Joint Data and Channel Estimation Using The Per-Branch Processing Method

M. Javad Omidi, P. G. Gulak, *Senior Member IEEE*, and S. Pasupathy, *Fellow* IEEE
Department of Electrical and Computer Engineering
University of Toronto, 10 King's College Road, Toronto, Ontario M5S 3G4, Canada
E-mail: omidi@eecg.toronto.edu

Abstract

Joint data and channel estimation in a wireless communication system and over frequency selective Rayleigh fading channels can be performed by implementing Maximum Likelihood Sequence Estimation (MLSE) using the Per-Survivor Processing (PSP) Method. However, PSP can be used only if there is one channel estimation per symbol interval. In this paper we introduce the Per-Branch Processing (PBP) method as a general case of PSP, which has the advantages of PSP and allows more than one estimation per symbol to improve the receiver error performance in fast fading.

The Kalman filter is considered for channel estimation and the overall bit error rate (BER) performance is shown to be superior to that of detection techniques using the RLS and LMS estimators. Three different square-root methods for implementation of the Kalman filter are analyzed and compared with the RLS and LMS algorithms based on different number of bits required for implementation.

Joint data and channel estimation using PBP and PSP

For data detection over fast fading mobile radio channels, MLSE is usually implemented via the Viterbi algorithm and in the case of an unknown channel, an estimate of the channel impulse response (CIR) is required at the receiver. The state space model of signal transmission in a wireless communication system over a Rayleigh fading channel is shown in Fig. 1 [1]. This model can be described by (1) and (2)

$$x_{k+1} = Fx_k + Gw_k \tag{1}$$

$$z_k = H_k x_k + n_k \tag{2}$$

This is a linear time varying system where the state vector x_k represents the CIR. The measurement matrix H_k depends on the transmitted data sequence, and the received signal z_k can be assumed to be a noisy measurement of the state of the system, where n_k is additive Gaussian noise. F is called the state transition matrix and G is the process noise coupling matrix. The vector w_k is a zero mean white Gaussian process.

Estimation methods can be employed to estimate the channel impulse response x_k, based on the noisy received signal z_k. However, the application of any estimation method requires the measurement matrix or the vector H_k, which depends on the transmitted data sequence, and the transmitted data is not available at the receiver. This problem is sometimes called "state estimation with model uncertainty".

A solution to this problem is proposed in [2] to implement the channel estimation in the Viterbi algorithm in a PSP fashion. In this method, to overcome the problem of uncertainty in H_k, a separate estimator is required for any of the hypothesized H_k vectors on the survivor paths. In this way, each estimator uses its own hypothesized data vector for H_k and based on that, it gives an estimation of the CIR. By this method, it is guaranteed that we are using the data sequence of the shortest path for the channel estimation along the same path, which is obviously the best available information at the receiver. This method also eliminates the problem of decision delay, since the detected data associated with each survivor path is used for channel estimation on the same path immediately.

However, the PSP method is limited to one channel estimation per symbol. In [3], it was found that improvement in the receiver error performance in fast fading is obtained if the detector processes more than one sample per channel symbol, and results in a substantial

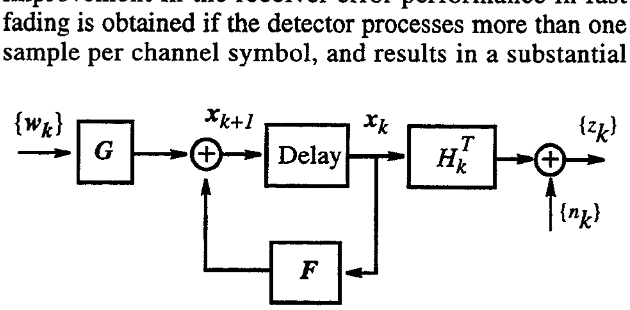

Fig. 1: Linear time varying model of signal transmission over a Rayleigh fading channel.

lowering of the error floor. Hence, to achieve a better BER result it is desirable to increase the number of samples per symbol and have several channel estimations per symbol interval. Here, we will introduce the general method of per-branch processing (PBP) for more than one channel estimation per symbol and we will show that in the special case of one sample per symbol it reduces to the PSP method.

