Stabilizing the LFTF algorithm by leakage control

Bernhard Nitsch Stephan Binde
Technische Hochschule Darmstadt, Institut für Netzwerk– und Signaltheorie
Merckstr. 25, D-64283 Darmstadt, Germany
e-mail: nitsch@nesi.e-technik.th-darmstadt.de

ABSTRACT

To stabilize the FTF algorithm [1] the accumulation of numerical errors can be prevented by introducing a leakage factor in the equation system as presented in [7]. In state space description the leakage factor causes a reduction of the eigenvalues of the linearized error system matrix. By an appropriate choice of the leakage factor the eigenvalues can be transformed into the unit circle of the z-plane resulting in a stable round-off error system. The structure of the linearized error system matrix shall be analysed and by comparing the Leakage FTF algorithm (LFTF) with the Stabilized FTF algorithm (SFTF) and the NLMS algorithm in a real-time environment the success of this method is shown.

1 Stability analysis

The LFTF algorithm as presented in table 1 uses a leakage in the filter equations of the forward and backward prediction error filters resulting in a reduction of numerical errors due to round-off noise. A different method to compute the conversion factor $\gamma_M(n)$ by a filtering operation and not a scalar one leads to a very robust behavior of the algorithm especially for speech input. The following analysis examines the dependence of a stable performance on the choice of the leakage factor. A state space analysis shows how round-off errors increase or decrease depending on this leakage factor. Stability can be reached if the leakage factor v is correctly adjusted.
Describing the LFTF algorithm with state variables yields the nonlinear state space equation

$$\underline{\Psi}(n) = f\{\underline{\Psi}(n-1), \underline{u}_{M+1}(n)\}, \qquad (1)$$

where $\underline{\Psi}(n)$ is the state vector and $\underline{u}_{M+1}(n)$ the vector of the excitation sequence. By realizing the algorithm in finite precision arithmetic round-off errors occur which let the algorithm follow a perturbed trajectory $\hat{\underline{\Psi}}(n)$ resulting in unstable behavior. If the perturbations $\underline{\Psi}(n) - \hat{\underline{\Psi}}(n)$ due to numerical errors are small and the wordlength is sufficiently long the system (1) can be linearized around the unperturbed trajectory $\underline{\Psi}$:

$$\Delta\underline{\Psi}(n) = A(n) \cdot \Delta\underline{\Psi}(n-1) + \underline{V}(n). \qquad (2)$$

$A(n)$ describes the matrix of the linearized error system and $\underline{V}(n)$ the component due to round-off noise.
The LFTF algorithm (table 1) can be analysed in state space description by defining the state vector

$$\begin{aligned} \Delta\underline{\Psi}^T(n) \quad = \quad & [\, \Delta\underline{a}_M^T(n), \Delta F_M(n), \Delta\underline{c}_M^T(n), \\ & \Delta B_M(n), \Delta\tilde{\underline{k}}_M^T(n), \Delta\gamma_M(n)\,]. \end{aligned} \qquad (3)$$

By calculating the partial differentials $\dfrac{\partial\underline{\Psi}_i^T(n)}{\partial\underline{\Psi}_j(n-1)}$ the system matrix

$$A(n) = \begin{bmatrix} F_a(v,n) & * & * & * & * & * \\ 0 & \lambda & * & * & * & * \\ 0 & 0 & F_c(v,n) & * & * & * \\ 0 & 0 & * & f_B(v,n) & * & * \\ * & 0 & * & * & F_k(v,n) & * \\ * & * & * & * & 0 & 0 \end{bmatrix}, \qquad (4)$$

of the linearized error system can be derived. Elements indicated with an asterisk are nonzero but their influence on the error system is negligible.
Using the averaging technique introduced in [3] it can be shown that the linear time-variant error system (2) converges under some assumptions to a linear time-invariant error system

$$\overline{\Delta\underline{\Psi}}(n) = \overline{A} \cdot \overline{\Delta\underline{\Psi}}(n-1), \qquad (5)$$

with a system matrix $\overline{A}$ that has only diagonal nonzero elements depending on the leakage factor v. By calculating these elements that are equal to the eigenvalues of the error system (5), stability can be examined.
An analysis of the eigenvalues of the forward prediction part shows no instabilities because all eigenvalues are less or equal to one:

$$F_a(v) = \begin{bmatrix} 1 & * \\ 0 & \lambda \cdot v \cdot I_M \end{bmatrix}. \qquad (6)$$

