

# GENERALIZED ADAPTIVE NOTCH FILTERS WITH FREQUENCY DEBIASING

Maciej Niedźwiecki, Adam Sobociński

Faculty of Electronics, Telecommunications and Computer Science  
 Department of Automatic Control, Gdańsk University of Technology  
 Narutowicza 11/12, Gdańsk, Poland  
 Tel: + 48 58 3472519, Fax: +48 58 3416132  
 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. Belonging to the class of causal adaptive filters, the generalized adaptive notch filtering algorithms yield biased frequency estimates. We show that this bias can be removed, or at least substantially reduced. The only price paid for the resulting improvement of the filter's tracking performance is in terms of a decision delay, which must be incorporated in the adaptive loop. Since decision delay is acceptable in many practical applications, the proposed bias/delay trade-off is an attractive alternative to the classical bias/variance compromise.

## 1. INTRODUCTION

The term "generalized adaptive notch filter" (GANF) was coined in [1] and denotes adaptive filtering algorithm capable of identification/tracking of quasi-periodically varying systems. Complex-valued quasi-periodically varying systems are governed by

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

where  $t = 1, 2, \dots$  denotes the normalized discrete time,  $y(t)$  denotes the system output,  $\varphi(t) = [\varphi_1(t), \dots, \varphi_n(t)]^T$  is the regression vector and  $v(t)$  is a complex white noise of variance  $\sigma_v^2$ . We will assume that  $E[v_R^2(t)] = E[v_I^2(t)] = \sigma_v^2/2$ ,  $E[v_R(t)v_I(s)] = 0$ ,  $\forall t, s$ , where  $v_R(t) = \text{Re}[v(t)]$ ,  $v_I(t) = \text{Im}[v(t)]$ . Finally,  $\theta(t) = [\theta_1(t), \dots, \theta_n(t)]^T$  denotes the vector of time varying coefficients, modeled as weighted sums of complex exponentials

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

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  $\alpha_i(t) = [a_{1i}(t), \dots, a_{ni}(t)]^T$  the vector of system coefficients associated with a particular frequency  $\omega_i$  and let  $\beta_i(t) = f_i(t) \alpha_i(t)$ , where  $f_i(t) = e^{j \sum_{s=1}^t \omega_i(s)}$ . Using the shorthand notation introduced above, system equation (1) can be rewritten in the form

$$y(t) = \sum_{i=1}^k \varphi^T(t) \beta_i(t) + v(t) \quad (3)$$

When the sequence of regression vectors  $\{\varphi(t)\}$  is wide-sense stationary and persistently exciting, with known covariance matrix  $\Phi = E[\varphi^*(t) \varphi^T(t)] > 0$ , the normalized steady state single-frequency ( $k = 1$ ) version of the GANF algorithm presented in [2], [3] can be written down in the form

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

Tracking properties of this algorithm are determined by two user-dependent tuning coefficients: the adaptation gain  $\mu$ ,  $0 < \mu \ll 1$ , which controls the rate of amplitude adaptation, and another adaptation gain  $\gamma$ ,  $0 < \gamma \ll 1$ , 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 blocks (subalgorithms), given by (4), into appropriately designed parallel or cascade structures - see [1] and Section 3.3.

Generalized adaptive notch filters can be applied to equalization of rapidly fading telecommunication channels [4], [5]. In a special case where  $n = 1$  and  $\varphi(t) = 1, \forall t$ , the model (1) - (2) becomes a description of a noisy nonstationary multi-frequency signal  $s(t) = \theta(t)$ . In this case generalized adaptive notch filters turn into "ordinary" adaptive notch filters (ANFs), the algorithms used for extraction or elimination of sinusoidal signals buried in noise - see [6], [7] and the references therein.

## 2. TRACKING PROPERTIES OF THE GANF ALGORITHM

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

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

Note that the assumed model of parameter variation can be rewritten in an explicit form as  $\theta(t) = \beta(t) = \beta_o e^{j \sum_{s=1}^t \omega(s)}$ , where  $\beta_o = \beta(0)$ . Since it holds that  $\|\beta(t)\|^2 = \|\beta_o\|^2 = \text{const}$ ,  $\forall t$ , parameter changes of the analyzed system can be attributed exclusively to changes of the instantaneous frequency  $\omega(t)$ .

