Generalized Adaptive Notch Filter With a Self-Optimization Capability

Maciej Niedźwiecki and Piotr Kaczmarek

***Abstract*—The paper presents a self-optimizing version of a generalized adaptive notch filter (GANF). 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. The tracking properties of a GANF algorithm depend on two adaptation gains, which should be chosen so as to match the degree of nonstationarity of the identified system. First, an analytical study of a tracking performance of a GANF algorithm is presented. Then, based on the obtained theoretical results, a self-optimizing GANF algorithm is proposed, capable of automatic tuning of its adaptation gains.**

***Index Terms*—Adaptive notch filtering, frequency estimation, system identification, time-varying processes.**

I. Introduction

A. Problem Statement

GENERALIZED adaptive notch filters [1]–[3] were designed for the purpose of identification/tracking of quasi-periodically varying complex-valued systems, i.e., systems 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 $t=1,2,\dots$ denotes the normalized discrete time, $y(t)$ denotes the system output, $\boldsymbol{\varphi}(t) = [\varphi_1(t),\dots,\varphi_n(t)]^{\mathrm{T}}$ is the regression vector, $v(t)$ is an additive noise, and $\boldsymbol{\theta}(t)=[\theta_1(t),\dots,\theta_n(t)]^{\mathrm{T}}$ denotes the vector of time-varying coefficients, modeled as weighted sums of complex exponentials

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

All quantities in (1) and (2), except angular frequencies $\omega_1(t),\dots,\omega_k(t)$, are complex-valued. Since the complex amplitudes $a_{li}(t)$ incorporate both magnitude and phase information, there is no explicit phase component in (2).

Manuscript received April 11, 2005; revised January 21, 2006. This work was supported by MNiSW under Grant N514 011 31/3091. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Steven L. Grant.

The authors are with the Faculty of Electronics, Telecommunications and Computer Science, Department of Automatic Control, Gdańsk University of Technology, Gdańsk, Poland (e-mail: maciekn@eti.pg.gda.pl, piokacz@proterians.net.pl).

Digital Object Identifier 10.1109/TSP.2006.881172

We will assume that both the amplitudes $a_{li}(t), l=1,\dots,n$ and frequencies $\omega_i(t)$ in (2) are slowly time-varying and that $v(t)=v_R(t)+jv_I(t)$, $\mathrm{E}[v_R^2(t)]=\mathrm{E}[v_I^2(t)]=\sigma_v^2/2$, $\mathrm{E}[v_R(y)v_I(t)]=0$, $\forall t$, is a complex-valued white noise of variance σ_v^2, independent of the sequence of regression vectors $\boldsymbol{\varphi}(t)$.

Denote by $\boldsymbol{\alpha}_i(t) = [a_{1i}(t),\dots,a_{ni}(t)]^{\mathrm{T}}$ 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 $f_i(t)=e^{j\sum_{\tau=1}^{t}\omega_i(\tau)}$. Using the shorthand notation introduced above, (1) and (2) can be rewritten in the form

$$y(t)=\sum_{i=1}^{k}\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\beta}_i(t)+v(t), \quad \boldsymbol{\theta}(t)=\sum_{i=1}^{k}\boldsymbol{\beta}_i(t).$$

One of the challenging potential applications, which under certain conditions admit the formulation presented above, is adaptive equalization of rapidly fading multipath telecommunication channels—see, e.g., [4]–[6].

The name "generalized notch filters" stems from the fact that the problem of identifying quasi-periodically varying systems is an extension, to the system case, of a classical signal processing task of either elimination or enhancement of complex-valued sinusoidal signals $s(t)=\theta(t)$ (called cisoids) buried in noise [7], [8]

$$y(t)=s(t)+v(t)=\sum_{i=1}^{k}a_i(t)e^{j\sum_{\tau=1}^{t}\omega_i(\tau)}+v(t). \qquad (3)$$

Note that (1) and (2) reduce to (3) after setting $n=1$ and $\varphi(t)=1$, $\forall t$. Therefore, when restricted to the special signal case discussed above, the results developed in this paper offer new solutions to the problem of adaptive notch filtering of complex signals.

B. Contribution and Novelty

Tracking capabilities of generalized adaptive notch filters are determined by two user-dependent tuning knobs: the small adaptation gain $\mu>0$, which controls the rate of amplitude adaptation, and another adaptation gain $\eta>0$ (also small), which decides upon the rate of frequency adaptation. It is a well-known fact that adaptation gains of any finite-memory adaptive filter should be chosen so as to trade off between the filter's tracking speed (which increases with growing gains) and its tracking accuracy (which decreases with growing gains) [9]. In a special case, where frequencies of a quasi-periodically varying system drift according to the random walk model, the optimal values of μ and η can be obtained analytically. For the problem of frequency tracking, such analytical study of tracking performance of a GANF algorithm was presented in our earlier paper [10]. We have shown there that the optimal gains are functions of a scalar coefficient ξ—the product of the signal-to-noise ratio and the variance of frequency changes—which can be regarded a measure of system nonstationarity. Since ξ is not a directly measurable quantity, such theoretical results have a limited practical value.

The contribution of this paper is twofold. First, the frequency tracking study, carried out in [10], is extended to an important problem of parameter (system) tracking. Second, based on the obtained theoretical results, a self-optimizing GANF algorithm is proposed, capable of automatic tuning of its adaptation gains.

This paper is organized as follows. Section II is devoted to tracking performance analysis of a basic GANF algorithm, designed for a system with a single frequency mode of parameter variation. After summarizing known results on frequency tracking, we derive new results on parameter (system) tracking. Section III presents a self-optimizing version of the basic algorithm; the obtained solution is next extended to systems with multiple frequency modes. Section IV presents results of simulation experiments and Section V concludes.

II. Tracking Analysis of a GANF Algorithm

As our starting point, we will choose the steady-state version of a simple generalized adaptive notch filter analyzed in [10], which combines the exponentially weighted least squares approach to amplitude tracking with gradient search approach to frequency tracking.

A. Basic Algorithm

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

$$y(t)=\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\beta}(t)+v(t),\quad \boldsymbol{\beta}(t)=e^{j\omega(t)}\boldsymbol{\beta}(t-1). \tag{4}$$

Note that the assumed model of parameter variation can be rewritten in a more explicit form as

$$\boldsymbol{\theta}(t)=\boldsymbol{\beta}(t)=\boldsymbol{\beta}_o e^{j\sum_{\tau=1}^{t}\omega(\tau)} \tag{5}$$

where $\boldsymbol{\beta}_o=\boldsymbol{\beta}(0)$. According to (5), parameter changes of the analyzed system can be attributed exclusively to changes of the instantaneous frequency $\omega(t)$. Frequency increments will be further denoted by $w(t)$

$$w(t)=\omega(t)-\omega(t-1).$$

Assume that the sequence of regression vectors $\{\boldsymbol{\varphi}(t)\}$, independent of $\{v(t)\}$ and $\{w(t)\}$, is wide-sense stationary and persistently exciting, and that the matrix $\boldsymbol{\Phi}=\mathrm{E}[\boldsymbol{\varphi}^*(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)]>0$ is known a priori. Then the steady-state single-frequency version of the GANF algorithm presented in [10] can be written in the form ([10, (31)])

$$\begin{aligned}\varepsilon(t)&=y(t)-e^{j\widehat{\omega}(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}(t-1)\\ \widehat{\boldsymbol{\beta}}(t)&=e^{j\widehat{\omega}(t)}\widehat{\boldsymbol{\beta}}(t-1)+\mu\boldsymbol{\Phi}^{-1}\boldsymbol{\varphi}^*(t)\varepsilon(t)\\ g(t)&=\mathrm{Im}[\varepsilon^*(t)e^{j\widehat{\omega}(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}(t-1)]\\ \widehat{\omega}(t+1)&=\widehat{\omega}(t)-\eta g(t)\\ \widehat{\boldsymbol{\theta}}(t)&=\widehat{\boldsymbol{\beta}}(t).\end{aligned} \tag{6}$$

In the above algorithm $\mu>0$ denotes a small adaptation gain, which controls the rate of amplitude adaptation, and $\eta>0$, also set close to zero, is another gain which decides upon the rate of frequency adaptation. Generally speaking, both design parameters should be chosen so as to trade off the tracking speed of a generalized adaptive notch filter (which increases with growing μ and η) and its noise rejection capability (which decreases with growing μ and η).

