Generalized Adaptive Notch Filters for Identification of Real Quasi-Periodically Varying Systems

*Maciej Niedźwiecki and Piotr Kaczmarek**
Faculty of Electronics, Telecommunications and Computer Science
Department of Automatic Control, Technical University of Gdańsk
ul. Narutowicza 11/12, Gdańsk , Poland
maciekn@eti.pg.gda.pl, piokacz@proterians.net.pl

ABSTRACT

The problem of identification/tracking of quasi-periodically varying systems is considered. This problem is a generalization, to the system case, of a classical signal processing task of either elimination or extraction of nonstationary sinusoidal signals buried in noise.

1 Problem statement

Consider the problem of identification/tracking of coefficients of a time varying system governed by

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

where $t = 1, 2, \ldots$ denotes the normalized discrete time, $y(t)$ denotes the system output, $\boldsymbol{\varphi}(t) = [u(t-1), \ldots, u(t-n)]^{\mathrm{T}}$ is the regression vector, made up of the past input samples, $v(t)$ is an additive white noise, uncorrelated with $\boldsymbol{\varphi}(t)$, and $\boldsymbol{\theta}(t) = [\theta_1(t), \ldots, \theta_n(t)]^{\mathrm{T}}$ denotes the vector of time varying system coefficients, modeled as weighted sums of sinusoidal and cosinusoidal functions

$$\begin{aligned} \theta_l(t) \ &= \ c_{l,\,0}(t) + \sum_{i=1}^{k}[c_{l,\,2i-1}(t)\sin\phi_i(t) \\ &+ \ c_{l,\,2i}(t)\cos\phi_i(t)] + c_{l,\,2k+1}(t)(-1)^t \quad (2) \\ & l = 1, \ldots, n \end{aligned}$$

where

$$\phi_i(t) = \sum_{s=1}^{t} \omega_i(s) \quad (3)$$

Inclusion of the DC component (with characteristic frequency $\omega_0 = 0$) and AC component (with characteristic frequency $\omega_{k+1} = \pi$) is optional - both terms were added merely for the sake of completeness. term

We will assume that for every i, the quantities $c_{l,\,i}(t), l = 1, \ldots, n$ and $\omega_i(t)$ are slowly time-varying. The system, governed by (1) - (2), which obeys the abovementioned limitation, will be further referred to as *quasi-periodically* time-varying.

of

*This work was supported by KBN under Grant 4 T11A 01225.

One of the challenging applications of identification of quasi-periodically varying systems is adaptive equalization of rapidly fading communication channels – see e.g. [1].

Note that in the special case where $n = 1$, and $u(t) = 1, \forall t$, the model (1) - (2) becomes a description of a nonstationary multifrequency *signal* buried in noise. Hence the problem solved in our paper is a generalization, to the system case, of a classical signal processing task of either elimination or extraction of nonstationary sinusoidal signals buried in noise.

The problem of identification of quasi-periodically varying complex systems (systems with input/output signals and parameters described by complex numbers) was analyzed in [3]. It should be stressed that, although sharing some common ideas and frameworks with [3], the material presented below is neither a special case, nor can it be obtained as a trivial extension of the results derived for complex systems.

2 Known frequencies

Suppose, for the time being, that both the amplitudes and angular frequencies in (2) are constant, i.e. that system parameters are almost periodic functions of time, governed by

$$\begin{aligned} \theta_l(t) \ &= \ c_{l,\,0} + \sum_{i=1}^{k}[c_{l,\,2i-1}\sin\omega_i t \\ &+ \ c_{l,\,2i}\cos\omega_i t] + c_{l,\,2k+1}(-1)^t \quad (4) \\ & l = 1, \ldots, n \end{aligned}$$

Denote by $\mathbf{f}_i(t)$ the basis functions associated with the ith mode: $\mathrm{f}_0(t) = 1$, $\mathrm{f}_{k+1}(t) = (-1)^t$ and

$$\mathbf{f}_i(t) = \begin{bmatrix} \sin\omega_i t \\ \cos\omega_i t \end{bmatrix}, \quad i = 1, \ldots, k$$