Let

$$b^2 = \beta^H(t) \Phi \beta(t) = \beta_o^H \Phi \beta_o$$

and

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

It can be shown that  $\{z(t)\}$  is a real-valued white noise with variance

$$\sigma_z^2 = \frac{\sigma_v^2}{2b^2} = \frac{1}{2\text{SNR}}$$

Finally, denote by  $\Delta \hat{\omega}(t) = \hat{\omega}(t) - \omega(t)$  the frequency estimation error and let  $w(t) = \omega(t) - \omega(t-1)$ .

The frequency tracking properties of (4) were analyzed in [2], [3] using the direct averaging approach [8] and the approximating linear filtering technique proposed in [6]. Approximating linear filters characterize the relation between the sequences of estimation errors and the sequences of measurement noise  $v(t)$  and of the one-step changes of the true frequency  $w(t)$ , provided that the analyzed algorithms operate in a neighborhood of their equilibrium state.

As shown in [2], [3] the frequency estimation errors yielded by the generalized adaptive notch filtering algorithm (4), applied to the system obeying (5), can be approximately described by the following linear equation

$$\Delta \hat{\omega}(t) = E_1(q^{-1})z(t) + E_2(q^{-1})w(t) \quad (6)$$

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

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

and  $\lambda = 1 - \mu$ ,  $\delta = 1 - \gamma$ . The filters  $E_1(q^{-1})$  and  $E_2(q^{-1})$  are asymptotically stable for any  $\lambda$  and  $\delta$  from the interval  $(0,1)$ .

It was also shown that when the frequency  $\omega(t)$  evolves according to the random walk model the optimally tuned algorithm (4) is (under Gaussian assumptions and in the range of applicability of the ALF approximation) a statistically efficient estimation procedure, i.e. the corresponding mean-squared frequency tracking error achieves its lower bound set by the Cramér-Rao inequality [2], [3].

### 3. FREQUENCY DEBIASING

#### 3.1 Frequency bias analysis

When the instantaneous frequency drifts according to the random walk model, i.e. when  $\{w(t)\}$  is a white noise sequence, one obtains  $E[\hat{\omega}(t)] = \omega(t)$ . This follows directly from (6) and means that, in the case considered, the algorithm (4) yields unbiased frequency estimates (at least up to the higher-order terms, neglected in the course of the ALF analysis).

It is important to realize that unbiasedness holds *only* under the random walk hypothesis, i.e. this property does not extend to other, perhaps more realistic, models of frequency variation. As an example, consider the situation where the instantaneous frequency changes linearly with time (linear chirp signal), that is

$$w(t) = \omega(t) - \omega(t-1) = \delta_\omega, \quad \forall t \quad (8)$$

By taking expectations of both sides of (6) one arrives at

$$E[\Delta \hat{\omega}(t)] = -\frac{\mu}{\gamma} \delta_\omega \quad (9)$$

which shows that the frequency estimates are in this case biased. This is a typical situation. The bias is caused by the fact that parameter estimates yielded by causal adaptive filters always lag behind the true signal/system parameters [9]. A typical way of increasing tracking capabilities of adaptive notch filters is by means of automatic gain/bandwidth tuning – in the signal case an interesting overview of different approaches to this problem can be found in [10]. Irrespective of tuning principles, all solutions mentioned above have the same main feature - they try to balance the estimation bias and the estimation variance. In order to achieve this, they increase adaptation gains when signal parameters change faster, and decrease adaptation gains when signal parameters slow down. From the qualitative viewpoint the approach advocated in this paper is quite different. We will show that the frequency bias can be removed, or at least significantly reduced, if the estimation delay introduced by the algorithm (4) is compensated. Since the bias can be eliminated without increasing the estimation variance, the resulting algorithm shows considerably improved tracking performance.

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

$$\hat{\omega}(t) = F_1(q^{-1})z(t) + F_2(q^{-1})\omega(t) \quad (10)$$

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

$$F_2(q^{-1}) = \frac{(1-\delta)q^{-1}}{1-(\lambda+\delta)q^{-1}+\lambda q^{-2}} \quad (11)$$

Denote by  $\bar{\omega}(t)$  the debiased (approximately) frequency estimate. The simplest solution is to set

$$\bar{\omega}(t-l) = \hat{\omega}(t) \quad (12)$$

where  $l$  is an integer measure of a delay introduced by the filter  $F_2(q^{-1})$ . We will show that  $l$  can be evaluated from

$$l = \text{int} \left[ \frac{\mu}{\gamma} \right] \quad (13)$$

where  $\text{int}[x]$  denotes an integer number closest to  $x$ .

