changes (slowly) with time, it can be replaced in (20) and (22) with the following estimate

$$\widehat{\boldsymbol{\Phi}}(t) = \lambda_o \widehat{\boldsymbol{\Phi}}(t-1) + (1-\lambda_o)\boldsymbol{\varphi}^*(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)$$

where $0 < \lambda_o < 1$ denotes a forgetting constant (e.g. $\lambda_o = 0.9$).

3 Computer simulations

Two simulation experiments were arranged to check the system tracking capabilities of the GANF algorithm (20) - (22). The simulated system, inspired by channel estimation applications, was governed by

$$y(t) = \theta(t)u(t) + v(t), \quad \theta(t) = ae^{j\sum_{s=1}^{t}\omega(s)}$$

i.e. it was a single-tap FIR system ($n = 1$) with a single frequency mode ($k = 1$). The weighting coefficient had a constant value $a = 2 - j$. The white 4-QAM sequence was used as the input signal ($u(t) = \pm 1 \pm j$, $\sigma_u^2 = 2$) and the noise was complex Gaussian with variance $\sigma_v^2 = 0.1$ (SNR=20 dB) or $\sigma_v^2 = 2$ (SNR=7 dB).

In the first experiment the instantaneous frequency was changed in a linear way: $\omega(t) = \pi/4 - 0.0001t$. The second experiment was more "realistic": the instantaneous frequency was changed in a sinusoidal fashion: $\omega(t) = [1 + 0.03\sin(\pi t/2000)] \cdot (\pi/4)$. Note that sinusoidal changes only locally can be approximated by the polynomial model (8). All filters were allowed to reach their steady state behavior before the frequency changes were enforced. The obtained results are summarized in Figures 1 and 2.

Figure 1 shows typical trajectories of frequency estimates yielded by the original GANF algorithm (20), and by its debiased versions (22) based on the first-order ($\mu = 0.02, \gamma = 0.0004, \eta = 0.95$) and second-order ($\mu = 0.02, \gamma = 0.0004, \eta = 0.97$) polynomial approximations. Different values of η were adopted for $m = 1$

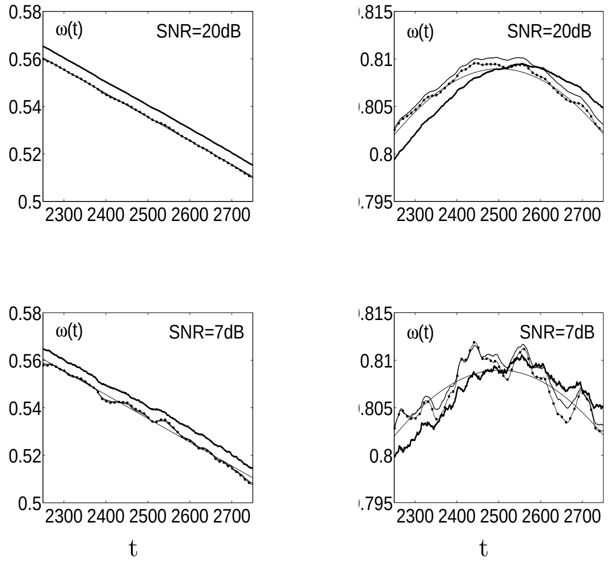

Fig. 1. True signal frequency changes (thin lines) and typical trajectories of frequency estimates yielded by the GANF algorithm (thick lines), and by its frequency-debiased versions based on the first-order (medium-thick lines) and second-order (lines with dots) polynomial approximations.

and $m = 2$ to guarantee that the compared adaptive filters have the same equivalent estimation memory – see (Niedźwiecki, 2000) for more details. Note how a clearly visible delay of the estimated trajectories, with respect to the true trajectories, is reduced by means of debiasing.