Denote by $\boldsymbol{\alpha}_i$ the corresponding vectors of coefficients $\boldsymbol{\alpha}_0 = [c_{1,\,0}, \ldots, c_{n,\,0}]^{\mathrm{T}}$, $\boldsymbol{\alpha}_{k+1} = [c_{1,\,2k+1}, \ldots, c_{n,\,2k+1}]^{\mathrm{T}}$, $\boldsymbol{\alpha}_i = [c_{1,\,2i-1}, c_{1,\,2i}, \ldots, c_{n,\,2i-1}, c_{n,\,2i}]^{\mathrm{T}}$, $i = 1, \ldots, k$. Finally, let $\boldsymbol{\psi}_i(t) = \boldsymbol{\varphi}(t) \otimes \mathbf{f}_i(t)$, where $\otimes$ denotes the Kronecker product of two matrices/vectors.

Note that $\boldsymbol{\alpha}_i$ is the vector of coefficients associated with a particular frequency ω_i and *not* with a particular impulse response parameter $\theta_i(t)$. Similarly, $\boldsymbol{\psi}_i(t)$ is the generalized regression vector associated with the ith frequency component. Such parameterization will allow us to easily derive the frequency-decoupled versions of the basic estimation algorithm. Using the short-hand notation, introduced above, (1) can be rewritten in the form

$$y(t) = \sum_{i=0}^{k+1} \boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{\alpha}_i + v(t) = \boldsymbol{\psi}^{\mathrm{T}}(t)\boldsymbol{\alpha} + v(t) \qquad (5)$$

where $\boldsymbol{\psi}(t) = [\boldsymbol{\psi}_0^{\mathrm{T}}(t), \ldots, \boldsymbol{\psi}_{k+1}^{\mathrm{T}}(t)]^{\mathrm{T}}$ and $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_0^{\mathrm{T}}, \ldots, \boldsymbol{\alpha}_{k+1}^{\mathrm{T}}]^{\mathrm{T}}$.

Suppose now that the vector $\boldsymbol{\alpha}$ is slowly varying with time. It is known that, in the case considered, one can track $\boldsymbol{\alpha}(t)$ using the method of exponentially weighted least squares (EWLS). The EWLS estimate of $\boldsymbol{\alpha}(t)$ can be obtained from

$$\widehat{\boldsymbol{\alpha}}(t) = \arg\min_{\boldsymbol{\alpha}} \sum_{s=0}^{t-1} \lambda^s \left(y(t-s) - \boldsymbol{\psi}^{\mathrm{T}}(t-s)\boldsymbol{\alpha}\right)^2$$

where λ $(0 < \lambda < 1, 1-\lambda \ll 1)$ denotes the so-called forgetting constant - the design parameter which controls the memory of the estimator, and hence allows one to trade off between its tracking speed and tracking accuracy.

The recursive algorithm for evaluation of $\widehat{\boldsymbol{\theta}}(t)$ is given by

$$\begin{aligned}
\widehat{\boldsymbol{\alpha}}(t) &= \widehat{\boldsymbol{\alpha}}(t-1) + \mathbf{k}_\alpha(t)\varepsilon(t) \\
\varepsilon(t) &= y(t) - \boldsymbol{\psi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\alpha}}(t-1) \\
\mathbf{k}_\alpha(t) &= \frac{\mathbf{P}_\alpha(t-1)\boldsymbol{\psi}(t)}{\lambda + \boldsymbol{\psi}^{\mathrm{T}}(t)\mathbf{P}_\alpha(t-1)\boldsymbol{\psi}(t)} \\
\mathbf{P}_\alpha(t) &= \lambda^{-1}[\mathbf{P}_\alpha(t-1) - \mathbf{k}_\alpha(t)\boldsymbol{\psi}^{\mathrm{T}}(t)\mathbf{P}_\alpha(t-1)] \\
\mathbf{f}_i(t) &= \mathbf{G}_i\mathbf{f}_i(t-1) \\
\boldsymbol{\psi}_i(t) &= \boldsymbol{\varphi}(t) \otimes \mathbf{f}_i(t) \\
\mathbf{D}_i(t) &= \mathbf{I}_n \otimes \mathbf{f}_i^{\mathrm{T}}(t) \\
i &= 0, \ldots, k+1 \\
\widehat{\boldsymbol{\theta}}(t) &= \sum_{i=0}^{k+1} \mathbf{D}_i(t)\widehat{\boldsymbol{\alpha}}_i(t) \qquad (6)
\end{aligned}$$