Analysis of the tracking properties of the basic algorithm will be an important step towards development of a fully adaptive procedure, i.e., procedure with built-in mechanisms for automatic adjustment of the adaptation gains μ and η.

B. Frequency Tracking Properties of the Basic Algorithm

The frequency tracking capabilities of the algorithm (6) were examined in [10].

Denote by $\Delta\widehat{\omega}(t)=\widehat{\omega}(t)-\omega(t)$ the frequency estimation error and let

$$e(t)=\boldsymbol{\beta}^{\mathrm{H}}(t)\boldsymbol{\varphi}^*(t)v(t)=e_R(t)+je_I(t)$$

, where $e_R(t)=\mathrm{Re}\,[e(t)]$ and $e_I(t)=\mathrm{Im}\,[e(t)]$. It can be easily checked that $e(t)$ is a complex-valued white noise with variance $\sigma_e^2=\mathrm{E}[|e(t)|^2]=b^2\sigma_v^2$, $b^2=\boldsymbol{\beta}_o^{\mathrm{H}}\boldsymbol{\Phi}\boldsymbol{\beta}_o$, and that $\mathrm{E}[e_R^2(t)]=\mathrm{E}[e_I^2(t)]=\sigma_e^2/2$, $\mathrm{E}[e_R(t)e_I(t)]=0$.

Combining the approximating linear filtering (ALF) technique proposed by Tichavský and Händel [7] for analysis of "classical" adaptive notch filters with a stochastic averaging method [11], which is a well-established approach to analysis of adaptive systems, the following approximation was obtained in [10]:

$$\Delta\widehat{\omega}(t)\cong H_1(q^{-1})e_I(t)+H_2(q^{-1})w(t) \tag{7}$$

where

$$\begin{aligned}H_1(q^{-1})&=\frac{(1-\delta)(1-q^{-1})q^{-1}}{b^2(1-(\lambda+\delta)q^{-1}+\lambda q^{-2})}\\ H_2(q^{-1})&=-\frac{1-\lambda q^{-1}}{1-(\lambda+\delta)q^{-1}+\lambda q^{-2}}\end{aligned} \tag{8}$$

and $\lambda=1-\mu$, $\delta=1-\eta b^2$ (q^{-1} denotes the backward shift operator).

As is straightforward to check, the approximating linear filters $H_1(q^{-1})$ and $H_2(q^{-1})$ are asymptotically stable for any λ and δ from the interval (0,1).

To obtain further insights into the tracking behavior of a GANF algorithm, we will assume that the instantaneous frequency $\omega(t)$ evolves according to the random walk model, i.e., that the frequency increments $w(t)$ form a white noise sequence, independent of $v(t)$. In a case like this, it holds that

$$\mathrm{E}[(\Delta\widehat{\omega}(t))^2] \cong I[H_1(z^{-1})]\ \mathrm{E}[e_I^2(t)] + I[H_2(z^{-1})]\ \mathrm{E}[w^2(t)]$$

where

$$I[X(z^{-1})] = \frac{1}{2\pi j}\oint X(z^{-1})X(z)\frac{dz}{z}$$

is an integral evaluated along the unit circle in the z plane and $X(z^{-1})$ denotes any stable proper rational transfer function.

By means of residue calculus [19], one obtains

$$\mathrm{E}[(\widehat{\omega}(t)-\omega(t))^2] \cong \frac{\gamma^2}{4b^2\mu}\,\sigma_v^2 + \left[\frac{\mu}{2\gamma}+\frac{1}{2\mu}\right]\sigma_w^2 \tag{9}$$

where $\gamma = 1-\delta = b^2\eta$ and the approximation holds for sufficiently small values of μ and γ.

Denote by μ_ω and γ_ω the values of μ and γ that minimize the mean-squared frequency estimation error. Straightforward calculations yield

$$\mu_\omega = \sqrt[4]{8\xi}, \quad \gamma_\omega = \sqrt{2\xi}$$
$$\mathrm{E}[(\widehat{\omega}(t)-\omega(t))^2|\mu_\omega,\gamma_\omega] \cong \sqrt[4]{2\xi^{-1}}\ \sigma_w^2 \tag{10}$$

where the scalar coefficient

$$\xi = \frac{b^2\sigma_w^2}{\sigma_v^2} \tag{11}$$

(the product of the signal-to-noise ratio b^2/σ_v^2 and the variance of frequency changes σ_w^2) can be regarded a measure of system nonstationarity.

Note that, according to (7), for random walk frequency variations the GANF algorithm (6) yields unbiased frequency estimates. Additionally, as was shown in [10], under Gaussian assumptions it is a statistically efficient estimation procedure, i.e., the minimum mean-squared frequency estimation error given by (10), achieved when the design parameters are optimally tuned, is equal to its limiting value set by the Cramér–Rao inequality.

C. Parameter (System) Tracking Properties of the Basic Algorithm

Denote by $\widehat{s}(t) = \boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}(t)$ the estimate of the noiseless system output $s(t) = \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\beta}(t)$. The quantity

$$\varepsilon_s(t) = \widehat{s}(t) - s(t) = \boldsymbol{\varphi}^{\mathrm{T}}(t)\Delta\widehat{\boldsymbol{\beta}}(t)$$

where $\Delta\widehat{\boldsymbol{\beta}}(t) = \widehat{\boldsymbol{\beta}}(t) - \boldsymbol{\beta}(t)$ will be further called the system tracking error. Let $f(t) = e^{j\sum_{\tau=1}^{t}\omega(\tau)}$ and

$$\Delta\widetilde{\boldsymbol{\beta}}(t) = f^*(t)\Delta\widehat{\boldsymbol{\beta}}(t).$$

As was argued in [10], when the adaptation gains μ and γ are small, the quantity $\Delta\widetilde{\boldsymbol{\beta}}(t)$ varies slowly compared to $\boldsymbol{\varphi}(t)$. Therefore, using the method of stochastic averaging, one arrives at

$$\begin{aligned}\mathrm{E}[|\varepsilon_s(t)|^2] &= \mathrm{E}[\Delta\widehat{\boldsymbol{\beta}}^{\mathrm{H}}(t)\boldsymbol{\varphi}^*(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)\Delta\widehat{\boldsymbol{\beta}}(t)]\\ &= \mathrm{E}[\Delta\widetilde{\boldsymbol{\beta}}^{\mathrm{H}}(t)\boldsymbol{\varphi}^*(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)\Delta\widetilde{\boldsymbol{\beta}}(t)]\\ &\cong \mathrm{E}[\Delta\widetilde{\boldsymbol{\beta}}^{\mathrm{H}}(t)\boldsymbol{\Phi}\Delta\widetilde{\boldsymbol{\beta}}(t)].\end{aligned} \tag{12}$$

Note that the mean-squared system tracking error, which is approximately equal to the excess mean-squared one step ahead prediction error

$$\mathrm{E}[|\varepsilon_s(t)|^2] \cong \mathrm{E}[|\varepsilon(t)|^2] - \sigma_v^2 \tag{13}$$

reflects the predictive ability of the system model. Hence, from the practical viewpoint, it is a more meaningful statistic than the mean-squared parameter tracking error $\mathrm{E}[||\widehat{\boldsymbol{\beta}}(t)-\boldsymbol{\beta}(t)||^2]$. Both measures coincide when $\boldsymbol{\Phi}$ is similar to an identity matrix.

As shown in the Appendix , the mean-squared system tracking error, yielded by the algorithm (6) applied to the system (4) subject to random walk frequency drift, can be approximated by the following expression:

$$\mathrm{E}[|\varepsilon_s(t)|^2] \cong \left[\frac{\gamma}{4\mu}+\frac{n\mu}{2}\right]\sigma_v^2 + \frac{b^2}{2\mu\gamma}\,\sigma_w^2 \tag{14}$$

where $n = \dim\boldsymbol{\theta}(t)$ denotes the number of estimated system coefficients.