The computational procedure of the PBP method for three samples per symbol interval is shown in Fig. 2 (a). The computation is on a trellis branch between states S_i and S_j. There are three received samples (Z) and three hypothesized impulse response vectors ($\boldsymbol{H}$) on this branch. After receiving the first received signal sample, a Branch Metric Generator (BMG) unit obtains a measure of the distance between the hypothesized value and the received value. At the same time an Estimator unit updates the channel estimation based on the received signal and the hypothesized $\boldsymbol{H}$. The new estimation will be passed to an accumulative BMG and another estimator to be processed with the second received sample and hypothesized $\boldsymbol{H}$. By processing all three samples in three stages, as shown in the figure, the branch metric is ready and after receiving the branch metrics from other branches the procedure of Add-Compare-Select can be started at node S_j to find the survivor branch. Once the survivor branch is known, the channel estimation along that branch will be retained and will be used as the output X from node S_j. Since the same routine has to be performed on all of the branches of the trellis diagram it is called PBP.

In the simple case of one sample per symbol interval we can consider Fig. 2 (b), in which only one estimator and one BMG unit are used. In this case it is possible to reduce the complexity and avoid unnecessary estimations. On all of the branches we can first compute the branch metrics and start the Add-Compare-Select procedures to find the survivor branch to each node. Then only the estimators on the surviving paths will be used to update the channel estimation for the next symbol interval. This method is PSP, where the number of estimations is reduced to the number of surviving paths which is equal to the number of states of the trellis. It should be noted that in the previous case (Fig. 2 (a)) with more than one sample per symbol, because of the data dependency it was not possible to postpone the channel estimations until the branch metrics are ready and the survivor path is known. By studying the data dependency on this diagram we can realize that only the last estimation could be deferred in this case.

Fig. 3 and Fig. 4 compare the results for the Kalman estimator and the RLS estimator for the situation of one channel estimation every symbol interval using the PSP method and the situation of three channel estimations per symbol interval using PBP, when DQPSK modulation is employed. As we can see there is a difference of about 2 dB at a BER=10^{-4} between the two methods for the Kalman estimator, which in some cases, might be tolerated to reduce the complexity of the receiver.

Any channel estimation algorithm can be utilized in the above methods. The difference in the results will be due to the tracking performance and precision of the estimators.