where $\mathbf{D}_i(t) = \mathbf{I}_n \otimes \mathbf{f}_i^{\mathrm{T}}(t) = \operatorname{diag}_n\{\mathbf{f}_i^{\mathrm{T}}(t), \ldots, \mathbf{f}_i^{\mathrm{T}}(t)\}$, $\mathbf{I}_n$ denotes the $n \times n$ identity matrix, $\mathbf{G}_0 = g_0 = 1$, $\mathbf{G}_{k+1} = g_{k+1} = -1$, and

$$\mathbf{G}_i = \begin{bmatrix} \cos\omega_i & \sin\omega_i \\ -\sin\omega_i & \cos\omega_i \end{bmatrix} \quad i = 1, \ldots, k$$

denote orthogonal rotation matrices ($\mathbf{G}_i^{-1} = \mathbf{G}_i^{\mathrm{T}}$).

Since the algorithm (6) combines the basis function parameterization with exponentially weighted least squares estimation, it will be further referred to as the exponentially weighted basis function (EWBF) algorithm [2].

Another, equivalent form of the EWBF estimator, which will be very useful for our purposes, can be obtained by rewriting (6) in a different system of coordinates. Using the linear time-varying transformation

$$\widehat{\boldsymbol{\beta}}(t) = \mathbf{A}^{-(t+1)}\widehat{\boldsymbol{\alpha}}(t), \quad \mathbf{k}_\beta(t) = \mathbf{A}^{-t}\mathbf{k}_\alpha(t)$$

$$\mathbf{P}_\beta(t) = \mathbf{A}^{-(t+1)}\mathbf{P}_\alpha(t)\mathbf{A}^{t+1}$$

where $\mathbf{A} = \operatorname{diag}\{\mathbf{A}_0, \ldots, \mathbf{A}_{k+1}\}$, $\mathbf{A}^{-1} = \mathbf{A}^{\mathrm{T}}$, $\mathbf{A}_0 = \mathbf{I}_n$, $\mathbf{A}_{k+1} = -\mathbf{I}_n$ and $\mathbf{A}_i = \mathbf{I}_n \otimes \mathbf{G}_i$, $\mathbf{A}_i^{-1} = \mathbf{A}_i^{\mathrm{T}}$, $i = 1, \ldots, k$, one can convert (6) into

$$\begin{aligned}
\widehat{\boldsymbol{\beta}}(t) &= \mathbf{A}^{\mathrm{T}}[\widehat{\boldsymbol{\beta}}(t-1) + \mathbf{k}_\beta(t)\varepsilon(t)] \\
\varepsilon(t) &= y(t) - \boldsymbol{\zeta}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}(t-1) \\
\mathbf{k}_\beta(t) &= \frac{\mathbf{P}_\beta(t-1)\boldsymbol{\zeta}(t)}{\lambda + \boldsymbol{\zeta}^{\mathrm{T}}(t)\mathbf{P}_\beta(t-1)\boldsymbol{\zeta}(t)} \\
\mathbf{P}_\beta(t) &= \lambda^{-1}\mathbf{A}^{\mathrm{T}}[\mathbf{P}_\beta(t-1) - \mathbf{k}_\beta(t)\boldsymbol{\zeta}^{\mathrm{T}}(t)\mathbf{P}_\beta(t-1)]\mathbf{A} \\
\boldsymbol{\zeta}_i(t) &= \boldsymbol{\varphi}(t) \otimes \mathbf{f}_i(0), \quad i = 0, \ldots, k+1 \\
\widehat{\boldsymbol{\theta}}(t) &= \sum_{i=0}^{k+1} \mathbf{E}_i\widehat{\boldsymbol{\beta}}_i(t) \qquad (7)
\end{aligned}$$

where $\mathbf{E}_0 = \mathbf{I}_n$, $\mathbf{E}_{k+1} = -\mathbf{I}_n$ and

$$\begin{aligned}
\mathbf{E}_i &= \mathbf{D}_i(-1) = \mathbf{I}_n \otimes \mathbf{f}_i^{\mathrm{T}}(-1) = \operatorname{diag}_n\{\mathbf{e}_i^{\mathrm{T}}, \ldots, \mathbf{e}_i^{\mathrm{T}}\} \\
\mathbf{e}_i^{\mathrm{T}} &= [\,-\sin\omega_i,\ \cos\omega_i\,], \quad i = 1, \ldots, k
\end{aligned}$$