Tracking capabilities of the compared algorithms were measured in terms of the accumulated frequency estimation errors $\Sigma_\omega = \sum_{t=2001}^{4000} [\widehat{\omega}(t) - \omega(t)]^2$ and the accumulated excess prediction errors $\Sigma_\varepsilon = \sum_{t=2001}^{4000} [|\,\varepsilon(t)|^2 - \sigma_v^2]$. All filters were allowed to reach their steady state behavior before their analysis/comparison was carried out. To reduce the number of degrees of freedom, the frequency adaptation gain γ was set to μ^2 – see (Niedźwiecki & Kaczmarek, 2006b). The forgetting factor η was fixed at $\eta = 0.95$ for $m = 1$, and at $\eta = 0.97$ for $m = 2$.

Figure 2 shows how ensemble averages of both error statistics (obtained for 25 different realizations of measurement noise) depend on the choice of μ. As expected debiasing based on the first-order (linear) polynomial approximation yields improved tracking results, both in terms of the minimum achievable errors and, more importantly, in terms of the algorithm's robustness to the choice of μ. Usefulness of the second-order (parabolic) approximation depends on the signal-to-noise ratio. For high SNR (20dB) application of the second-order approximation further improves tracking results. However, this is not true any more for low SNR (7dB). This is not a surprise as high-order polynomial approximations are known to be sensitive to noise.

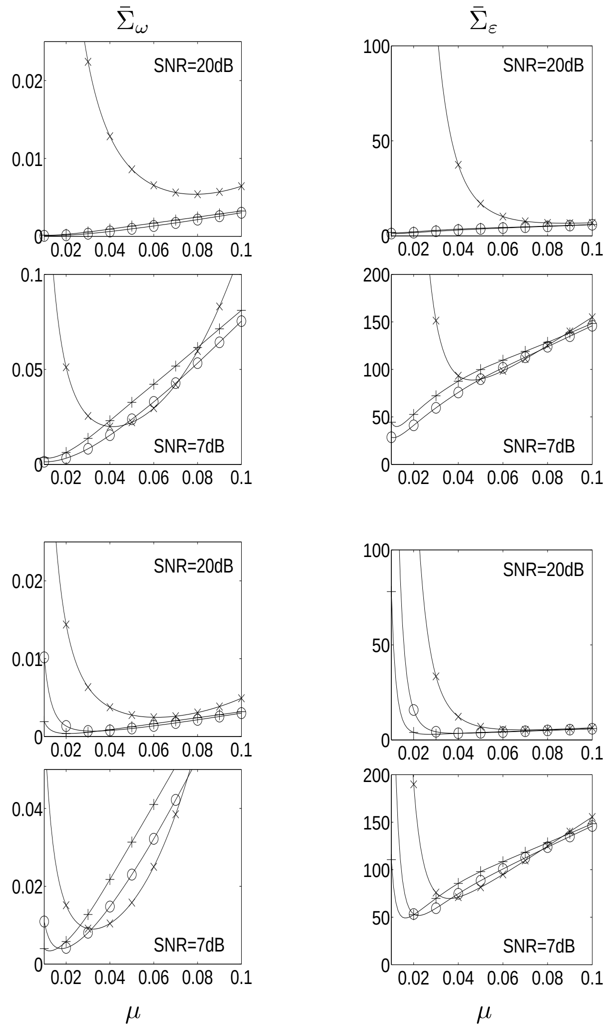

Fig. 2. Dependence of the averaged sums of the squared frequency estimation errors $\bar{\Sigma}_\omega$ and excess prediction errors $\bar{\Sigma}_\varepsilon$ on the adaptation gain μ. Comparison involves the estimates yielded by the original GANF algorithm ($\times$) and by its frequency-debiased versions proposed in the paper, based on the first-order ($\circ$) and second-order ($+$) polynomial approximation. The top four plots corresponds to linear frequency changes and the bottom four plots – to sinusoidal changes. All plots were evaluated on a grid of 90 equidistant values of μ.