First, consider the case of linear frequency changes. Note that the relationship (8) can be rewritten in an explicit form as  $\omega(t) = \omega_o + \delta_\omega t$ . According to (9), it holds that

$$E[\hat{\omega}(t)] = \omega_o + \delta_\omega \left( t - \frac{\mu}{\gamma} \right) = \omega(t - \tau)$$

where

$$\tau = \frac{\mu}{\gamma} \quad (14)$$

which leads directly to (13).

The second argument in favor of (13) comes from the classical filtering theory. The time-shifting properties of a lowpass filter  $F(e^{j\omega}) = A(\omega)e^{j\phi(\omega)}$  can be roughly characterized by its nominal (low-frequency) delay time<sup>1</sup>

$$\tau_{\text{nom}} = \lim_{\omega \rightarrow 0} \tau_p(\omega) = \lim_{\omega \rightarrow 0} \tau_g(\omega) \quad (15)$$

<sup>1</sup>  $\omega$  stands here for the standard Fourier-domain frequency variable and *not* for the angular frequency of the analyzed signal - the latter quantity is denoted in this paper by  $\omega(t)$ .

where  $\tau_p(\omega) = -\phi(\omega)/\omega$  denotes the so-called phase delay, used as a measure of delay at specific frequencies, and  $\tau_g(\omega) = -d\phi(\omega)/d\omega$  denotes the group delay, serving as a measure of delay over a band (group) of frequencies. Note that for the values of  $\omega$  close to zero it holds that  $\sin \omega \cong \omega$ ,  $\cos \omega \cong 1$ , leading to

$$F_2(e^{j\omega}) \cong \frac{-j\gamma\omega}{\gamma + j(\mu - \gamma)\omega} = A_2(\omega)e^{j\phi_2(\omega)}$$

where

$$\begin{aligned} A_2(\omega) &= \frac{\gamma^2\omega^2}{\gamma^2 + (\mu - \gamma)^2\omega^2} \\ \phi_2(\omega) &= -\omega - \arctan \frac{\mu - \gamma}{\gamma} \omega \end{aligned}$$

The nominal delay of  $F_2(q^{-1})$ , evaluated according to (15), is equal to  $\tau_{\text{nom}} = \mu/\gamma$ , which is identical with (14).

### 3.2 Two-step algorithm

The final solution we propose is a cascade of two filters: the pilot<sup>2</sup> adaptive notch filter (4), which yields preliminary frequency estimates  $\hat{\omega}(t)$ , and the frequency-guided adaptive notch filter, fed with the debiased estimates (12), which works out the final signal estimates  $\hat{s}(t)$ .

Since the proposed debiasing scheme is noncausal, the frequency-guided algorithm will in fact operate on a time-delayed input data sequence. The resulting decision delay of  $k$  sampling intervals is acceptable in majority of signal processing applications, such as adaptive line enhancement or adaptive noise canceling. The proposed two-step algorithm can be summarized as follows

*pilot filter:*

$$\begin{aligned} \varepsilon(t) &= y(t) - e^{j\hat{\omega}(t)}\hat{\beta}(t-1) \\ \hat{\beta}(t) &= e^{j\hat{\omega}(t)}\hat{\beta}(t-1) + \mu\varepsilon(t) \\ g(t) &= \text{Im} \left[ \frac{\varepsilon^*(t)e^{j\hat{\omega}(t)}}{\hat{\beta}^*(t-1)} \right] \\ \hat{\omega}(t+1) &= \hat{\omega}(t) - \gamma g(t) \\ \hat{s}(t) &= \hat{\beta}(t) \end{aligned} \quad (16)$$

*frequency-guided filter:*

$$\begin{aligned} \bar{\varepsilon}(t-l) &= y(t-l) - e^{j\hat{\omega}(t)}\bar{\beta}(t-l-1) \\ \bar{\beta}(t-l) &= e^{j\hat{\omega}(t)}\bar{\beta}(t-l-1) + \mu\bar{\varepsilon}(t-l) \\ \bar{s}(t-l) &= \bar{\beta}(t-l) \end{aligned} \quad (17)$$