Note that the regression vector $\boldsymbol{\zeta}(t) = \mathbf{A}^{-t}\boldsymbol{\psi}(t) = [\boldsymbol{\zeta}_0^{\mathrm{T}}(t), \ldots, \boldsymbol{\zeta}_{k+1}^{\mathrm{T}}(t)]^{\mathrm{T}}$, which appears in (7), does not depend on the frequencies $\omega_1, \ldots, \omega_k$, while the components of the generalized regression vector $\boldsymbol{\psi}(t)$, appearing in the original algorithm (6), are frequency dependent. For this reason the transformed algorithm will be a more convenient starting point for derivation of the frequency-adaptive version of the EWBF algorithm.

2.1 Parallel decomposition

Let $\boldsymbol{\theta}^{(i)}(t) = \mathbf{D}_i(t)\boldsymbol{\alpha}_i$ be the vector of all parameter terms that change with the frequency ω_i. Once again, the quantity $\boldsymbol{\theta}^{(i)}(t)$ should not be confused with $\theta_i(t)$ - the ith component of $\boldsymbol{\theta}(t)$. Denote by

$$\begin{aligned}
y_i(t) &= \boldsymbol{\varphi}^{\mathrm{T}}(t)\boldsymbol{\theta}^{(i)}(t) + v(t) = \boldsymbol{\psi}_i^{\mathrm{T}}(t)\boldsymbol{\alpha}_i + v(t) \\
&= \boldsymbol{\zeta}_i^{\mathrm{T}}(t)\boldsymbol{\beta}_i(t) + v(t)
\end{aligned}$$

where $\boldsymbol{\zeta}_i(t) = \boldsymbol{\varphi}(t) \otimes \mathbf{f}_i(0)$ and $\boldsymbol{\beta}_i(t) = \mathbf{A}_i^{-(t+1)}\boldsymbol{\alpha}_i$, the output of the ith subsystem of (5), i.e. the subsystem associated with the frequency ω_i. Even though the signal $y_i(t)$ is not available, one can easily estimate it using the formula

$$\widehat{y}_i(t) = y(t) - \sum_{\substack{l=0 \\ l \neq i}}^{k+1} \widehat{y}_l(t|t-1)$$

where $\widehat{y}_i(t|t-1) = \boldsymbol{\psi}_i^{\mathrm{T}}(t)\widehat{\boldsymbol{\alpha}}_i(t-1) = \boldsymbol{\zeta}_i^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1)$ is the predicted value of $y_i(t)$, yielded by the estimation algorithm designed to track parameters of the ith subsystem.

Estimation of $y_i(t)$, in the way described above, allows one to decompose the tracking algorithm, i.e. to replace one 'global' algorithm (7) with $k+2$ mutually coupled 'local' algorithms, each of which takes care of a particular subsystem.

The ith component algorithm can be easily derived from (7) by setting $y(t) = \widehat{y}_i(t)$, $\boldsymbol{\zeta}(t) = \boldsymbol{\zeta}_i(t)$ and $\mathbf{A} = \mathbf{A}_i$. To add some extra design flexibility, we will equip each subalgorithm with an independently assigned forgetting factor λ_i. The resulting decoupled parallel-form algorithm can be written down as

