8

## 8.1 Introduction

Among the large number of algorithms that solve the least-squares problem in a recursive form, the fast transversal recursive least-squares (FTRLS) algorithms are very attractive due to their reduced computational complexity [1]-[7].

The FTRLS algorithms can be derived by solving simultaneously the forward and backward linear prediction problems, along with two other transversal filters: the joint-process estimator and an auxiliary filter whose desired signal vector has one as its first and unique nonzero element (i.e., \(d(0) = 1\)). Unlike the lattice-based algorithms, the FTRLS algorithms require only time-recursive equations. However, a number of relations required to derive some of the FTRLS algorithms can be taken from the previous chapter on LRLS algorithms. The FTRLS algorithm can also be considered a fast version of an algorithm to update the transversal filter for the solution of the RLS problem, since a fixed-order update for the transversal adaptive filter coefficient vector is computed at each iteration.

The relations derived for the backward and forward prediction in the lattice-based algorithms can be used to derive the FTRLS algorithms. The resulting algorithms have computational complexity of order \(N\) making them especially attractive for practical implementation. When compared to the lattice-based algorithms, the computational complexity of the FTRLS algorithms is lower due to the absence of order-updating equations. In particular, FTRLS algorithms typically require \(7N\) to \(11N\) multiplications and divisions per output sample, as compared to \(14N\) to \(29N\) for the LRLS algorithms. Therefore, FTRLS algorithms are considered the fastest implementation solutions of the RLS problem [1]-[7].

Several alternative FTRLS algorithms have been proposed in the literature. The so-called fast Kalman algorithm [1], which is certainly one of the earlier fast transversal RLS algorithms, has computational complexity of \(11N\) multiplications and divisions per output sample. In a later stage of research development in the area of fast transversal algorithms, the fast \textit{a posteriori} error sequential technique (FAEST) [2], and the fast transversal filter (FTF) [3] algorithms were proposed, both requiring an order of \(7N\) multiplications and divisions per output sample. The FAEST and FTF algorithms have
the lowest complexity known for RLS algorithms, and are useful for problems where the input vector elements consist of delayed versions of a single input signal. Unfortunately, these algorithms are very sensitive to quantization effects and become unstable if certain actions are not taken [5]-[7], [9].

In this chapter, a particular form of the FTRLS algorithm is presented, where most of the derivations are based on those presented for the lattice algorithms. It is well known that the quantization errors in the FTRLS algorithms present exponential divergence [1]-[7]. Since the FTRLS algorithms have unstable behavior when implemented with finite-precision arithmetic, we discuss the implementation of numerically stable FTRLS algorithms, and provide the description of a particular algorithm [8]-[10].

8.2 RECURSIVE LEAST-SQUARES PREDICTION

All fast algorithms explore some structural property of the information data in order to achieve low computational complexity. In the particular case of the fast RLS algorithms discussed in this text, the reduction in the computational complexity is achieved for the cases where the input signal consists of consecutively delayed samples of the same signal. In this case, the patterns of the fast algorithms are similar in the sense that the forward and backward prediction filters are essential parts of these algorithms. The predictors perform the task of modeling the input signal, which as a result allows the replacement of matrix equations by vector and scalar relations.

In the derivation of the FTRLS algorithms, the solutions of the RLS forward and backward prediction problems are required in the time-update equations. In this section, these solutions are reviewed with emphasis on the results that are relevant to the FTRLS algorithms. As previously mentioned, we will borrow a number of derivations from the previous chapter on lattice algorithms. It is worth mentioning that the FTRLS could be introduced through an independent derivation, however the derivation based on the lattice is probably more insightful and certainly more straightforward at this point.

8.2.1 Forward Prediction Relations

The instantaneous \( a \ posteriori \) forward prediction error for an \( N \)th-order predictor is given by

\[
ε_f(k, N) = x(k) - w_f^T(k, N)x(k - 1, N)
\]

\[
= x^T(k, N + 1) \left[ \begin{array}{c} 1 \\ -w_f(k, N) \end{array} \right]
\]  

(8.1)

The relationship between \( a \ posteriori \) and \( a \ priori \) forward prediction error, first presented in equation (7.49) and repeated here for convenience, is given by

\[
e_f(k, N) = \frac{ε_f(k, N)}{γ(k - 1, N)}
\]

(8.2)