In the special signal case ($n=1$), the result derived above becomes identical with the expression for the mean-squared signal tracking error given by [10, (18)]. We note, however, that derivation of (14) is not a trivial extension of the analogous derivation presented in [10].

Denote by μ_s and γ_s the values of μ and γ that minimize (14). It is easy to check that

$$\mu_s = \sqrt[4]{\frac{2\xi}{n^2}}, \quad \gamma_s = \sqrt{2\xi}$$
$$\mathrm{E}[|\varepsilon_s(t)|^2|\mu_s,\gamma_s] \cong \sqrt[4]{2\xi n^2}\ \sigma_v^2. \tag{15}$$

Note that according to (10) and (15), the settings that are optimal from the system tracking (prediction) viewpoint are not the best choice from the frequency tracking viewpoint and vice versa. It should be stressed that—in the majority of system-oriented applications—minimization of system tracking errors, rather than minimization of frequency tracking errors, is our main objective. For example, when tracking the time-varying impulse response coefficients of a rapidly fading communication channel, one is usually not interested in its frequency components analysis—all that matters is how well the predicted channel output approximates the true channel response.

III. Optimization of a GANF Algorithm

The analytical results presented in the previous section have some obvious limitations. First, the random walk model of frequency variation can be criticized as rather naive. Second, in order to optimize filter settings, one should know the rate of nonstationarity of the identified system, which is not a directly measurable quantity.

The first objection does not seem to be critical. The point is that, due to its finite-memory property, the GANF algorithm does not rely heavily on information coming from the remote past, and hence its local performance analysis does not require an adequate "global" model of frequency variation. As long as system frequencies change slowly over time, the random walk model with appropriately chosen rate of change σ_w^2 can be considered a good local (in time) description of real frequency variations, similarly as a linear regression model can be often regarded a satisfactory local (in space) description of a nonlinear plant around a given operating point.

The second limitation is more important from the practical viewpoint. When the rate of system nonstationarity ξ is not known a priori, or when it changes with time, an autotuning mechanism of some kind should be provided to perform online optimization of the algorithm's tracking performance. One can attempt to solve this problem in two different ways.

Using the indirect optimization approach, one may set

$$\mu(t) = f(\widehat{\xi}(t)), \quad \gamma(t) = g(\widehat{\xi}(t))$$

where the functions $f(\cdot)$ and $g(\cdot)$ are chosen in accordance with (10) or (15) and $\widehat{\xi}(t)$ denotes a local estimate of the rate of system nonstationarity. In the signal processing case, the indirect approach was suggested (but not elaborated) by Tichavský and Händel [7].

The second solution, advocated in this paper, is based on direct optimization. The term "direct optimization" is used to emphasize the fact that in this approach all adjustments are made by means of direct minimization of the *observed* prediction errors, regarded as functions of μ and γ—in contrast to the indirect approach, where all decisions are based on *anticipated* effects which, according to theory, μ and γ should have on prediction errors. From the practical viewpoint the direct method, which incorporates decision feedback, is more advisable than the indirect method, which rests on feedforward compensation only.

Direct optimization can be performed using either sequential or parallel estimation techniques. The first case uses a single tracking algorithm equipped with adjustable adaptation gains. The second case takes several algorithms with different gain settings, runs them in parallel, and compares them according to their predictive abilities—see [12] and [9].

In this paper, we will exploit the sequential approach, which is computationally less demanding and which provided very good tracking results in all preliminary tests. We will show that the adaptation constants μ and γ can be optimized using the method of recursive prediction error (RPE).

A. Tuning Rule

Consider the problem of minimization of a system tracking error. Before we derive the basic autotuning rule, we will preoptimize the GANF algorithm. Observe that irrespective of which variant of the optimization strategy is chosen, the optimal value of γ is proportional to the square of the corresponding value of μ. In particular, when minimization of system tracking errors is our main objective, it holds that $\gamma_s = n\mu_s^2$. This suggests that setting $\gamma = b^2\eta = n\mu^2$ may be a good way of reducing the number of design degrees of freedom of a GANF algorithm from two (μ, γ) to one (μ). Assume, for the time being, that the covariance matrix $\boldsymbol{\Phi}$ and the scalar coefficient $b^2 = \boldsymbol{\beta}_o^{\mathrm{H}}\boldsymbol{\Phi}\boldsymbol{\beta}_o$ are constant and known (we will relax both assumptions later on). Then the preoptimized version of (6) can be written in the form

$$\begin{aligned}
\varepsilon(t) &= y(t) - e^{j\widehat{\omega}(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}(t-1)\\
\widehat{\boldsymbol{\beta}}(t) &= e^{j\widehat{\omega}(t)}\widehat{\boldsymbol{\beta}}(t-1) + \mu\boldsymbol{\Phi}^{-1}\boldsymbol{\varphi}^*(t)\varepsilon(t)\\
g(t) &= \mathrm{Im}[\varepsilon^*(t)e^{j\widehat{\omega}(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}(t-1)]\\
\widehat{\omega}(t+1) &= \widehat{\omega}(t) - \kappa\mu^2 g(t)\\
\widehat{\boldsymbol{\theta}}(t) &= \widehat{\boldsymbol{\beta}}(t)
\end{aligned} \tag{16}$$

where $\kappa = n/b^2$.

We will adjust the adaptation gain μ recursively, by minimizing the following local measure of fit, made up of exponentially weighted prediction errors

$$V(t,\mu) = \frac{1}{2}\sum_{\tau=1}^{t}\rho^{t-\tau}|\varepsilon(\tau,\mu)|^2.$$

The forgetting constant ρ $(0 < \rho < 1)$ decides upon the effective averaging range. Note that, according to (13), minimization of the mean-squared system tracking error is equivalent to minimization of the mean-squared prediction error, and that $V(t,\mu)$ is the local estimate of the latter quantity (up to the scaling factor). Therefore minimization of $V(t,\mu)$ is fully consistent with the system tracking oriented optimization strategy.

To evaluate the estimate $\widehat{\mu}(t) = \arg\min_\mu V(t,\mu)$, we will use the standard RPE approach. According to Söderström and Stoica [13], the RPE algorithm can be expressed in the form

$$\begin{aligned}
\widehat{\mu}(t) &= \widehat{\mu}(t-1) - [V''(t,\widehat{\mu}(t-1))]^{-1}\,V'(t,\widehat{\mu}(t-1))\\
V'(t,\widehat{\mu}(t-1)) &\cong \mathrm{Re}\left[\varepsilon(t,\widehat{\mu}(t-1))\frac{\partial\varepsilon^*(t,\widehat{\mu}(t-1))}{\partial\mu}\right]\\
V''(t,\widehat{\mu}(t-1)) &\cong \rho V''(t-1,\widehat{\mu}(t-2)) + \left|\frac{\partial\varepsilon(t,\widehat{\mu}(t-1))}{\partial\mu}\right|^2
\end{aligned}$$

where all derivatives are taken with respect to μ.

Denote

$$\begin{aligned}
\zeta(t) &= \frac{\partial\varepsilon(t,\widehat{\mu}(t-1))}{\partial\mu}, \quad \boldsymbol{\psi}(t) = \frac{\partial\widehat{\boldsymbol{\beta}}(t,\widehat{\mu}(t-1))}{\partial\mu}\\
\varrho(t) &= \frac{\partial g(t,\widehat{\mu}(t-1))}{\partial\mu}, \quad \chi(t) = \frac{\partial\widehat{\omega}(t,\widehat{\mu}(t-1))}{\partial\mu}\\
r(t) &= V''(t,\widehat{\mu}(t-1)).
\end{aligned}$$

Straightforward calculations lead to

$$\begin{aligned}
\boldsymbol{\vartheta}(t) &= e^{j\widehat{\omega}(t)}[j\chi(t)\widehat{\boldsymbol{\beta}}(t-1) + \boldsymbol{\psi}(t-1)]\\
\zeta(t) &= -\,\boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\vartheta}(t)\\
\boldsymbol{\psi}(t) &= \boldsymbol{\vartheta}(t) + \boldsymbol{\Phi}^{-1}\boldsymbol{\varphi}^*(t)[\varepsilon(t) + \widehat{\mu}(t-1)\zeta(t)]\\
\varrho(t) &= \mathrm{Im}[\zeta^*(t)e^{j\widehat{\omega}(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}(t-1) - \varepsilon^*(t)\zeta(t)]\\
r(t) &= \rho\, r(t-1) + |\zeta(t)|^2\\
\widehat{\mu}(t) &= \widehat{\mu}(t-1) - \frac{\mathrm{Re}[\varepsilon(t)\zeta^*(t)]}{r(t)}\\
\chi(t+1) &= \chi(t) - \kappa\widehat{\mu}(t)[2g(t) + \widehat{\mu}(t)\varrho(t)].
\end{aligned}$$

A simple gradient-based version of the optimization strategy described above was used for automatic tuning of classical system identification algorithms: the least mean squares (LMS) algorithm [14] and the exponentially weighted least squares (EWLS) algorithm [15, p. 160], [16]. In the signal processing case, the RPE approach was used, for tuning of "ordinary" adaptive notch filters, by Dragošević and Stanković [17], [18].