### 3.3 Extension to the multiple frequencies case

Denote by

$$y_i(t) = \varphi^T(t)\beta_i(t) + v(t)$$

the output of the  $i$ th subsystem of (3), i.e. subsystem associated with the frequency  $\omega_i$ . 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  $\theta(t) = \sum_{i=1}^k \beta_i(t)$ , the final estimation result could be easily obtained by combining the partial estimates

$$\hat{\theta}(t) = \sum_{i=1}^k \hat{\beta}_i(t)$$

Even though the signals  $y_i(t)$  are not available, one can easily estimate them using the formula

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

where  $\hat{y}_i(t|t-1) = e^{j\hat{\omega}_i(t)}\varphi^T(t)\hat{\beta}_i(t-1)$  denotes the predicted value of  $y_i(t)$ , yielded by the estimation algorithm designed to track parameters of the  $i$ th subsystem. Note that after replacing  $y_i(t)$  with  $\hat{y}_i(t)$  one obtains  $\varepsilon_1(t) = \dots = \varepsilon_k(t) = y(t) - \varphi^T(t)\sum_{i=1}^k e^{j\hat{\omega}_i(t)}\hat{\beta}_i(t-1) = \varepsilon(t)$  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  $\theta(t)$ .

The resulting parallel-form algorithm is summarized below. To add some extra design flexibility, we have equipped each subalgorithm with independently assigned adaptation gains  $\mu_i$  and  $\gamma_i$ .

*pilot filter:*

$$\begin{aligned} \varepsilon(t) &= y(t) - \varphi^T(t) \sum_{i=1}^k e^{j\hat{\omega}_i(t)}\hat{\beta}_i(t-1) \\ \hat{\beta}_i(t) &= e^{j\hat{\omega}_i(t)}\hat{\beta}_i(t-1) + \mu_i\Phi^{-1}\varphi^*(t)\varepsilon(t) \\ g_i(t) &= \text{Im} \left[ \frac{\varepsilon^*(t)e^{j\hat{\omega}_i(t)}\varphi^T(t)\hat{\beta}_i(t-1)}{\hat{\beta}_i^H(t-1)\Phi\hat{\beta}_i(t-1)} \right] \\ \hat{\omega}_i(t+1) &= \hat{\omega}_i(t) - \gamma_i g_i(t) \\ & \quad i = 1, \dots, k \\ \hat{\theta}(t) &= \sum_{i=1}^k \hat{\beta}_i(t) \end{aligned} \quad (18)$$

*frequency-guided filter:*

$$\begin{aligned} \bar{\varepsilon}(t-l) &= y(t-l) \\ & - \varphi^T(t-l) \sum_{i=1}^k e^{j\hat{\omega}_i(t-l+l_i)}\bar{\beta}_i(t-l-1) \\ \bar{\beta}_i(t-l) &= e^{j\hat{\omega}_i(t-l+l_i)}\bar{\beta}_i(t-l-1) \\ & + \mu_i\Phi^{-1}\varphi^*(t-l)\bar{\varepsilon}(t-l) \\ & \quad i = 1, \dots, k \\ \bar{\theta}(t-l) &= \sum_{i=1}^k \bar{\beta}_i(t-l) \end{aligned} \quad (19)$$

where  $l = \max\{l_1, \dots, l_k\}$  and  $l_i = \text{int}[\mu_i/\gamma_i]$ ,  $i = 1, \dots, k$ .

<sup>2</sup>In this paper the term ‘‘pilot filter’’ has a different meaning than the analogous term in [10].

**Remark 1**

When the matrix  $\Phi$  is not known, or when it changes with time, it can be replaced in (18) and (19) with the following estimate

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

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

**Remark 2**

When  $k > 1$ , the ALF analysis carried out above is valid as long as the estimated frequencies remain well-separated. When frequency tracking is disabled ( $\gamma = 0, \widehat{\omega}(0) = \omega_o$ ), the first-order parameter tracking properties of (4) are characterized by the relationship  $E[\widehat{\theta}(t)] \cong T(q^{-1})\theta(t)$  where  $T(q^{-1}) = \mu/[1 - (1 - \mu)e^{j\omega_o}q^{-1}]$  is a narrow-band extraction filter centered at the frequency  $\omega_o$ , with bandwidth  $B_{3dB} \cong 2\mu$ . Therefore, the frequency separation condition for (18) can be approximately expressed in the form

$$|\omega_i(t) - \omega_j(t)| > \mu_i + \mu_j, \quad \forall i \neq j, \quad \forall t \quad (20)$$

When the above condition is not fulfilled, the bandwidths of the extraction filters partially overlap, making the behavior of the entire structure difficult to predict. It should be stressed, however, that the proposed algorithm works pretty well even if the frequency separation condition is violated.