While the elements of $F_k(v)$ describe the zeros of the backward prediction filter which lie inside the unit circle as it was shown in [3] this matrix has only non-critical elements. In the same way the elements of the forward prediction part are not critical as long as v and λ are smaller than 1. The method of the LFTF algorithm to calculate the conversion factor $\gamma_M(n)$ by a filtering operation and not by recursive scalar operations results in the independence of $\gamma_M(n)$ and $\gamma_M(n-1)$. For this reason the eigenvalue of the conversion part of the error system $\overline{A}$ is zero and causes no instability therefore.

The critical element in the algorithm is the backward predictor with the matrix $F_c(v)$ and the scalar $f_B(v)$. If there is no stabilization Gilloire and Slock have shown [2, 3] that the eigenvalues of the backward prediction part converge to $1/\lambda$ resulting in an exponentially unstable performance.

An analysis of the LFTF algorithm shows that there is only one critical variable in the algorithm, the matrix $F_c(v)$ of the backward prediction part:

$$F_c(v) = \left\{ v + [v^2(2-\lambda)] \left(\frac{B_M(n)}{\lambda\, B_M(n-1)} - 1 \right) \right\} I_M . \quad (7)$$

Without leakage control all eigenvalues of the backward prediction part lie outside the unit circle and the algorithm is exponentially unstable. This behavior is illustrated in figure 1, a computer simulation with white noise excitation: The eigenvalues of the backward system error matrix $F_c(v)$ are all greater than 1 so the algorithm becomes unstable. By choosing an appropriate value for the leakage factor the eigenvalues can be transformed into the unit circle of the z-plane resulting in a stable error system as shown in figure 2.

To guarantee fast convergence properties of the algorithm the leakage factor should be as close to 1 as possible. On the other hand stabilization requires eigenvalues inside the unit circle resulting in a leakage factor that fulfills the condition

$$v + [v^2(2-\lambda)] \left(\frac{B_M(n)}{\lambda\, B_M(n-1)} - 1 \right) \leq 1. \quad (8)$$

Equation (8) states the dependence of the leakage factor v on the relation of the actual backward prediction error energy $B_M(n)$ and the previous one $B_M(n-1)$.

With a constant leakage factor no stabilization can be guaranteed. The excitation signal always contains sections in which the short-time energy increases. This behavior influences the term $B_M(n)/[\lambda\, B_M(n-1)]$, which also grows. If there are strong fluctuations in the short-time input power as in speech signals the above quotient results in large values. In this case the eigenvalues lie far outside the unit circle and make the algorithm unstable immediately.

An appropriate leakage factor should depend on the mentioned quotient in (8) so that an elimination of the backward prediction error energies is possible. A problem that arises here is that the actual backward prediction error energy $B_M(n)$ is unknown at the time the leakage factor should be calculated, actually the calculation of $B_M(n)$ depends on the leakage factor. Knowing the fact that forward and backward error energies are