B. Refinements

When the covariance matrix $\boldsymbol{\Phi}$ is not known and/or it is time-varying, one can replace it with the following exponentially weighted 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 the forgetting constant. We note that the inverse of $\widehat{\boldsymbol{\Phi}}(t)$ can be also evaluated recursively by exploiting the well-known matrix inversion lemma [13].

The coefficient b^2, which was also assumed to be known and constant, can be replaced with

$$\widehat{b}^2(t) = \widehat{\boldsymbol{\beta}}^{\mathrm{H}}(t)\widehat{\boldsymbol{\Phi}}(t)\widehat{\boldsymbol{\beta}}(t).$$

Our last modification is a safety valve: to prevent the algorithm from erratic behavior in extreme situations (e.g., in the presence of sudden amplitude and/or frequency jumps), it is advisable to set the maximum allowable value of μ. Each time the calculated value of $\widehat{\mu}(t)$ exceeds its upper limit, it should be truncated, for safety reasons, to $\mu_{\max}$. Similarly, $\widehat{\mu}(t)$ should be set to zero whenever the calculated value gets below zero.

C. Extension to the Multiple Frequencies Case

So far we have been assuming that the identified quasi-periodically varying system has a single frequency mode. Now we will show how the obtained results can be extended to the multiple frequencies case. We will use the concept of frequency decomposition of the identified system [2]. Denote by

$$y_i(t) = \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\beta}_i(t) + v(t)$$

the output of the ith subsystem of (1), i.e., subsystem associated with the frequency ω_i. If the signals $y_1(t), \ldots, y_k(t)$ were available, one could design k independent GANF algorithms of the form

$$\begin{aligned}
\varepsilon_i(t) &= y_i(t) - e^{j\widehat{\omega}_i(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1)\\
\widehat{\boldsymbol{\beta}}_i(t) &= e^{j\widehat{\omega}_i(t)}\widehat{\boldsymbol{\beta}}_i(t-1) + \widehat{\mu}_i(t)\boldsymbol{\Phi}^{-1}\boldsymbol{\varphi}^*(t)\varepsilon_i(t)\\
&\text{etc.}
\end{aligned}$$

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 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

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

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

$$\varepsilon_1(t) = \cdots = \varepsilon_k(t) = y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\sum_{i=1}^{k}e^{j\widehat{\omega}_i(t)}\widehat{\boldsymbol{\beta}}_i(t-1) = \varepsilon(t)$$

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 $\boldsymbol{\theta}(t)$.

D. Self-Optimizing Algorithm

After combining all earlier results, the proposed self-optimizing multiple-frequency generalized adaptive notch filtering algorithm can be summarized as follows:

$$\begin{aligned}
\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{\Phi}}(t) &= \lambda_o\widehat{\boldsymbol{\Phi}}(t-1) + (1-\lambda_o)\boldsymbol{\varphi}^*(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)\\
\boldsymbol{\vartheta}_i(t) &= e^{j\widehat{\omega}_i(t)}[j\chi_i(t)\widehat{\boldsymbol{\beta}}_i(t-1) + \boldsymbol{\psi}_i(t-1)]\\
\zeta_i(t) &= -\ \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\vartheta}_i(t)\\
\boldsymbol{\psi}_i(t) &= \boldsymbol{\vartheta}_i(t) + \widehat{\boldsymbol{\Phi}}^{-1}(t)\boldsymbol{\varphi}^*(t)[\varepsilon(t) + \widehat{\mu}_i(t-1)\zeta_i(t)]\\
\varrho_i(t) &= \mathrm{Im}[\zeta_i^*(t)e^{j\widehat{\omega}_i(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1) - \varepsilon^*(t)\zeta_i(t)]\\
r_i(t) &= \rho_i r_i(t-1) + |\zeta_i(t)|^2\\
\widetilde{\mu}_i(t) &= \widehat{\mu}_i(t-1) - \frac{\mathrm{Re}[\varepsilon(t)\zeta_i^*(t)]}{r_i(t)}\\
\widehat{\mu}_i(t) &= \begin{cases} 0, & \text{if } \widetilde{\mu}_i(t) < 0\\ \widetilde{\mu}_i(t) & \text{if } 0 \le \widetilde{\mu}_i(t) \le \mu_{max}\\ \mu_{max}, & \text{if } \widetilde{\mu}_i(t) > \mu_{max}\end{cases}\\
\widehat{\boldsymbol{\beta}}_i(t) &= e^{j\widehat{\omega}_i(t)}\widehat{\boldsymbol{\beta}}_i(t-1) + \widehat{\mu}_i(t)\widehat{\boldsymbol{\Phi}}^{-1}(t)\boldsymbol{\varphi}^*(t)\varepsilon(t)\\
g_i(t) &= \mathrm{Im}[\varepsilon^*(t)e^{j\widehat{\omega}_i(t)}\boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1)]\\
\widehat{b}_i^2(t) &= \widehat{\boldsymbol{\beta}}_i^{\mathrm{H}}(t)\widehat{\boldsymbol{\Phi}}(t)\widehat{\boldsymbol{\beta}}_i(t)\\
\kappa_i(t) &= \frac{n}{\widehat{b}_i^2(t)}\\
\widehat{\omega}_i(t+1) &= \widehat{\omega}_i(t) - \kappa_i(t)\widehat{\mu}_i^2(t)g_i(t)\\
\chi_i(t+1) &= \chi_i(t) - \kappa_i(t)\widehat{\mu}_i(t)[2g_i(t) + \widehat{\mu}_i(t)\varrho_i(t)]\\
i &= 1, \ldots, k\\
\widehat{\boldsymbol{\theta}}(t) &= \sum_{i=1}^{k}\widehat{\boldsymbol{\beta}}_i(t).
\end{aligned} \tag{17}$$

Interestingly, as a "byproduct" of the system-oriented analysis, carried out in this paper, one obtains a new signal processing algorithm: after setting $n = 1$, $\varphi(t) = 1$, and $\widehat{\Phi}(t) = 1$, $\forall t$, the algorithm (17) reduces down to a novel self-optimizing adaptive notch filter for complex-valued signals.

The recommended way of selecting initial conditions for the algorithm described above is by means of preprocessing. We have shown in [3] that all initial conditions needed to start (or restart) the GANF algorithm can be inferred from the nonparametric DFT-based system identification results, obtained for a short startup fragment of the input/output data of length $N \cong 2.8/\mu$. This includes the number of frequency modes k, the initial values of frequency estimates $\widehat{\omega}_i(N/2+1)$, the initial values of parameter estimates $\widehat{\boldsymbol{\beta}}_i(N/2)$, and the initial estimate of the covariance matrix $\widehat{\boldsymbol{\Phi}}(N/2)$ (the algorithm is deliberately started in the middle of the analysis interval).

The initial values of $\boldsymbol{\psi}_i(N/2)$, $\chi_i(N/2+1)$, and $r_i(N/2)$ can be set to zero. However, to avoid rough start of the autotuning part of the algorithm, it is recommended that the adaptation gains $\widehat{\mu}_i(t)$ are kept constant at their startup values $\widehat{\mu}_i(N/2)$ until the quantities $r_i(t)$ reach their steady-state levels.