**Remark 3**

So far we have been assuming that the adaptation gains in (18)-(19) are fixed. In order to optimize tracking performance of a GANF filter one can equip it with an additional gain-tuning loop. Such self-optimizing version of the algorithm, capable of automatic tuning of its adaptation gains, was proposed in [11]. Quite obviously, when the adaptation gains are time-varying, the estimation delay  $\tau$  is also a time-dependent quantity. The value of  $\tau(t)$  can be determined by examining the “response” of the GANF algorithm to linear frequency changes.

**4. COMPUTER SIMULATIONS**

The aim of the simulation experiment was to check the system tracking capabilities of the GANF algorithm (18) - (19). The simulated system, inspired by channel estimation applications, was governed by

$$y(t) = \theta(t)u(t) + v(t)$$

where

$$\theta(t) = a_1 e^{j\sum_{s=1}^t \omega_1(s)} + a_2 e^{j\sum_{s=1}^t \omega_2(s)}$$

i.e. it was a single-tap FIR system ( $n = 1$ ) with two modes of parameter variation ( $k = 2$ ). The weighting coefficients had constant values  $a_1 = 2 - j$  and  $a_2 = 1 + 2j$ . 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  $\sigma_v^2 = 1$  (SNR=13 dB).

Figure 1 shows evolution of the instantaneous frequencies  $\omega_1(t)$  and  $\omega_2(t)$ , and trajectories of the corresponding frequency estimates obtained from the regular multiple-frequency GANF algorithm (18) for  $\mu = \mu_1 = \mu_2 = 0.03$  and  $\gamma = \gamma_1 = \gamma_2 = 0.00045$ . The plots gathered in

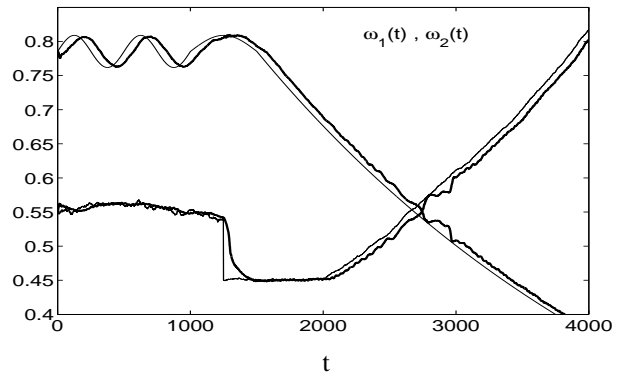


Figure 1: True system frequency changes (thin lines) and typical trajectories of frequency estimates (thick lines) yielded by the GANF algorithm ( $\mu = 0.03$ , SNR=13 dB).

Figure 2 allow one to compare tracking performance of the regular algorithm and of its debiased version. To obtain meaningful quantitative results the comparison was made in five analysis intervals:  $T_1 = [501, 1000]$ ,  $T_2 = [1001, 1500]$ ,  $T_3 = [1501, 2000]$ ,  $T_4 = [2001, 2500]$  and  $T_5 = [2501, 3000]$ , corresponding to different types of frequency changes. Since the frequency trajectories  $\omega_1(t)$  and  $\omega_2(t)$  intersect at instant  $t = 2750$ , the frequency separation condition (20) is not fulfilled in the interval  $T_5$ .