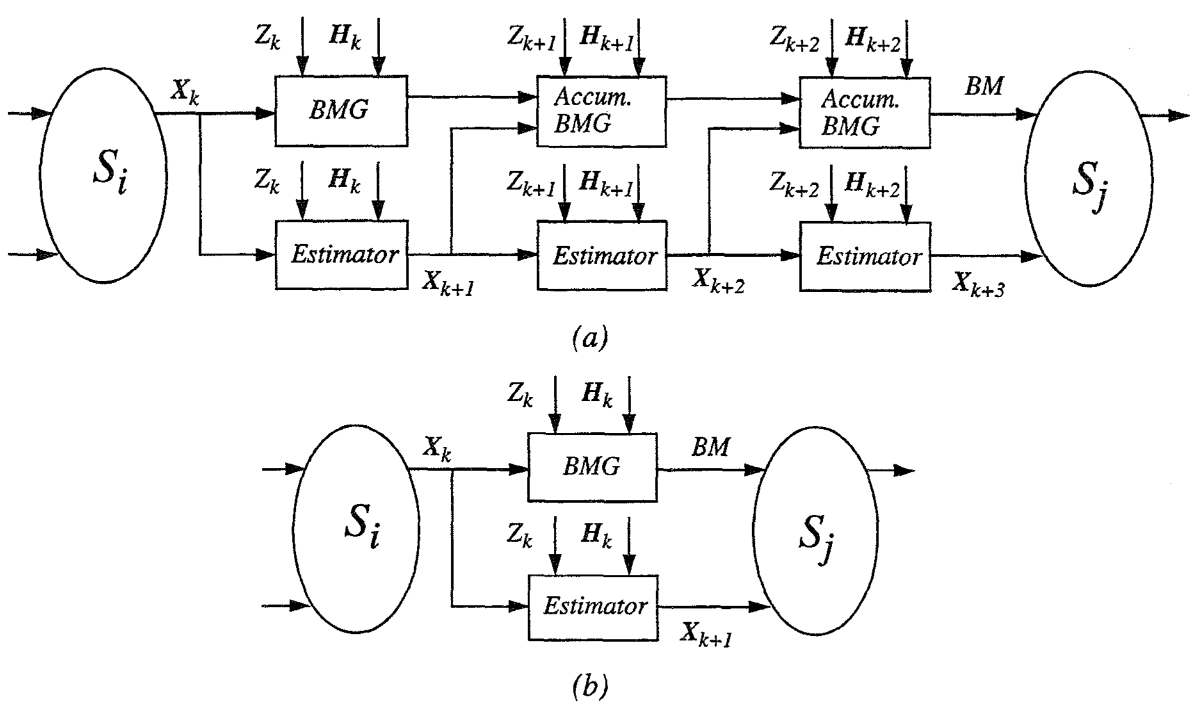

Fig. 2: (a) Data flow for PBP with three samples per symbol interval (b) Data flow for PBP with one sample per symbol interval which can be reduced to PSP.

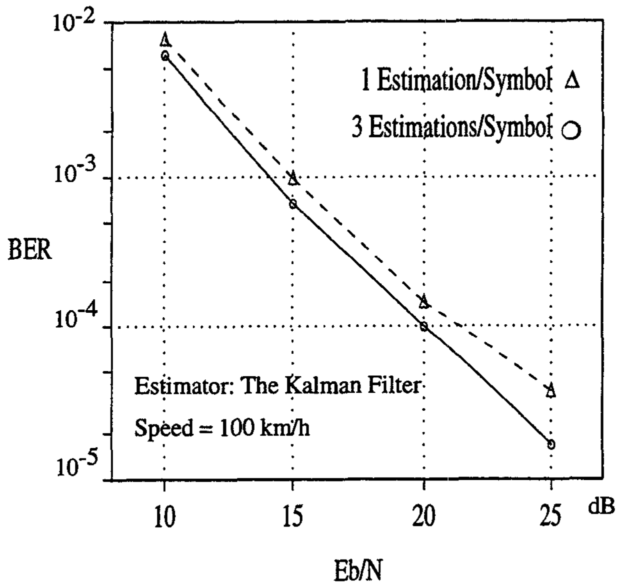

Fig. 3: The result of changing the estimation rate for the Kalman estimator.

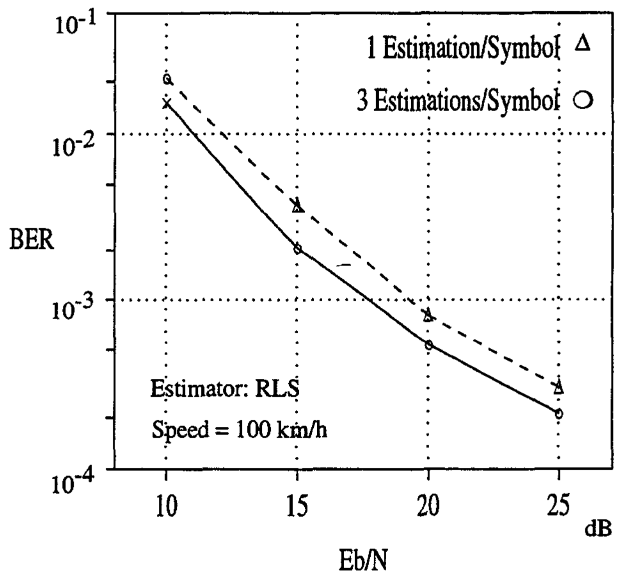

Fig. 4: The result of changing the estimation rate for the RLS estimator.

By using the channel model of [4] it is possible to employ the Kalman estimator which is the optimal estimation method and, as is shown in [4], its performance is superior to that of other estimators.