It is known that in order to guarantee that a hierarchical, multilayer adaptive system works reliably, the adaptation time constants of consecutive layers should be much larger than the analogous constants of the preceding layers—see, e.g., remarks on the frequency-domain design guidelines for adaptive systems, presented in [20, p. 302]. This useful rule of thumb leads to the following constraint:

$$1 - \rho_i \ll \mu_i \tag{18}$$

which can be incorporated in practice by setting $\rho_i(t) = 1 - 0.1\widehat{\mu}_i(t-1)$. Computer simulations confirm that the forgetting factor λ_o plays a relatively insignificant role; it can be set, for example, to $\lambda_o = 0.95$. Finally, the upper truncation level $\mu_{\max}$ can be set equal to 0.2.

IV. Simulation Results

To check validity of (14), the following two-tap finite impulse response (FIR) system (inspired by channel equalization applications) was simulated:

$$y(t) = \theta_1(t)u(t) + \theta_2(t)u(t-1) + v(t) \tag{19}$$

where $u(t)$ denotes a white 4-QAM input sequence ($u(t) = \pm 1 \pm j$, $\sigma_u^2 = 2$) and $v(t)$ denotes a complex-valued Gaussian measurement noise. The impulse response coefficients of this system were modeled as nonstationary cisoids $\theta_i(t) = a_i e^{j\sum_{\tau=1}^{t}\omega(\tau)}$, $i = 1, 2$, with time-invariant complex "amplitudes" $\boldsymbol{\alpha} = [a_1, a_2]^{\mathrm{T}} = [2 - j\ ,\ 1 + 2j\ ]^{\mathrm{T}}$. Note that in this case $\boldsymbol{\beta}_o = \boldsymbol{\alpha}$, $\boldsymbol{\varphi}(t) = [u(t), u(t-1)]^{\mathrm{T}}$, and $\boldsymbol{\Phi} = \mathbf{I}_2\sigma_u^2$.

The frequency $\omega(t)$ evolved according to the random walk model with the starting value set equal to $\omega(0) = \pi/2$ and with the variance of frequency increments set equal to $\sigma_w^2 = 10^{-7}$. Four noise levels were considered ($\sigma_v^2 = 20, 2\sqrt{10}, 2$, and 0.2) to check tracking performance of the GANF algorithm under different SNR conditions (0, 5, 10, and 20 dB, respectively).

Fig. 1 shows comparison of theoretical evaluations, based on (14), with the results of computer simulations. Each plot illustrates dependence of the mean-squared system tracking error on μ in the case where $\gamma = n\mu^2 = 2\mu^2$. All experimental points were obtained by means of double averaging: over time (10 000 iterations) and over different realizations of $\{w(t)\}$ (50 realizations). Note good agreement between the theoretical curves and the results of computer simulations.

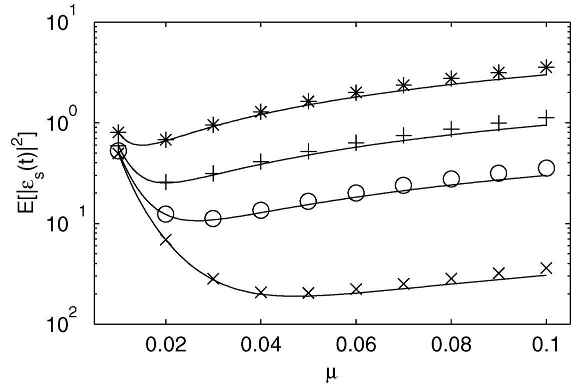

Fig. 1. Variance of the system tracking error $\varepsilon_s(t)$ for a two-tap FIR system with a single frequency mode subject to a random walk drift. The theoretical results (solid lines) are compared with simulation results obtained for different values of μ, given $\gamma = 2\mu^2$; the corresponding signal-to-noise ratios were 0 dB ($*$), 5 dB ($+$), 10 dB ($\circ$), and 20 dB ($\times$).

The second experiment was set to check effectiveness of the proposed gain-scheduling rule. The simulated FIR system, also governed by (19), had two quasi-periodic modes of parameter variation ($k = 2$)

$$\theta_i(t) = a_{i1}e^{j\sum_{\tau=1}^{t}\omega_1(\tau)} + a_{i2}e^{j\sum_{\tau=1}^{t}\omega_2(\tau)}, \quad i = 1, 2$$

with $\boldsymbol{\alpha}_1 = [a_{11}, a_{12}]^{\mathrm{T}} = [2 - j\ , 1 + 2j]^{\mathrm{T}}$ and $\boldsymbol{\alpha}_2 = [a_{21}, a_{22}]^{\mathrm{T}} = [1 - 2j\ , 2 + j]^{\mathrm{T}}$. The instantaneous frequencies $\omega_1(t)$ and $\omega_2(t)$ were modeled as two independent random walk processes with time-varying rates of change

$$\sigma_{w_1}^2(t) = \begin{cases} 10^{-7} & \text{for } t = 1, \ldots, 6000 \\ 2.5 \cdot 10^{-6} & \text{for } t = 6001, \ldots, 10\,000 \end{cases}$$
$$\sigma_{w_2}^2(t) = \begin{cases} 10^{-7} & \text{for } t = 1, \ldots, 4000 \\ 1.6 \cdot 10^{-6} & \text{for } t = 4001, \ldots, 10\,000 \end{cases}$$

and initial conditions set to $\omega_1(0) = \pi/8$, $\omega_2(0) = \pi/3$. The variance of the measurement noise $v(t)$ was also time-dependent

$$\sigma_v^2(t) = \begin{cases} 4 & \text{for } t = 1, \ldots, 8000 \\ 16 & \text{for } t = 8001, \ldots, 10\,000 \end{cases}.$$

The evaluation of the gain scheduling rule was started at the instant $t = 2001$, after the GANF algorithm (17) has reached its steady state. The entire analysis interval $T = [2001, 10\,000]$, covering 8000 samples, was divided into four subintervals $T_1 = [2001, 4000]$, $T_2 = [4001, 6000]$, $T_3 = [6001, 8000]$, and $T_4 = [8001, 10\,000]$, corresponding to four different variance patterns. The optimal values of adaptation gains, evaluated according to (15), were $\mu_1(t) = \mu_2(t) = 0.022$ for $t \in T_1$; $\mu_1(t) = 0.022$, $\mu_2(t) = 0.045$ for $t \in T_2$; $\mu_1(t) = 0.050$, $\mu_2(t) = 0.045$ for $t \in T_3$; and $\mu_1(t) = 0.035$, $\mu_2(t) = 0.031$ for $t \in T_4$.

The self-optimizing GANF algorithm (17) was implemented with $\rho_1 = \rho_2 = 0.995$, which was consistent, for the analyzed system, with (18) and which yielded satisfactory tracking results; adoption of smaller values of ρ, e.g., $\rho_1 = \rho_2 = 0.99$, led to performance deterioration.