The accumulated frequency tracking errors were defined as

$$\Sigma_{\omega} = \sum_{t \in T} \min \{ [\widehat{\omega}_1(t) - \omega_1(t)]^2 + [\widehat{\omega}_2(t) - \omega_2(t)]^2, \\ [\widehat{\omega}_1(t) - \omega_2(t)]^2 + [\widehat{\omega}_2(t) - \omega_1(t)]^2 \}$$

and the system tracking errors were evaluated according to

$$\Sigma_{\theta} = \sum_{t \in T} |(\widehat{\theta}(t) - \theta(t))u(t)|^2 = \sigma_u^2 \sum_{t \in T} |\widehat{\theta}(t) - \theta(t)|^2$$

The same settings were adopted for both subalgorithms:  $\mu = \mu_1 = \mu_2$ ,  $\gamma = \gamma_1 = \gamma_2$ . To reduce the number of design degrees of freedom, the frequency adaptation gain  $\gamma$  was set to  $\mu^2/2$  – see [3]. The plots gathered in Figure 2 show how performance of the compared algorithms – the original GANF algorithm (4) and its debiased version (18) - (19) – depends on the choice of the adaptation gain  $\mu$  for different types of frequency changes. The plots depict ensemble averages (corresponding to 50 realizations of measurement noise) of the sums of the squared frequency and signal estimation errors. In all cases frequency debiasing led to improved tracking results.

The ruggedness of the plots corresponding to the interval  $T_5$  is caused by the frequency switching effect, which can be observed when the separation condition (20) is not fulfilled. When passbands of two (G)ANF filters overlap, the corresponding estimates, say  $\widehat{\omega}_1(t)$  and  $\widehat{\omega}_2(t)$ , “switch” at random moments between  $\omega_1(t)$  and  $\omega_2(t)$ . This means that in some time intervals  $\widehat{\omega}_1(t)$  follows  $\omega_1(t)$  and  $\widehat{\omega}_2(t)$  follows  $\omega_2(t)$ , while in some other intervals  $\widehat{\omega}_1(t)$  follows  $\omega_2(t)$  and  $\widehat{\omega}_2(t)$  follows  $\omega_1(t)$ . Frequency switching produces jitter which can be seen at both error plots.

Summing up, frequency debiasing allows one to improve tracking capabilities of GANF algorithms. First, irrespective of the choice of design variables  $\mu$  and  $\gamma$ , the corrected