4 Conclusion

We have shown that when system frequencies change in a smooth way, the frequency estimates yielded by the GANF algorithm can be effectively debiased. The proposed solution is a cascade of three filters. The "pilot" generalized adaptive notch filter provides preliminary (biased) frequency estimates. The estimates yielded by the pilot algorithm are passed through a correction filter and fed into the third algorithm - the "frequency-guided" generalized adaptive notch filter. Frequency debiasing allows for improvement of the tracking performance of generalized adaptive notch filters.

References

Bakkoury, J., D. Roviras, M. Ghogho & F. Castanie (2000). Adaptive MLSE receiver over rapidly fading channels. *Signal Processing*, **vol. 80**, 1347–1360.

Niedźwiecki, M. (2000). *Identification of Time-varying Processes.* Wiley. New York.

Niedźwiecki, M. & P. Kaczmarek (2004). Generalized adaptive notch filters. *Proc. 2004 IEEE Int. Conf. on Acoustics, Speech and Signal Proc.*, Montreal, Canada, II-657–II-660.

Niedźwiecki, M. & P. Kaczmarek (2006a). Tracking analysis of a generalized adaptive notch filter. *IEEE Trans. on Signal Processing*, **vol. 54**, 304–314.

Niedźwiecki, M. & P. Kaczmarek (2006b). Generalized adaptive notch filter with a self-optimization capability. To appear in *IEEE Trans. on Signal Processing.*

Orfanidis, S. (1996). *Introduction to Signal Processing.* Prentice Hall. Englewood Cliffs.

Pei, S.-C. & C.-C. Tseng (1994). Complex adaptive IIR notch filter algorithm and its application. *IEEE Trans. Circ. and Syst. II: Analog and Digital Signal Proc.*, **vol. 41**, 158–163.

Tichavský P. and P. Händel (1995). Two algorithms for adaptive retrieval of slowly time-varying multiple cisoids in noise. *IEEE Trans. on Signal Processing*, **vol. 43**, 1116–1127.

Tichavský P. & A. Nehorai (1997). Comparative study of four adaptive frequency trackers. *IEEE Trans. on Signal Processing*, **vol. 45**, 1473–1484.

Tichavský P. and P. Händel (1997a). Recursive estimation of frequencies and frequency rates of multiple cisoids in noise. *Signal Processing*, **vol. 58**, 117–129.

Tichavský P. and P. Händel (1997b). Recursive estimation of linearly or harmonically modulated frequencies of multiple cisoids in noise. *Proc. Int. Conf. on Acoustics, Speech and Signal Processing*, Munich, Germany, 1925–1928.

Tsatsanis, M.K. & G.B. Giannakis (1996). Modeling and equalization of rapidly fading channels. *Int. J. Adaptive Contr. Signal Processing*, **vol. 10**, 159–176.

APPENDIX

According to (12) and (13) the steady state relationship between $\bar{\omega}(t)$ and $\widehat{\omega}(t)$ can be written down in the form

$$\bar{\omega}(t) = \sum_{i=0}^{\infty} h_{m,\tau,\eta}(t,i)\widehat{\omega}(t-i) \qquad (23)$$

where

$$h_{m,\tau,\eta}(t,i) = = \eta^i \mathbf{f}^{\mathrm{T}}(t+\tau)\left[\sum_{i=0}^{\infty}\eta^i \mathbf{f}(t-i)\mathbf{f}^{\mathrm{T}}(t-i)\right]^{-1} \mathbf{f}(t-i) \qquad (24)$$