$$
\begin{aligned}
\varepsilon_i(t) &= \widehat{y}_i(t) - \boldsymbol{\zeta}_i^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1) \\
\widehat{\boldsymbol{\beta}}_i(t) &= \mathbf{A}_i^{\mathrm{T}}[\widehat{\boldsymbol{\beta}}_i(t-1) + \mathbf{k}_i(t)\varepsilon_i(t)] \\
\mathbf{k}_i(t) &= \frac{\mathbf{P}_i(t-1)\boldsymbol{\zeta}_i(t)}{\lambda_i + \boldsymbol{\zeta}_i^{\mathrm{T}}(t)\mathbf{P}_i(t-1)\boldsymbol{\zeta}_i(t)} \\
\mathbf{P}_i(t) &= \frac{1}{\lambda_i}\mathbf{A}_i^{\mathrm{T}}[\mathbf{P}_i(t-1) - \mathbf{k}_i(t)\boldsymbol{\zeta}_i^{\mathrm{T}}(t)\mathbf{P}_i(t-1)]\mathbf{A}_i \\
i &= 0,\ldots,k+1 \\
\widehat{\boldsymbol{\theta}}(t) &= \sum_{i=0}^{k+1}\mathbf{E}_i\widehat{\boldsymbol{\beta}}_i(t) \qquad (8)
\end{aligned}
$$

Observe that $\varepsilon_0(t) = \ldots = \varepsilon_{k+1}(t) = y(t) - \sum_{i=0}^{k+1}\boldsymbol{\zeta}_i^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1) = \varepsilon(t)$. Hence all subalgorithms are in fact driven by the same global prediction error $\varepsilon(t)$.

2.2 Cascade decomposition

The decoupled algorithm presented in the previous subsection is a parallel structure made up of k identical (from the functional viewpoint) blocks. Each block is designed to track a particular frequency component of the parameter vector $\boldsymbol{\theta}(t)$. Connecting the same blocks so that they form a cascade, one obtains an interesting alternative to the parallel decomposition. To obtain the cascade variant of the EWBF algorithm, the first two recursions in (8) should be replaced with

$$
\begin{aligned}
\varepsilon_i(t) &= \varepsilon_{i-1}(t) - \boldsymbol{\zeta}_i^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1) \\
\widehat{\boldsymbol{\beta}}_i(t) &= \mathbf{A}_i^{\mathrm{T}}[\widehat{\boldsymbol{\beta}}_i(t-1) + \mathbf{k}_i(t)\varepsilon_i(t)] \qquad (9)
\end{aligned}
$$

where $\varepsilon_{-1}(t) = y(t)$.

3 Unknown frequencies

Even though the EWBF filter is robust to small local changes in frequencies, it will fail to identify the system correctly in the presence of a frequency drift. For this reason in this section we will derive two frequency-adaptive EWBF algorithms, capable of tracking the time-varying frequencies $\omega_i(t), i = 1,\ldots,k$.

Denote by $V(t,\omega_i) = \frac{1}{2}\sum_{s=1}^{t}\gamma_i^{t-s}\varepsilon_i^2(s)$ the local exponentially weighted measure of fit, where γ_i, $0<\gamma_i<1$, is the forgetting constant, which will be used to control the speed of the frequency adaptation. To evaluate the estimate $\widehat{\omega}_i(t) = \arg\min_{\omega_i} V(t,\omega_i)$ we will use the recursive prediction error (RPE) approach – see e.g. Söderström and Stoica [4].

Consider the following standardized form of the ith prediction error,

$$\varepsilon_i(t) = z_i(t) - \boldsymbol{\zeta}_i^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1)$$

where $z_i(t) = \widehat{y}_i(t)$, $i \geq 0$ in the parallel implementation, and $z_0(t) = y(t)$, $z_i(t) = \varepsilon_{i-1}(t)$, $i > 0$ in the cascade implementation.

Let $\varepsilon_i(t) = \varepsilon_i(t,\widehat{\omega}_i(t))$, $\widehat{\boldsymbol{\beta}}_i(t) = \widehat{\boldsymbol{\beta}}_i(t,\widehat{\omega}_i(t))$, $\widehat{\mathbf{A}}_i(t) = \mathbf{A}_i(\widehat{\omega}_i(t))$, $\eta_i(t) = \partial\varepsilon_i(t,\widehat{\omega}_i(t-1))/\partial\omega_i$, $\boldsymbol{\xi}_i(t) = \partial\widehat{\boldsymbol{\beta}}_i(t,\widehat{\omega}_i(t))/\partial\omega_i$, $\mathbf{k}_i(t) = \mathbf{k}_i(t,\widehat{\omega}_i(t))$, $\widetilde{\mathbf{k}}_i(t) = \partial\mathbf{k}_i(t,\widehat{\omega}_i(t-1))/\partial\omega_i$, $\mathbf{P}_i(t) = \mathbf{P}_i(t,\widehat{\omega}_i(t))$, $\widetilde{\mathbf{P}}_i(t) = \partial\mathbf{P}_i(t,\widehat{\omega}_i(t))/\partial\omega_i$ and $\nu_i(t) = V_i''(t,\widehat{\omega}_i(t-1))$.