For the computation of the branch metrics, the Euclidean distance has been employed for the simulations presented in this paper. This is an accurate measure for Gaussian noise with constant variance. However, the noise variance can be different from sample to sample due to fading. The Kalman filter yields an estimation of the noise variance as one of the intermediate results in the channel estimation process. In this case, it is possible to employ the log-likelihood function with the Kalman filter, where the updated noise variance is used in the computation of the log-likelihood function. The performance of the Viterbi detector will improve by employing the log-likelihood metrics, since the receiver is not assuming a constant noise variance and an estimation of this value will be updated by each channel estimation.

Another advantage of the PBP method is that it is possible to employ the log-likelihood metrics with PBP and the Kalman filter. In this case, for computing the branch metrics the estimated noise variance is obtained from the Kalman filter, while, it is not possible to postpone the channel estimations and implement the PSP method.

Implementation of the channel estimator

The RLS and LMS algorithms are widely used in the estimation of CIR. The RLS algorithm is almost identical to the *measurement update equations* of the Kalman filter, and the Kalman filter can provide more accurate estimations of the CIR [5]. In this section we will study the effects of accuracy in channel estimation on the overall BER performance of a receiver in which joint data and channel estimation is performed by the Viterbi algorithm in a PSP or PBP fashion.

The RLS algorithm and the Kalman filter are sensitive to round-off errors, and different implementation methods may result in different numerical stabilities for these estimators. The square-root filter implementations have generally a better error performance over other conventional methods. For the implementation of the RLS algorithm and the *measurement update equations* of the Kalman filter a modified version of the square-root method proposed in [6] is adopted. This algorithm is based on working with ***LDU*** factorizations of the covariance matrices. Parallel structures are proposed in the literature for the implementation of this algorithm for the Kalman filter [6], and for the RLS algorithm [7].

The above algorithm would be sufficient for RLS but to complete the implementation of the Kalman filter an algorithm is required to compute the *time update equation*. Different computation methods could be considered for implementing the *time update equations*. These computation methods are generally equivalent and lead to the same result if carried out using full precision and sufficient number of bits per word. However, to reduce the hardware complexity it is usually desired to employ as few number of bits per word as possible. By reducing the number of bits, different implementation methods show different robustness against numerical errors. Three different computational methods for the *time update equations* are considered in this paper and the overall effects of the numerical stability on the BER performance of the PSP receiver for data communication over a wireless mobile radio channel is studied.

The first method for implementing the *time update equations* employs the weighted Gram-Schmidt orthogonalization [8], to compute the covariance update

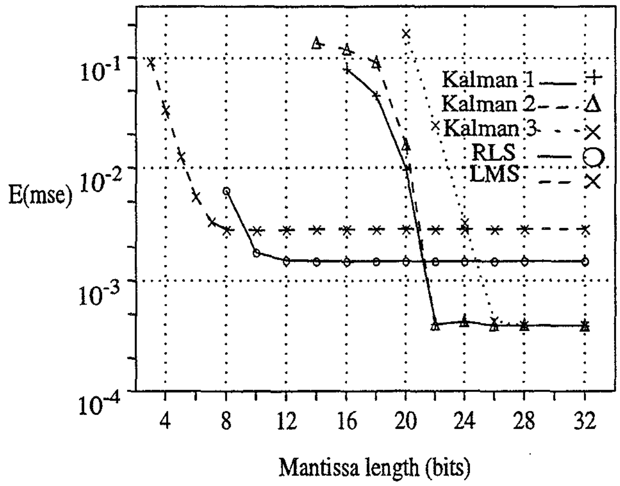

Fig. 5: The effects of changing the word length on estimation methods (Eb/N=15 dB). Kalman-1 represents the Gram-Schmidt method, Kalman-2 is for the correction method, and Kalman-3 is the direct method. PSP method is employed for detection.

equation of the Kalman filter based on a block matrix factorization. The second method is based on an L-D correction algorithm explained in [9], where it is shown to have substantial computational saving when compared with the Gram-Schmidt algorithm if the process noise covariance is time invariant. The third approach is the direct computation of the covariance update equation. The covariance matrix has to be factorized in the ***LDU*** form using a factorization algorithm presented in [10], and this method requires less computation relative to the previous methods.