The results, gathered in Table I, show that the proposed gain scheduling rule does a pretty good job in optimizing

TABLE I
AVERAGE VALUES OF THE EXCESS OUTPUT PREDICTION ERRORS OBSERVED FOR THE OPTIMAL CHOICE OF ADAPTATION GAINS, FOR ADAPTIVE CHOICE OF ADAPTATION GAINS AND FOR FIVE FIXED VALUES OF $\mu_1 = \mu_2$; ALL AVERAGES WERE COMPUTED FROM THE RESULTS OF 50 SIMULATION EXPERIMENTS FOR THE ENTIRE ANALYSIS INTERVAL (T) AND FOR EACH OF ITS FOUR SUBINTERVALS (T_1, T_2, T_3, T_4)

μ_1, μ_2	T	T_1	T_2	T_3	T_4
optimal	1.05	0.38	0.59	0.87	2.36
adaptive	1.14	0.42	0.65	0.93	2.57
0.01	7.83	1.12	5.26	12.10	12.90
0.02	1.94	0.38	1.26	2.63	3.50
0.03	1.17	0.43	0.72	1.16	2.38
0.04	1.17	0.56	0.69	0.88	2.55
0.05	1.36	0.72	0.79	0.89	3.04

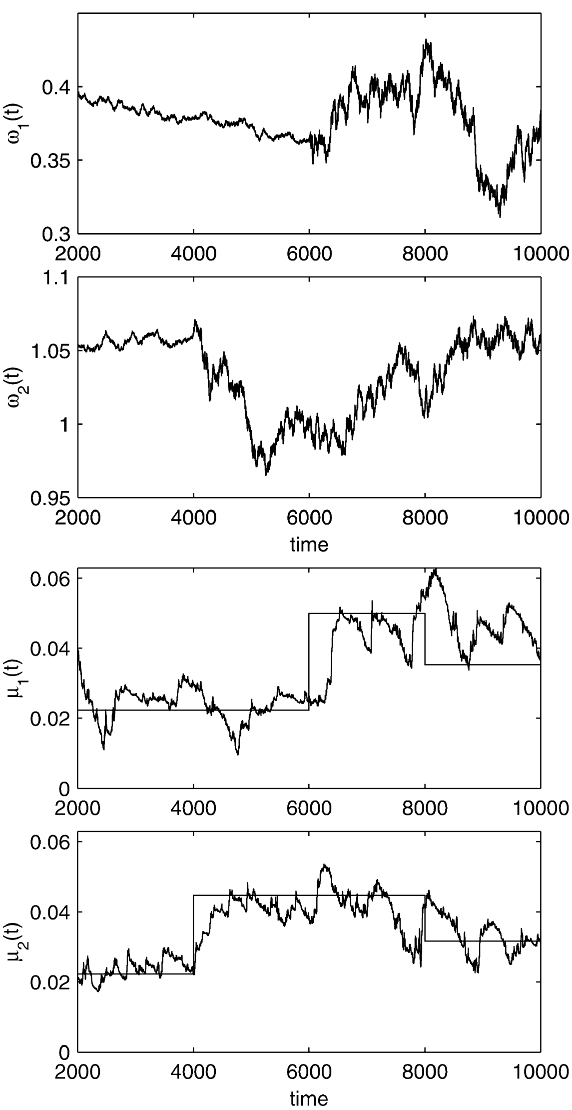

Fig. 2. Evolution of the instantaneous frequencies $\omega_1(t)$ and $\omega_2(t)$ (two upper plots) and the corresponding gain estimates $\widehat{\mu}_1(t)$ and $\widehat{\mu}_2(t)$ (two lower plots) observed in a typical simulation experiment; thin lines show the optimal values of μ_1 and μ_2.

the tracking performance of a GANF algorithm. Note that the self-optimizing algorithm works better than any of the

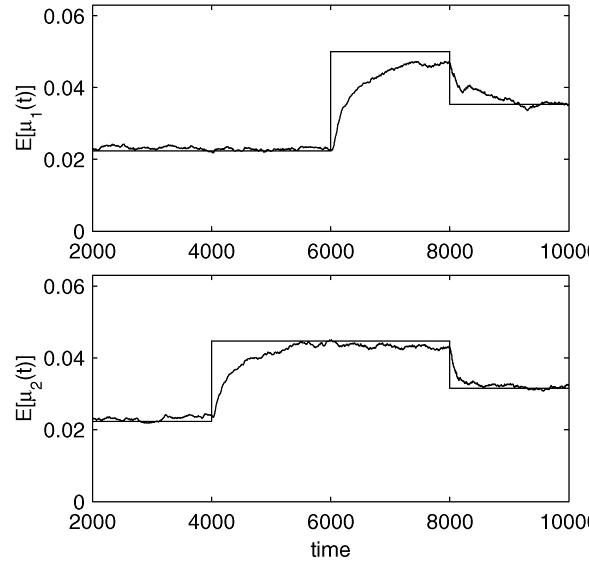

Fig. 3. Ensemble averages of the gain estimates $\widehat{\mu}_1(t)$ and $\widehat{\mu}_2(t)$ obtained from 50 simulations; thin lines show the optimal values of μ_1 and μ_2.

fixed-gain algorithms, and that it is only slightly (about 10%) worse than the optimally tuned algorithm, which incorporates knowledge of the true rates of system nonstationarity, not available in practice. Even though the single-realization estimation results, shown in Fig. 2, may look excessively noisy (this is clearly a consequence of a low sensitivity of the prediction error $\varepsilon(t)$ to μ_1 and μ_2 in the explored range of μ), the ensemble averages, depicted in Fig. 3, fully confirm that the RPE-based gain scheduling rule works correctly.

V. CONCLUSION

We have shown that a generalized adaptive notch filter, used for identification/tracking of quasi-periodically varying systems, can be equipped with an autotuning loop, capable of performing online optimization of the algorithm's tracking performance. The proposed adaptation mechanism combines analytical results, derived for GANF algorithms, with the recursive prediction error approach to optimization of adaptive filters. Results of computer simulations fully confirm effectiveness of the proposed scheme.

APPENDIX
DERIVATION OF (14)

It can be shown that (see [10, Appendix III])

$$\Delta\widetilde{\boldsymbol{\beta}}(t) \cong \lambda\Delta\widetilde{\boldsymbol{\beta}}(t-1) + j\lambda\boldsymbol{\beta}_o\Delta\widehat{\omega}(t) + \mu\boldsymbol{\Phi}^{-1}\mathbf{n}(t) \qquad (20)$$

where $\mathbf{n}(t) = f^*(t)\boldsymbol{\varphi}^*(t)v(t)$. Since $v(t) = v_R(t) + jv_I(t)$, one obtains $\mathbf{n}(t) = \mathbf{n}_A(t) + j\mathbf{n}_B(t)$, where $\mathbf{n}_A(t) = f^*(t)\boldsymbol{\varphi}^*(t)v_R(t)$ and $\mathbf{n}_B(t) = f^*(t)\boldsymbol{\varphi}^*(t)v_I(t)$. Note that $\mathbf{n}_A(t)$ and $\mathbf{n}_B(t)$ are complex-valued (vector) random variables, i.e., $\mathbf{n}_A(t)$ is *not* a real part of $\mathbf{n}(t)$ and $\mathbf{n}_B(t)$ is *not* its imaginary part. Note also that both variables are uncorrelated (due to the uncorrelatedness of $v_R(t)$ and $v_I(t)$)

and $\mathrm{E}[\mathbf{n}_A(t)\mathbf{n}_A^{\mathrm{H}}(t)] = \mathrm{E}[\mathbf{n}_B(t)\mathbf{n}_B^{\mathrm{H}}(t)] = (\sigma_v^2/2)\mathbf{\Phi}$. Since $e(t) = \boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{n}(t)$, it holds that

$$e_I(t) = \mathrm{Im}[e(t)] = \frac{\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{n}(t) - \boldsymbol{\beta}_o^{\mathrm{T}}\mathbf{n}^*(t)}{2j} = z_B(t) - jz_A(t) \tag{21}$$

where

$$z_A(t) = \frac{\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{n}_A(t) - \boldsymbol{\beta}_o^{\mathrm{T}}\mathbf{n}_A^*(t)}{2}$$
$$z_B(t) = \frac{\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{n}_B(t) + \boldsymbol{\beta}_o^{\mathrm{T}}\mathbf{n}_B^*(t)}{2}.$$

After combining (20) with (7) and (21), one arrives at

$$\begin{aligned}
\Delta\widetilde{\boldsymbol{\beta}}(t) &\cong \mathbf{q}_A(t) + \mathbf{q}_B(t) + \mathbf{q}_C(t)\\
\mathbf{q}_A(t) &= \frac{\lambda\boldsymbol{\beta}_o H_1(q^{-1})}{1-\lambda q^{-1}}\, z_A(t) + \frac{\mu\mathbf{\Phi}^{-1}}{1-\lambda q^{-1}}\,\mathbf{n}_A(t)\\
\mathbf{q}_B(t) &= j\left[\frac{\lambda\boldsymbol{\beta}_o H_1(q^{-1})}{1-\lambda q^{-1}}\, z_B(t) + \frac{\mu\mathbf{\Phi}^{-1}}{1-\lambda q^{-1}}\,\mathbf{n}_B(t)\right]\\
\mathbf{q}_C(t) &= j\,\frac{\lambda\boldsymbol{\beta}_o H_2(q^{-1})}{1-\lambda q^{-1}}\, w(t).
\end{aligned}$$