To avoid matrix inversion in (24), we will express $h_{m,\tau,\eta}(t,i)$ in terms of the orthonormal basis functions $\widetilde{f}_0(i),\ldots,\widetilde{f}_m(i)$ of the subspace F spanned by $f_0(i),\ldots,f_m(i)$. Denote by $\widetilde{\mathbf{f}}(i) = [\widetilde{f}_0(i),\ldots,\widetilde{f}_m(i)]^{\mathrm{T}}$ the basis vector of F which fulfills the following weighted orthonormality condition

$$\sum_{i=0}^{\infty}\eta^i \widetilde{\mathbf{f}}(i+1)\widetilde{\mathbf{f}}^{\mathrm{T}}(i+1) = \mathbf{I}_{m+1} \qquad (25)$$

where $\mathbf{I}_{m+1}$ is the $(m+1)\times(m+1)$ identity matrix. To evaluate $h_{1,\tau}(t,i)$ and $h_{2,\tau}(t,i)$ we need to know the first three orthogonalized basis functions of the subspace spanned by powers of time. Using the Gram-Schmidt procedure one arrives at

$$\begin{aligned}
\widetilde{f}_0(i) &= \sqrt{1-\eta} \\
\widetilde{f}_1(i) &= \sqrt{\frac{1-\eta}{\eta}}\,[(1-\eta)(i-1)-\eta] \\
\widetilde{f}_2(i) &= \frac{\sqrt{1-\eta}}{2\eta}[(1-\eta)^2(i-1)^2 \\
&\quad + (3\eta^2-2\eta-1)(i-1)+2\eta^2]
\end{aligned} \qquad (26)$$

Suppose there exists a nonsingular $(m+1)\times(m+1)$ matrix $\mathbf{D}(t)$ such that $\mathbf{D}(t)\mathbf{f}(t-i) = \widetilde{\mathbf{f}}(i+1), \forall t, i$. Then (24) can be rewritten in the form

$$\begin{aligned}
&h_{m,\tau,\eta}(t,i) = \eta^i \mathbf{f}^{\mathrm{T}}(t+\tau)\mathbf{D}^{\mathrm{T}}(t)\times \\
&\times\left[\sum_{i=0}^{\infty}\eta^i\mathbf{D}(t)\mathbf{f}(t-i)\mathbf{f}^{\mathrm{T}}(t-i)\mathbf{D}^{\mathrm{T}}(t)\right]^{-1}\mathbf{D}(t)\,\mathbf{f}(t-i) \\
&= \eta^i\widetilde{\mathbf{f}}^{\mathrm{T}}(1-\tau)\left[\sum_{i=0}^{\infty}\eta^i\widetilde{\mathbf{f}}(i+1)\widetilde{\mathbf{f}}^{\mathrm{T}}(i+1)\right]^{-1}\widetilde{\mathbf{f}}(i+1) \\
&= \eta^i\widetilde{\mathbf{f}}^{\mathrm{T}}(1-\tau)\widetilde{\mathbf{f}}(i+1) = h_{m,\tau,\eta}(i)
\end{aligned} \qquad (27)$$

which is much easier to handle.
Consider the case where $m = 1$ (linear approximation). Let

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix},$$

$$\mathbf{C} = \sqrt{\frac{1-\eta}{\eta}}\begin{bmatrix} \sqrt{\eta} & 0 \\ (1-\eta)\tau - 1 & 1-\eta \end{bmatrix}$$

Observe that: $\mathbf{A}\mathbf{f}(t) = \mathbf{f}(t-1)$, $\mathbf{B}\mathbf{f}(t) = \mathbf{f}(-t)$ and $\mathbf{C}\mathbf{f}(t) = \widetilde{\mathbf{f}}(t)$, $\forall t$. Therefore, for all t, it holds that $\mathbf{CBA}^{t+1}\mathbf{f}(t-i) = \mathbf{CBf}(-i-1) = \mathbf{Cf}(i+1) = \widetilde{\mathbf{f}}(i+1)$, which means that one can set $\mathbf{D}(t) = \mathbf{CBA}^{t+1}$. In the analogous way one can construct the matrix $\mathbf{D}(t)$ for $m > 1$.
Combining (26) with (27) one obtains

$$h_{1,\tau,\eta}(i) = \eta^i(1-\eta)\left[1 - \frac{(1-\eta)\tau+\eta}{\eta}[(1-\eta)i-\eta]\right]$$

which leads, after applying the Z-transform, to (17). In the similar way, after elementary but tedious calculations, one arrives at (18).