Using the shorthands introduced above and noting that $\boldsymbol{\zeta}_0(t) = \boldsymbol{\zeta}_{k+1}(t) = \boldsymbol{\varphi}(t)$, $\boldsymbol{\zeta}_1(t) = \cdots = \boldsymbol{\zeta}_k(t) = \boldsymbol{\varphi}_0(t) = [0,u(t-1),\ldots,0,u(t-n)]^{\mathrm{T}}$, the frequency-adaptive parallel-form and cascade-form EWBF algorithms can be written down in the following unified form

$$
\begin{aligned}
\varepsilon_0(t) &= z_0(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_0(t-1) \\
\delta_0(t) &= \lambda_0 + \boldsymbol{\varphi}^{\mathrm{T}}(t)\mathbf{P}_0(t-1)\boldsymbol{\varphi}(t) \\
\mathbf{k}_0(t) &= \delta_0^{-1}(t)\mathbf{P}_0(t-1)\boldsymbol{\varphi}(t) \\
\widehat{\boldsymbol{\beta}}_0(t) &= \widehat{\boldsymbol{\beta}}_0(t-1) + \mathbf{k}_0(t)\varepsilon_0(t) \\
\mathbf{P}_0(t) &= \lambda_0^{-1}[\mathbf{P}_0(t-1) - \mathbf{k}_0(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)\mathbf{P}_0(t-1)]
\end{aligned}
$$