Note that

$$\mathrm{E}[\Delta\widetilde{\boldsymbol{\beta}}^{\mathrm{H}}(t)\mathbf{\Phi}\Delta\widetilde{\boldsymbol{\beta}}(t)] = \mathrm{tr}\{\mathbf{\Phi\Sigma}\} \tag{22}$$

where $\mathbf{\Sigma} = \mathrm{E}[\Delta\widetilde{\boldsymbol{\beta}}(t)\Delta\widetilde{\boldsymbol{\beta}}^{\mathrm{H}}(t)]$. Furthermore

$$\mathbf{\Sigma} = \frac{1}{2\pi}\int_{-\pi}^{\pi}\mathbf{S}_{\Delta\widetilde{\boldsymbol{\beta}}}(\omega)\,d\omega \tag{23}$$

where $\mathbf{S}_{\Delta\widetilde{\boldsymbol{\beta}}}(\omega)$ denotes the spectral density matrix of the sequence $\{\Delta\widetilde{\boldsymbol{\beta}}(t)\}$

$$\mathbf{S}_{\Delta\widetilde{\boldsymbol{\beta}}}(\omega) = \sum_{\tau=-\infty}^{\infty}\mathbf{R}_{\Delta\widetilde{\boldsymbol{\beta}}}(\tau)e^{j\omega\tau}$$
$$\mathbf{R}_{\Delta\widetilde{\boldsymbol{\beta}}}(\tau) = \mathrm{E}[\Delta\widetilde{\boldsymbol{\beta}}(t)\Delta\widetilde{\boldsymbol{\beta}}^{\mathrm{H}}(t-\tau)].$$

To evaluate $\mathbf{S}_{\Delta\widetilde{\boldsymbol{\beta}}}(\omega)$ we will refer to the standard linear filtering results. Suppose that

$$\mathbf{y}(t) = \sum_{i=1}^{m}\mathbf{K}_i(q^{-1})\mathbf{x}_i(t)$$

where $\mathbf{x}_1(t),\ldots,\mathbf{x}_m(t)$ are wide-sense stationary random processes and $\mathbf{K}_1(q^{-1}),\ldots,\mathbf{K}_m(q^{-1})$ are stable transfer matrices. Then it holds that

$$\mathbf{S}_{\mathbf{y}}(\omega) = \sum_{i=1}^{m}\sum_{l=1}^{m}\mathbf{K}_i(e^{-j\omega})\mathbf{S}_{\mathbf{x}_i\mathbf{x}_l}(\omega)\mathbf{K}_l^{\mathrm{H}}(e^{-j\omega})$$

where $\mathbf{S}_{\mathbf{x}_i\mathbf{x}_l}(\omega), i,l = 1,\ldots,m$ denote the cross-spectral density (matrix) functions

$$\mathbf{S}_{\mathbf{x}_i\mathbf{x}_l}(\omega) = \sum_{\tau=-\infty}^{\infty}\mathbf{R}_{\mathbf{x}_i\mathbf{x}_l}(\tau)e^{j\omega\tau}$$
$$\mathbf{R}_{\mathbf{x}_i\mathbf{x}_l}(\tau) = \mathrm{E}[\mathbf{x}_i(t)\mathbf{x}_l^{\mathrm{H}}(t-\tau)].$$

Since the processes $\{z_A(t),\mathbf{n}_A(t)\}$, $\{z_B(t),\mathbf{n}_B(t)\}$ and $\{w(t)\}$ are mutually uncorrelated, the cross-spectral density functions for the pairs $(\mathbf{q}_A(t),\mathbf{q}_B(t))$, $(\mathbf{q}_A(t),\mathbf{q}_C(t))$, and $(\mathbf{q}_B(t),\mathbf{q}_C(t))$ are zero, leading to

$$\mathbf{S}_{\Delta\widetilde{\boldsymbol{\beta}}}(\omega) = \mathbf{S}_{\mathbf{q}_A}(\omega) + \mathbf{S}_{\mathbf{q}_B}(\omega) + \mathbf{S}_{\mathbf{q}_C}(\omega).$$

It is straightforward to check that

$$\begin{aligned}
\mathbf{S}_{z_A}(\omega) &= \mathrm{E}[|z_A(t)|^2] = \frac{\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{\Phi}\boldsymbol{\beta}_o}{4}\,\sigma_v^2 - \frac{\mathrm{Re}\{\mathrm{E}[(\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{n}_A(t))^2]\}}{2}\\
\mathbf{S}_{z_B}(\omega) &= \mathrm{E}[|z_B(t)|^2] = \frac{\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{\Phi}\boldsymbol{\beta}_o}{4}\,\sigma_v^2 + \frac{\mathrm{Re}\{\mathrm{E}[(\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{n}_B(t))^2]\}}{2}\\
\mathbf{S}_{z_A\mathbf{n}_A}(\omega) &= \mathrm{E}[z_A(t)\mathbf{n}_A^{\mathrm{H}}(t)]\\
&= \frac{\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{\Phi}}{2}\,\sigma_v^2 - \frac{\boldsymbol{\beta}_o^{\mathrm{T}}\mathrm{E}[\mathbf{n}_A^*(t)\mathbf{n}_A^{\mathrm{H}}(t)]}{2} = \mathbf{S}_{\mathbf{n}_A z_A}^{\mathrm{H}}(\omega)\\
\mathbf{S}_{z_B\mathbf{n}_B}(\omega) &= \mathrm{E}[z_B(t)\mathbf{n}_B^{\mathrm{H}}(t)]\\
&= \frac{\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{\Phi}}{2}\,\sigma_v^2 + \frac{\boldsymbol{\beta}_o^{\mathrm{T}}\mathrm{E}[\mathbf{n}_B^*(t)\mathbf{n}_B^{\mathrm{H}}(t)]}{2} = \mathbf{S}_{\mathbf{n}_B z_B}^{\mathrm{H}}(\omega).
\end{aligned}$$

Combining all relationships derived above and taking into account the fact that $\mathrm{E}[(\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{n}_A(t))^2] = \mathrm{E}[(\boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{n}_B(t))^2]$ and $\mathrm{E}[\mathbf{n}_A^*(t)\mathbf{n}_A^{\mathrm{H}}(t)] = \mathrm{E}[\mathbf{n}_B^*(t)\mathbf{n}_B^{\mathrm{H}}(t)]$, one arrives at

$$\begin{aligned}
&\mathbf{S}_{\mathbf{q}_A}(\omega) + \mathbf{S}_{\mathbf{q}_B}(\omega)\\
&\quad= \frac{\sigma_v^2}{2|1-\lambda e^{-j\omega}|^2}\Big\{\lambda^2 b^2\boldsymbol{\beta}_o\boldsymbol{\beta}_o^{\mathrm{H}}|H_1(e^{-j\omega})|^2\\
&\qquad+\lambda\mu\boldsymbol{\beta}_o\boldsymbol{\beta}_o^{\mathrm{H}}H_1(e^{-j\omega}) + \lambda\mu\boldsymbol{\beta}_o\boldsymbol{\beta}_o^{\mathrm{H}}H_1(e^{j\omega}) + \mathbf{\Phi}^{-1}\mu^2\Big\}\\
&\mathbf{S}_{\mathbf{q}_C}(\omega) = \boldsymbol{\beta}_o\boldsymbol{\beta}_o^{\mathrm{H}}\left|\frac{\lambda H_2(e^{-j\omega})}{1-\lambda e^{-j\omega}}\right|^2\sigma_w^2.
\end{aligned}$$