In Fig. 5 the mean square error (mse) for the above Kalman estimators along with the RLS and LMS algorithms are illustrated for different number of bits per mantissa in the floating point operations. It is clear from the figure that the minimum achievable mse is lower than that of LMS and RLS algorithms, but a larger number of bits is required for the Kalman filter. Direct implementation of the Kalman filter which has the least complexity requires at least 26 bits per mantissa while two other methods require 22 bits.

For the application of joint data and channel estimation using the PSP method, the effect of reducing the number of bits on the overall BER performance is shown in Fig. 6. We see that using the Kalman channel estimator can lead to a good BER performance, while requiring a longer word length compared to LMS and RLS algorithms. Also the required mantissa length is less than what we obtain by observing the mean square error, and we can see that the methods of [8] and [9] are very close in performance and are much more efficient than the direct method.

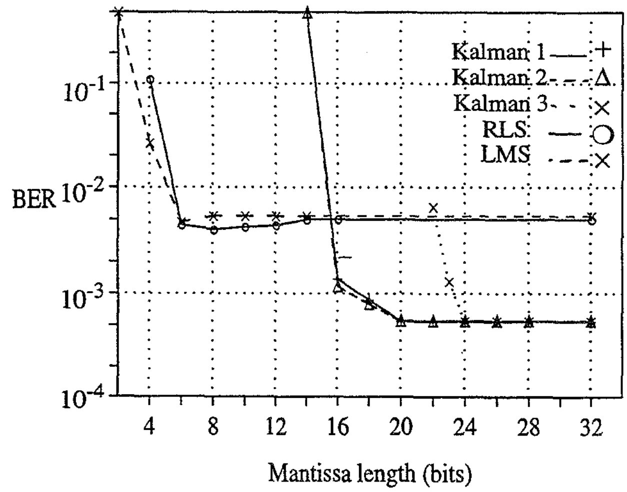

Fig. 6: The effects of changing the word length on estimation methods (Eb/N=15 dB). See Fig. 5 for details.

References

[1] Dai, Q. and Shwedyk, E., "Detection of bandlimited signals over frequency selective Rayleigh Fading channels," *IEEE Transaction on Communications*, vol. 42, pp. 941-950, Feb/Mar/Apr 1994.

[2] Raheli R., Polydoros A., Tzou C., "Per-survivor Processing: a general approach to MLSE in uncertain environments," *IEEE Transaction on Communications*, vol. 43, pp. 354-364, Feb/ Mar/Apr 1995.

[3] Vitetta G. M., Taylor D. P., "Multisampling receivers for uncoded and coded PSK signal sequences transmitted over Rayleigh frequency-flat fading channels," *IEEE Transaction on Communications*, vol. 44, pp. 130-133, Feb. 1996.

[4] Omidi M. J., Pasupathy S., Gulak P.G.,"Joint data and Kalman estimation of fading channel using a generalized Viterbi algorithm," *International Conference on Communications (ICC'96)*, pp. 1198-1203, 1996.

[5] Anderson, B. D. O., Moore, J. B., "Optimal Filtering," *Prentice-Hall*, 1979.

[6] Jover M., and Kailath T., "A parallel architecture for Kalman filter measurement update and parameter estimation," *Automatica*, vol. 22, No. 1, pp. 43-57, 1986.

[7] Omidi M. J., Gulak P.G., Pasupathy S., "Parallel structures for joint channel estimation and data detection over fading channels," *Proceedings of IEEE VLSI Signal Processing IX*, pp. 325-336, 1996.

[8] Grewal M. S., and Andrews A. P., "Kalman filtering, theory and practice," *Prentice-Hall*, 1993.

[9] Raghavan V., Pattipati K. R., and Bar-Shalom Y., "Efficient L-D factorization algorithms for PDA, IMM, and IMMPDA filters," *IEEE Transaction on Aerospace and Electronic Systems*, vol. 29, pp. 1297-1310 Oct. 1993.

[10] Golub G. H., Van Loan C. F., "Matrix computations," *Johns Hopkins University Press*, 2nd edition, pp. 137, 1991.