$$
\begin{aligned}
\varepsilon_i(t) &= z_i(t) - \boldsymbol{\varphi}_0^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_i(t-1) \\
\eta_i(t) &= -\boldsymbol{\varphi}_0^{\mathrm{T}}(t)\boldsymbol{\xi}_i(t-1) \\
\nu_i(t) &= \gamma_i\nu_i(t-1) + \eta_i^2(t) \\
\widehat{\omega}_i(t) &= \widehat{\omega}_i(t-1) - \nu_i^{-1}(t)\eta_i(t)\varepsilon_i(t) \\
\widehat{\mathbf{G}}_i(t) &= \begin{bmatrix} \cos\widehat{\omega}_i(t) & \sin\widehat{\omega}_i(t) \\ -\sin\widehat{\omega}_i(t) & \cos\widehat{\omega}_i(t) \end{bmatrix} \\
\widehat{\mathbf{A}}_i(t) &= \operatorname*{diag}_n\{\widehat{\mathbf{G}}_i(t),\ldots,\widehat{\mathbf{G}}_i(t)\} \\
\widehat{\mathbf{e}}_i^{\mathrm{T}}(t) &= [\,-\sin\widehat{\omega}_i(t)\ ,\ \cos\widehat{\omega}_i(t)\,] \\
\widehat{\mathbf{E}}_i(t) &= \operatorname*{diag}_n\{\,\widehat{\mathbf{e}}_i^{\mathrm{T}}(t),\ldots,\widehat{\mathbf{e}}_i^{\mathrm{T}}(t)\} \\
\delta_i(t) &= \lambda_i + \boldsymbol{\varphi}_0^{\mathrm{T}}(t)\mathbf{P}_i(t-1)\boldsymbol{\varphi}_0(t) \\
\mathbf{k}_i(t) &= \delta_i^{-1}(t)\mathbf{P}_i(t-1)\boldsymbol{\varphi}_0(t) \\
\widetilde{\mathbf{k}}_i(t) &= \delta_i^{-1}(t)[\widetilde{\mathbf{P}}_i(t-1)\boldsymbol{\varphi}_0(t) \\
&\quad - \mathbf{k}_i(t)\boldsymbol{\varphi}_0^{\mathrm{T}}(t)\widetilde{\mathbf{P}}_i(t-1)\boldsymbol{\varphi}_0(t)] \\
\widehat{\boldsymbol{\beta}}_i(t) &= \widehat{\mathbf{A}}_i^{\mathrm{T}}(t)[\widehat{\boldsymbol{\beta}}_i(t-1) + \mathbf{k}_i(t)\varepsilon_i(t)] \\
\mathbf{P}_i(t) &= \lambda_i^{-1}\widehat{\mathbf{A}}_i^{\mathrm{T}}(t)[\mathbf{P}_i(t-1) \\
&\quad - \mathbf{k}_i(t)\boldsymbol{\varphi}_0^{\mathrm{T}}(t)\mathbf{P}_i(t-1)]\widehat{\mathbf{A}}_i(t) \\
\widetilde{\mathbf{P}}_i(t) &= \mathbf{J}^{\mathrm{T}}\mathbf{P}_i(t) + \mathbf{P}_i(t)\mathbf{J} + \lambda_i^{-1}\widehat{\mathbf{A}}_i^{\mathrm{T}}(t)[\widetilde{\mathbf{P}}_i(t-1) \\
&-\widetilde{\mathbf{k}}_i(t)\boldsymbol{\varphi}_0^{\mathrm{T}}(t)\mathbf{P}_i(t-1) - \mathbf{k}_i(t)\boldsymbol{\varphi}_0^{\mathrm{T}}(t)\widetilde{\mathbf{P}}_i(t-1)]\widehat{\mathbf{A}}_i(t)
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\xi}_i(t) &= \mathbf{J}^{\mathrm{T}}\widehat{\boldsymbol{\beta}}_i(t) + \widehat{\mathbf{A}}_i^{\mathrm{T}}(t)[\boldsymbol{\xi}_i(t-1) \\
&+ \mathbf{k}_i(t)\eta_i(t) + \widetilde{\mathbf{k}}_i(t)\varepsilon_i(t)] \\
i &= 1,\ldots,k \\
\varepsilon_{k+1}(t) &= z_{k+1}(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_{k+1}(t-1) \\
\delta_{k+1}(t) &= \lambda_{k+1} + \boldsymbol{\varphi}^{\mathrm{T}}(t)\mathbf{P}_{k+1}(t-1)\boldsymbol{\varphi}(t) \\
\mathbf{k}_{k+1}(t) &= \delta_{k+1}^{-1}(t)\mathbf{P}_{k+1}(t-1)\boldsymbol{\varphi}(t) \\
\widehat{\boldsymbol{\beta}}_{k+1}(t) &= -\widehat{\boldsymbol{\beta}}_{k+1}(t-1) - \mathbf{k}_{k+1}(t)\varepsilon_{k+1}(t) \\
\mathbf{P}_{k+1}(t) &= \lambda_{k+1}^{-1}[\mathbf{P}_{k+1}(t-1) \\
&- \mathbf{k}_{k+1}(t)\boldsymbol{\varphi}^{\mathrm{T}}(t)\mathbf{P}_{k+1}(t-1)]
\end{aligned}
$$

$$
\widehat{\boldsymbol{\theta}}(t) = \widehat{\boldsymbol{\beta}}_0(t) - \widehat{\boldsymbol{\beta}}_{k+1}(t) + \sum_{i=1}^{k}\widehat{\mathbf{E}}_i(t)\widehat{\boldsymbol{\beta}}_i(t) \quad (10)
$$

where $\mathbf{J} = \mathbf{I}_n \otimes \mathbf{J}_o = \operatorname{diag}_n\{\mathbf{J}_o,\ldots,\mathbf{J}_o\}$ and

$$
\mathbf{J}_o = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}
$$

Note that in the parallel realization $\varepsilon_0(t) = \ldots = \varepsilon_{k+1}(t) = y(t) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_0(t-1) - \boldsymbol{\varphi}^{\mathrm{T}}(t)\widehat{\boldsymbol{\beta}}_{k+1}(t-1) - \boldsymbol{\varphi}_0^{\mathrm{T}}(t)\sum_{i=1}^{k}\widehat{\boldsymbol{\beta}}_i(t-1)$.