Observe that $\mathrm{tr}\{\mathbf{\Phi}\boldsymbol{\beta}_o\boldsymbol{\beta}_o^{\mathrm{H}}\} = \boldsymbol{\beta}_o^{\mathrm{H}}\mathbf{\Phi}\boldsymbol{\beta}_o = b^2$ and $\mathrm{tr}\{\mathbf{\Phi\Phi}^{-1}\} = \mathrm{tr}\{\mathbf{I}_n\} = n$, leading to

$$\begin{aligned}
\mathrm{tr}\{\mathbf{\Phi S}_{\Delta\widetilde{\boldsymbol{\beta}}}(\omega)\} &= \frac{\sigma_v^2}{2|1-\lambda e^{-j\omega}|^2}\{|\lambda b^2 H_1(e^{-j\omega})|^2\\
&\quad+\lambda\mu b^2 H_1(e^{-j\omega}) + \lambda\mu b^2 H_1(e^{j\omega}) + n\mu^2\}\\
&\quad+b^2\sigma_w^2\left|\frac{\lambda H_2(e^{-j\omega})}{1-\lambda e^{-j\omega}}\right|^2 = \frac{(2n-1)\sigma_v^2}{2}|F(e^{-j\omega})|^2\\
&\quad+\frac{\sigma_v^2}{2}\,|G_1(e^{-j\omega})|^2 + b^2\sigma_w^2\,|G_2(e^{-j\omega})|^2
\end{aligned}$$

where

$$\begin{aligned}
F(q^{-1}) &= \frac{1-\lambda}{1-\lambda q^{-1}}\\
G_1(q^{-1}) &= \frac{\lambda b^2 H_1(q^{-1}) + \mu}{1-\lambda q^{-1}} = \frac{1-\lambda+(\lambda-\delta)q^{-1}}{1-(\lambda+\delta)q^{-1}+\lambda q^{-2}}\\
G_2(q^{-1}) &= \frac{\lambda H_2(q^{-1})}{1-\lambda q^{-1}} = -\,\frac{\lambda}{1-(\lambda+\delta)q^{-1}+\lambda q^{-2}}.
\end{aligned}$$

Using (22) and (23), one obtains

$$\begin{aligned}
\mathrm{E}[\Delta\widetilde{\boldsymbol{\beta}}^{\mathrm{H}}(t)\mathbf{\Phi}\Delta\widetilde{\boldsymbol{\beta}}(t)] &= \frac{1}{2\pi}\int_{-\pi}^{\pi}\mathrm{tr}\{\mathbf{\Phi S}_{\Delta\widetilde{\boldsymbol{\beta}}}(\omega)\}\,d\omega\\
&= \frac{(2n-1)\sigma_v^2}{2}I[F(z^{-1})] + \frac{\sigma_v^2}{2}I[G_1(z^{-1})]\\
&\quad+ b^2\sigma_w^2 I[G_2(z^{-1})].
\end{aligned}$$

Finally, by means of residue calculus, one arrives at

$$I[F(z^{-1})] = \frac{1-\lambda}{1+\lambda} \cong \frac{\mu}{2}$$

$$I[G_1(z^{-1})] = \frac{1+\delta-\lambda-3\lambda\delta+2\lambda^2}{(1-\lambda)(1+2\lambda+\delta)} \cong \frac{\gamma}{2\mu} + \frac{\mu}{2}$$

$$I[G_2(z^{-1})] = \frac{\lambda^2(1+\lambda)}{(1-\lambda)(1-\delta)(1+2\lambda+\delta)} \cong \frac{1}{2\mu\gamma}$$

which leads in a straightforward way to (14).

REFERENCES

[1] M. Niedźwiecki and P. Kaczmarek, "Estimation and tracking of quasi-periodically varying processes," in *Proc. 13th IFAC Symp. System Ident.*, Rotterdam, The Netherlands, 2003, pp. 1102–1107.

[2] ——, "Generalized adaptive notch filters," in *Proc. 2004 IEEE Int. Conf. Acoust., Speech, Signal Process.*, Montreal, PQ, Canada, 2004, pp. II-657–II-660.

[3] ——, "Identification of quasi-periodically varying systems using the combined nonparametric/parametric approach," *IEEE Trans. Signal Process.*, vol. 53, pp. 4588–4598, 2005.

[4] M. K. Tsatsanis and G. B. Giannakis, "Modeling and equalization of rapidly fading channels," *Int. J. Adapt. Contr. Signal Process.*, vol. 10, pp. 159–176, 1996.

[5] G. B. Giannakis and C. Tepedelenlioglu, "Basis expansion models and diversity techniques for blind identification and equalization of time-varying channels," *Proc. IEEE*, vol. 86, pp. 1969–1986, 1998.

[6] J. Bakkoury, D. Roviras, M. Ghogho, and F. Castanie, "Adaptive MLSE receiver over rapidly fading channels," *Signal Process.*, vol. 80, pp. 1347–1360, 2000.

[7] P. Tichavský and P. Händel, "Two algorithms for adaptive retrieval of slowly time-varying multiple cisoids in noise," *IEEE Trans. Signal Process.*, vol. 43, pp. 1116–1127, 1995.

[8] P. Tichavský and A. Nehorai, "Comparative study of four adaptive frequency trackers," *IEEE Trans. Signal Process.*, vol. 45, pp. 1473–1484, 1997.

[9] M. Niedźwiecki, *Identification of Time-Varying Processes*. New York: Wiley, 2000.

[10] M. Niedźwiecki and P. Kaczmarek, "Tracking analysis of a generalized adaptive notch filter," *IEEE Trans. Signal Process.*, vol. 54, pp. 304–314, 2006.

[11] K. J. Kushner, *Approximation and Weak Convergence Methods for Random Processes with Applications to Stochastic System Theory*. Cambridge, MA: MIT Press, 1984.

[12] M. Niedźwiecki, "Identification of nonstationary stochastic systems using parallel estimation schemes," *IEEE Trans. Autom. Control*, vol. 35, pp. 329–334, 1990.

[13] T. Söderström and P. Stoica, *System Identification*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[14] K. J. Kushner and J. Yang, "Analysis of adaptive stepsize SA algorithms for parameter tracking," *IEEE Trans. Autom. Control*, vol. 40, pp. 1403–1410, 1995.

[15] A. Benveniste, M. Métivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximations*. New York: Springer-Verlag, 1990.

[16] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[17] M. V. Dragošević and S. S. Stanković, "An adaptive notch filter with improved tracking properties," *IEEE Trans. Signal Process.*, vol. 43, pp. 2068–2077, 1995.

[18] ——, "Fully adaptive constrained notch filter for tracking multiple frequencies," *Electron. Lett.*, vol. 31, pp. 1215–1217, 1995.

[19] M. Jury, *Theory and Application of the $\mathcal{Z}$-Transform Method*. New York: Wiley, 1964.

[20] B. D. O. Anderson, *Stability of Adaptive Systems: Passivity and Averaging Analysis*. Cambridge, MA: MIT Press, 1986.

Maciej Niedźwiecki was born in Poznań, Poland, in 1953. He received the M.Sc. and Ph.D. degrees from Gdańsk University of Technology, Gdańsk, Poland, in 1977 and 1981, respectively, and the Dr.Hab. (D.Sc.) degree from the Technical University of Warsaw, Warsaw, Poland, in 1991.

He spent three years (1986–1989) as a Research Fellow with the Department of Systems Engineering, Australian National University. In 1990–1993, he was Vice Chairman of the Technical Committee on Theory of the International Federation of Automatic Control. He is a Professor and Head of the Department of Automatic Control, Faculty of Electronics, Telecommunications and Computer Science, Gdańsk University of Technology. He is author of *Identification of Time-Varying Processes* (New York: Wiley, 2000). His main areas of research interests include system identification, signal processing, and adaptive systems.

Piotr Kaczmarek received the M.Sc. degree in automatic control from Gdańsk University of Technology, Gdańsk, Poland, in 2000, where he is currently pursuing the Ph.D. degree. He also graduated from the European Master Degree Course in Control and Management of Lean Manufacturing in Network Systems conducted in cooperation with Gdańsk University of Technology, Catholic University of Louvain, Belgium, and the University of Karlsruhe, Germany.

Since 1998, he has been involved in many projects for the biggest Polish air conditioning systems manufacturers. His interests include system identification and adaptive filtering as well as optimization of production techniques.