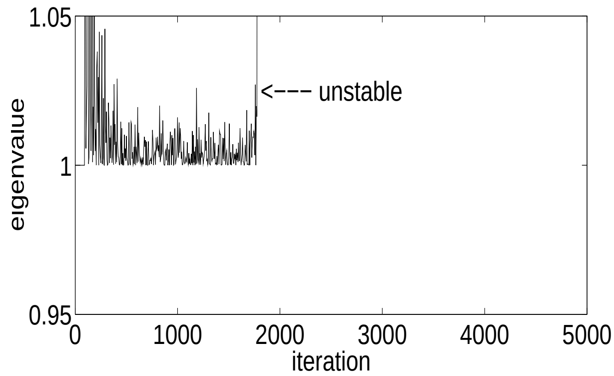

Figure 1: eigenvalues of the backward prediction part without stabilization

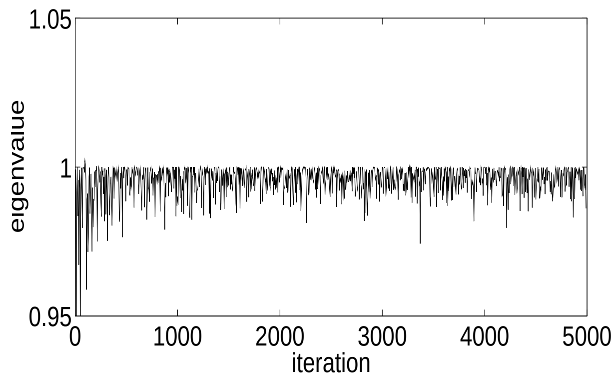

Figure 2: eigenvalues of the backward prediction part with stabilization by leakage control

concatenated by the conversion factor, $\gamma_M(n)\, F_M(n) = \lambda^M\, B_M(n)$, and that the conversion factor $\gamma_M(n)$ shows only slow variations in time a replacement of the backward prediction error energy B_M by the forward prediction error energy F_M can be performed.

We propose a suitable leakage factor:

$$v = \frac{\lambda\, F_M(n-1)}{F_M(n)}. \quad (9)$$

This choice guarantees a leakage factor close to 1 if the excitation power shows no strong variations ($F_M(n) \approx F_M(n-1)$). On the other hand an abrupt increase in the excitation power ($F_M(n) >> F_M(n-1)$) reduces the leakage factor and leads to small eigenvalues which stabilize the algorithm when steep slopes of the excitation short-time power occur.

2 Real-time implementations

To compare the stability and tracking ability of the Leakage FTF algorithm (LFTF) with other algorithms such as the NLMS [1] or the Stabilized FTF algorithm (SFTF) by Slock [3] all algorithms were implemented on

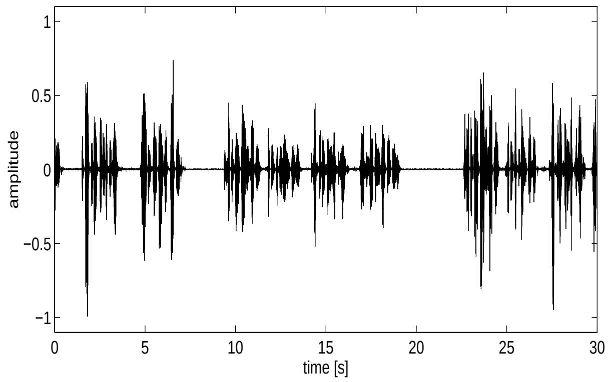

Figure 3: excitation signal (male speaker)

a multi digital signal processor system based on processor type 32C from AT&T.

The algorithms work in a frequency subband environment described in [6]. Due to complexity reduction of the subband concept by using a sampling rate reduction of 13 large filter lengths are possible although the algorithm complexity of the FTF algorithms is high ($11M$ for the LFTF- and $9M$ for the SFTF algorithm). We use a 16 channel subband filter bank sampled by 8 kHz and a subband compensator length of 80 coefficients, which can be compared to 890 coefficients in a fullband filter. Due to the filter design of the analysis- and synthesis filterbanks for subband processing a maximum echo reduction loss enhancement (ERLE) of -40 dB of can be achieved.

The algorithms were tested with speech as excitation signal illustrated in figure 3. The input sequence contains several parts of speech pauses to test how the algorithms behave in case of non persistent excitation. For comparable measurements the Loudspeaker-Room-Microphone (LRM) system was simulated by two measured impulse responses of an office room. To illustrate the tracking ability of all algorithms a switch over the two - 950 coefficients long - impulse responses was performed after 10 seconds. We choose no background noise and therefore a constant step factor equal to 1 for all algorithms.

Figures 4 to 6 depict the powers of the desired- and the error signals. All curves are smoothed by a fist order IIR filter:

$$\overline{x}^2(n) = \alpha \cdot \overline{x}^2(n-1) + (1-\alpha) \cdot x^2(n),\ \alpha = 0.99. \quad (10)$$

A comparison of figures 4 and 5 indicates the comparability of convergence speed of the LFTF and the SFTF algorithm. Both algorithms behave stable for more than one hour even after the measurement time of 30 s.