4 Computer simulations

The system identification/tracking results, shown in the figures 1 and 2, were obtained for a hypothetical two-tap (n=2) time-varying communication channel with $\theta_1(t) = 0.5 + 2\sin\phi_1(t) - 0.5\sin\phi_2(t) + 0.7\sin\phi_3(t) - 0.2\sin\phi_4(t)$, $\theta_2(t) = -1 - 0.5\cos\phi_1(t) + 1.5\cos\phi_2(t) - 0.7\cos\phi_3(t) - 1.5\cos\phi_4(t)$. The uncorrelated PRBS sequence ($u(t) = \pm 1$) was used as the input signal and the noise was white and Gaussian with variance $\sigma_v^2(t) = 0.05$ (average SNR $\cong$ 20 dB). The forgetting factors were $\lambda_0 = \lambda_1 = \ldots = \lambda_4 = 0.99$ and $\gamma_1 = \ldots = \gamma_4 = 0.98$.
The simulated frequency changes were of the ramp type. To check the 'steady state' tracking capabilities of the compared algorithms, the linear changes in frequencies were enforced only after the initial convergence period was over. Since the trajectories of $\omega_1(t)$ and $\omega_2(t)$ intersect in the middle of the analysis interval, one of our main concerns was the behavior of the tracking algorithms in the vicinity of the crossover point. Although both algorithms show satisfactory tracking performance, the parallel-form adaptive notch filter yields consistently better results than the cascade-form filter. This is confirmed by both the frequency estimation plots and parameter tracking plots.

References

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

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

[3] Niedźwiecki, M. and P. Kaczmarek (2003). Estimation and Tracking of Quasi-Periodically Varying Processes. *13th IFAC Symposium on System Identification*, 1102–1107.

[4] Söderström, T. and P. Stoica (1988). *System Identification.* Prentice Hall. Englewood Cliffs NJ.

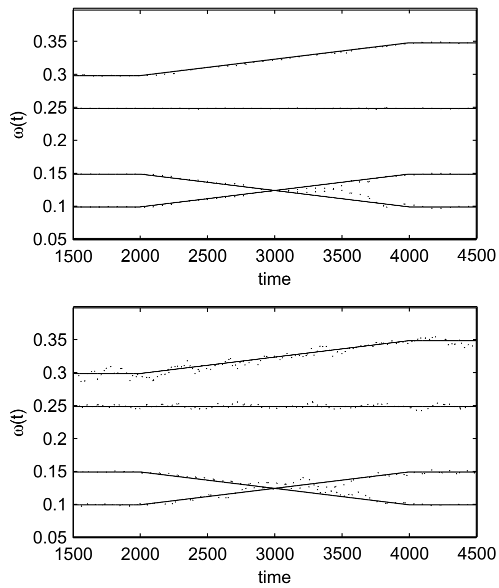

Figure 1: Instantaneous Doppler frequencies of a simulated mobile radio channel (solid lines) and their estimates (dotted lines) obtained using the parallel-form algorithm (upper plot) and the cascade-form algorithm (lower plot).

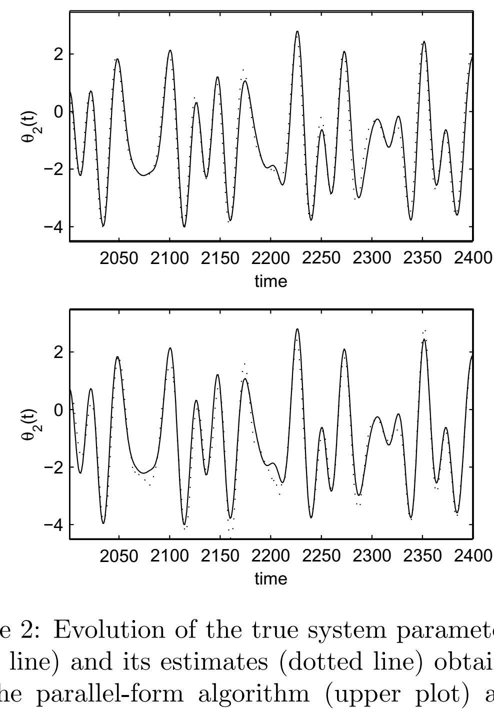

Figure 2: Evolution of the true system parameter $\theta_2(t)$ (solid line) and its estimates (dotted line) obtained using the parallel-form algorithm (upper plot) and the cascade-form algorithm (lower plot).