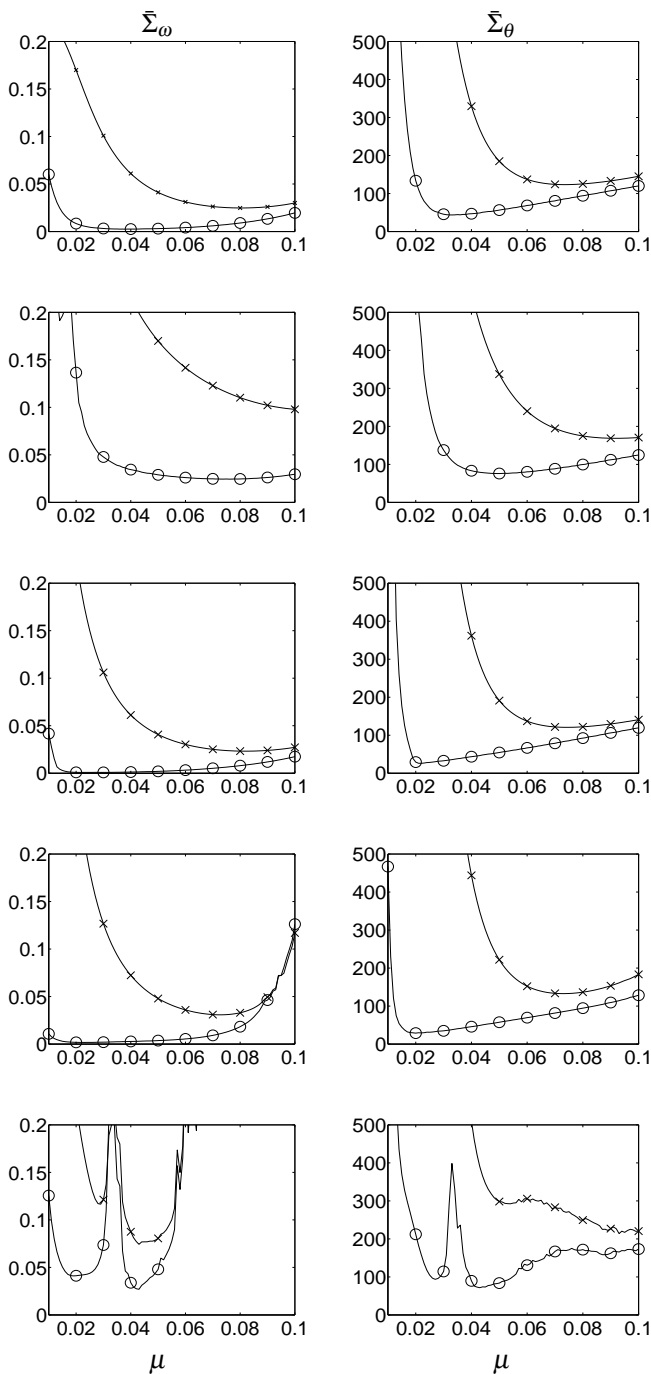


Figure 2: Dependence of the averaged sums of the squared frequency estimation errors  $\bar{\Sigma}_\omega$  and system estimation errors  $\bar{\Sigma}_\theta$  on the adaptation gain  $\mu$ . Comparison involves the estimates yielded by the original GANF algorithm ( $\times$ ) and by its frequency-debiased version proposed in the paper ( $\circ$ ). In each pair of figures, corresponding to one of 5 analysis intervals described in the paper ( $T_1$ ,  $T_2$ ,  $T_3$ ,  $T_4$  and  $T_5$  – from top to bottom), the left figure depicts accumulated frequency errors and the right figure – the accumulated system errors. All plots were evaluated on a grid of 100 equidistant values of  $\mu$ .

estimates are always more accurate than the original estimates. Second, and perhaps more importantly from the practical viewpoint, the debiased algorithms are more robust to the choice of design variables.

## 5. CONCLUSION

Bias (lag) errors seriously limit tracking capabilities of adaptive filters. We have shown that frequency biases, which arise in generalized adaptive notch filtering (GANF) algorithms, can be significantly reduced by incorporating in the adaptive loop a judiciously chosen decision delay. Such delay is acceptable in many practical applications. The proposed solution is a cascade of two filters. The “pilot” generalized adaptive notch filter provides preliminary (biased) frequency estimates. The estimates yielded by the pilot algorithm are fed into the second algorithm – the “frequency-guided” generalized adaptive notch filter, which operates on a delayed data sequence. We have shown that frequency debiasing improves tracking performance of GANF algorithms and increases their robustness to the choice of design parameters.

## REFERENCES

- [1] M. Niedźwiecki and P. Kaczmarek, “Generalized adaptive notch filters,” *Proc. 2004 IEEE Int. Conf. on Acoustics, Speech and Signal Proc.*, Montreal, Canada, 2004, pp. II-657–II-660.
- [2] M. Niedźwiecki, “On frequency tracking properties of a generalized notch filter,” *Proc. 44th IEEE Conference on Decision and Control and European Control Conference*, Seville, Spain, 2005, pp. 4994–4999.
- [3] M. Niedźwiecki and P. Kaczmarek, “Tracking analysis of a generalized adaptive notch filter,” *IEEE Trans. on Signal Processing*, vol. 54, pp. 304–314, 2006.
- [4] M.K. Tsatsanis and G.B. Giannakis, “Modeling and equalization of rapidly fading channels,” *Int. J. Adaptive Contr. Signal Processing*, vol. 10, 1996, pp. 159–176.
- [5] J. Bakkoury, D. Roviras, M. Ghogho and F. Castanie, “Adaptive MLSE receiver over rapidly fading channels,” *Signal Processing*, vol. 80, 2000, pp. 1347–1360.
- [6] P. Tichavský and P. Händel, “Two algorithms for adaptive retrieval of slowly time-varying multiple cisoids in noise,” *IEEE Trans. on Signal Processing*, vol. 43, pp. 1116–1127, 2005.
- [7] P. Tichavský and A. Nehorai, “Comparative study of four adaptive frequency trackers,” *IEEE Trans. on Signal Processing*, vol. 45, pp. 1473–1484, 1997.
- [8] K.J. Kushner, *Approximation and Weak Convergence Methods for Random Processes with Applications to Stochastic System Theory*, MIT Press, 1984.
- [9] M. Niedźwiecki, *Identification of Time-varying Processes*, Wiley, 2000.
- [10] Y.-C.Lim, Y.-X. Zou and N. Zheng, “A piloted adaptive notch filter,” *IEEE Trans. on Signal Processing*, vol. 53, pp. 1310–1323, 2005.
- [11] M. Niedźwiecki and P. Kaczmarek, “Generalized adaptive notch filter with a self-optimization capability,” 2005, to appear in *IEEE Trans. on Signal Processing*.