The operation of the algorithms in a two channel hands free telephone system has shown that the LFTF algorithm allows a choice of initialization parameters in a wider range. The advantage of the LFTF algorithm is its robust behavior against variations of the power of the excitation signal. A proposal for the choice of initialization parameters can be found in [4].

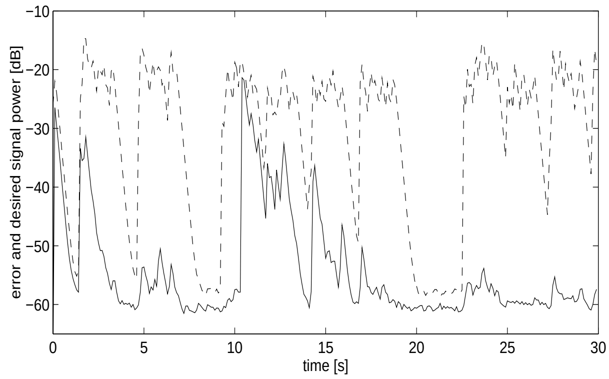

Figure 4: Convergence speed of the LFTF algorithm by Binde ($\lambda = 0.9999, \rho = 0.99$)

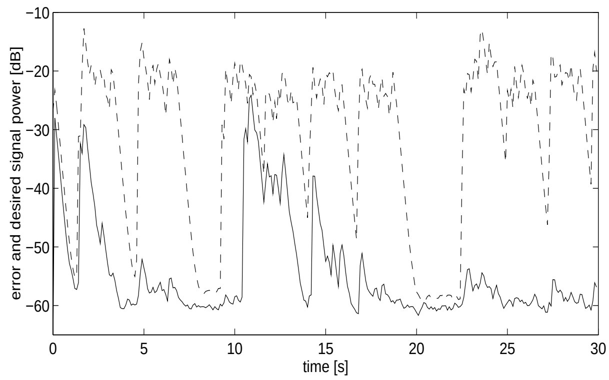

Figure 5: Convergence speed of the SFTF algorithm by Slock ($\lambda = 0.9999, \rho = 0.99$)

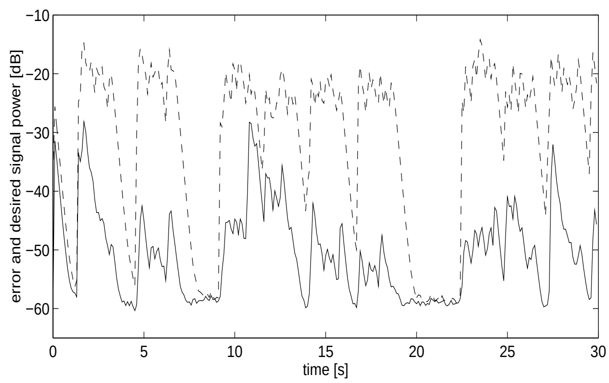

Figure 6: Convergence speed of the NLMS algorithm

Regarding the tracking behavior the FTF algorithms show no significant improvement in convergence speed compared to the NLMS algorithm in this real-time environment. But in periods of weak excitation such as speech pauses the FTF algorithms are less disturbed than the NLMS algorithm. Due to normalization of the NLMS algorithm and a smaller compensator length than LRM system length the misadjustment of the NLMS algorithm increases in speech pauses. This effect can be compensated by an appropriate step factor control, which locks adaptation during low input level periods.

3 Conclusion

In this paper the stability properties of the LFTF algorithm is examined by analyzing the eigenvalues of the linearized round off-error system. It was shown that stability can be achieved by an appropriate leakage control depending on the error energy $B_M(n)$ of the backward predictor. The theoretical results were reconsidered by simulations and real-time implementations.
Furthermore we compared the tracking ability of the LFTF algorithm with other adaptive filter algorithms. In our real-time subband environment we found no significant improvement in convergence speed of the FTF algorithms compared with the NLMS algorithm but a more robust behavior during periods of low signal to noise ratio of the excitation signal.

References

[1] S. Haykin, Adaptive Filter Theory. Second Edition, Prentice-Hall International Inc., Englewood Cliffs, N.J., 1991.

[2] A. Benallal, A. Gilloire, "A new method to stabilize fast RLS algorithms based on a first order model of the propagation of numerical errors," Proc. ICASSP 88, New York, 1988.

[3] D.T.M. Slock, T. Kailath, "Numerically stable fast transversal filters for recursive least squares adaptive filtering," IEEE Trans. on Signal Processing, Vol. 39, No. 1, Jan. 1991.

[4] T. Petillon, A. Gilloire, S. Theodoridis, "The Fast Newton Transversal Filter: An efficient scheme for acoustic echo cancelation in mobile radio," IEEE Trans. on Signal Processing, Vol. 42, No. 3, March 1994.

[5] S. Binde, "A numerically stable fast transversal filter with leakage correction," IEEE Signal Processing Letters, June 1995.

[6] S. Binde, "Implementation of a subband acoustic echo canceller," Third International Workshop on Acoustic Echo Control, September 1993, Plestin les Grèves, France, pp. 177-185.

[7] S. Binde, B. Nitsch, T. Schneider, "Application of Fast RLS algorithms for real-time acousic echo compensation", International Workshop on Acoustic Echo and Noise Control, Røros, Norway, June 1995

Prediction part

$$
\begin{aligned}
\eta_M(n) &= \underline{a}_M^H(n-1)\underline{u}_{M+1}(n) \\
f_M(n) &= \gamma_M(n-1)\eta_M(n) \\
F_M(n) &= \lambda \cdot F_M(n-1) + \gamma_M(n-1)\,|\eta_M(n)|^2 \\
\underline{\tilde{k}}_{M+1}(n) &= V_a \cdot \begin{pmatrix} 0 \\ \underline{\tilde{k}}_M(n-1) \end{pmatrix} \\
&\quad + V_a \cdot \frac{\eta_M(n)}{\lambda\, F_M(n-1)} \underline{a}_M(n-1) \\
\underline{a}_M(n) &= V_a \cdot \underline{a}_M(n-1) \\
&\quad - V_a \cdot f_M^*(n) \cdot \begin{pmatrix} 0 \\ \underline{\tilde{k}}_M(n-1) \end{pmatrix} \\
\psi_M(n) &= \lambda \cdot B_M(n-1) \cdot \tilde{k}_{M+1,M+1}(n) \\
\begin{pmatrix} \underline{\tilde{k}}_M(n) \\ 0 \end{pmatrix} &= \underline{\tilde{k}}_{M+1}(n) - V_c \cdot \tilde{k}_{M+1,M+1}(n) \cdot \underline{c}_M(n-1) \\
\gamma_M(n) &= \frac{1}{1 + \underline{\tilde{k}}_M^H(n) \cdot \underline{u}_M(n)} \\
b_M(n) &= \gamma_M(n) \cdot \psi_M(n) \\
B_M(n) &= \lambda \cdot B_M(n-1) + \psi_M(n) \cdot b_M^*(n) \\
\underline{c}_M(n) &= V_c \cdot \underline{c}_M(n-1) - V_c \cdot b_M^*(n) \cdot \begin{pmatrix} \underline{\tilde{k}}_M(n) \\ 0 \end{pmatrix}
\end{aligned}
$$

Filtering part

$$
\begin{aligned}
\alpha_M(n) &= d(n) - \underline{\hat{w}}_M^H(n-1) \cdot \underline{u}_M(n) \\
\underline{\hat{w}}_M(n) &= \underline{\hat{w}}_M(n-1) + \frac{\gamma_M(n) \cdot \alpha_M^*(n)}{1 - \rho \cdot \gamma_M(n)} \cdot \underline{\tilde{k}}_M(n)
\end{aligned}
$$

$$
\begin{aligned}
V_a &= \mathrm{diag}_{M+1}\{1, v, ..., v\} \\
V_c &= \mathrm{diag}_{M+1}\{v, ..., v, 1\} \quad , \; v < 1
\end{aligned}
$$

Table 1: Leakage FTF algorithm