[8] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*. Cambridge, MA: M.I.T. Press, 1987.
[9] C. Caraiscos and B. Liu, "A roundoff error analysis of the LMS adaptive algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, no. 1, pp. 34-41, Feb. 1984.

Performance of the SMLR Deconvolution Algorithm

Chong-Yung Chi

***Abstract*—In this correspondence, we present a performance analysis regarding false alarms, correct detections, and the resolution of the well-known suboptimal maximum-likelihood deconvolution (MLD) algorithm, called the single most likely replacement (SMLR) algorithm. We assume that the source wavelet and statistical parameters are given *a priori*. We analytically show that the performance improves as the signal-to-noise ratio (SNR) increases and as the mainlobe width of the normalized autocorrelation function of the source wavelet decreases. For the same performance, a higher SNR is required as the mainlobe width of the normalized autocorrelation function increases. We also show some simulation results which are consistent with the analytic results.**

I. Introduction

Seismic deconvolution deals with the estimation of the reflectivity sequence $\mu(k)$ with noisy measurements $z(k)$, $k = 1, 2, \cdots, N$, based on the convolutional model

$$z(k) = \mu(k) * v(k) + n(k) = \sum_{i=0}^{\infty} v(i)\mu(k-i) + n(k) \quad (1)$$

where $n(k)$ is the measurement noise and $v(k)$ is the source wavelet. In the past decade, Mendel [1], Kormylo [2], Kormylo and Mendel [3] proposed a Bernoulli–Gaussian (B-G) model for the reflectivity sequence $\mu(k)$ as follows:

$$\mu(k) = r(k) \cdot q(k) \quad (2)$$

where $r(k)$ is a white Gaussian random process with zero mean and variance σ_r^2 and $q(k)$ is a Bernoulli process for which

$$P_r[q(k)] = \begin{cases} \lambda, & q(k) = 1 \\ 1 - \lambda, & q(k) = 0. \end{cases} \quad (3)$$

Kormylo and Mendel [3] developed a suboptimal maximum-likelihood deconvolution (MLD) algorithm, called the single most likely replacement (SMLR) algorithm, based on the B-G model and the assumption that $n(k)$ is white Gaussian with zero mean and variance σ_n^2. This algorithm performs well and is computationally efficient [4], [5]. Other algorithms for estimating the B-G signal $\mu(k)$ with $z(k)$, $k = 1, 2, \cdots, N$, can be found in [6]–[10].

Regarding the performance of the SMLR algorithm, we observed that for some wavelets there are fewer false alarms and more correct detections, while for some other wavelets there are many false alarms and missing detections even when the signal-to-noise ratio (SNR) is the same. These observations motivated a performance analysis.

In this correspondence, we assume that statistical parameters λ, σ_r^2, σ_n^2 and the source wavelet $v(k)$ are given *a priori* and present, in addition to SNR, which characteristics of $v(k)$ determine the performance of the SMLR algorithm. In Section II, we briefly review the background of the SMLR deconvolution algorithm for detection of $q(k)$ and estimation of $r(k)$. We then present the associated performance analysis for the SMLR algorithm in Section III. In Section IV, some simulation results are shown to support the proposed analysis. Finally, we draw some conclusions from this analysis.

II. Background of the SMLR Deconvolution Algorithm

The convolutional model (1) can be expressed in the following linear vector form:

$$z = VQr + n \quad (4)$$

where $z = (z(1), z(2), \cdots, z(N))'$, $Q = \text{diag}\,(q(1), q(2), \cdots, q(N))$, $r = (r(1), r(2), \cdots, r(N))'$, $n = (n(1), n(2), \cdots, n(N))'$ and $V = (v_1, v_2, \cdots, v_N)$ in which

$$v_k = (0, 0, \cdots, v(0), \cdots, v(N-k))'. \quad (5)$$

The covariance matrix Ω of z is given by

$$\Omega = E[zz'|q] = \sigma_r^2 VQV' + \sigma_n^2 I \quad (6)$$

where $q = (q(1), q(2), \cdots, q(N))'$ and I is an $N \times N$ identity matrix.

The SMLR algorithm is an iterative algorithm for detecting $q(k)$ based on the log-likelihood ratio

$$\ln \Lambda(k, q_r) = \ln \frac{S\{q_k|z\}}{S\{q_r|z\}} \quad (7)$$

where $S\{q|z\} = p(z|q)$, q_r is a reference sequence and q_k is a test sequence which differs from q_r only at a single time location k. The detection procedure [3] is summarized as follows:

a) Compute $\ln \Lambda(k, q_r)$ for $k = 1, 2, \cdots, N$.

b) Assume that $\ln \Lambda(k', q_r) = \max\{\ln \Lambda(k, q_r), 1 \le k \le N\}$. If $\ln \Lambda(k', q_r) > 0$, update $q_r(k')$ by $1 - q_r(k')$ and go to (a).

When $\ln \Lambda(k, q_r) \le 0$ for all $1 \le k \le N$, the detection procedure is finished. The log-likelihood ratio $\ln \Lambda(k, q_r)$ was shown [3] to be

$$\ln \Lambda(k, q_r) = \frac{1}{2} \frac{\sigma_r^2 f_k^2 (1 - 2q_r(k))}{1 + \sigma_r^2 (1 - 2q_r(k)) a_k} - \frac{1}{2} \ln [1 + \sigma_r^2 (1 - 2q_r(k)) a_k] + (1 - 2q_r(k)) \ln \left(\frac{\lambda}{1 - \lambda} \right) \quad (8)$$

where

$$f_k = v_k' \Omega_r^{-1} z \quad (9)$$

$$a_k = v_k' \Omega_r^{-1} v_k \quad (10)$$

and $\Omega_r = \Omega(q = q_r)$ (see (6)).

After detection of q, the maximum-likelihood estimate, r_{ML}, which is also equal to the minimum-variance estimate r_{MV} because r and z are jointly Gaussian when $q(k)$ is known, is given by [1],

Manuscript received July 28, 1989; revised August 11, 1990. This work was supported by the National Science Council under Grant NSC 78 0404-E-002-27.

The author is with the Department of Electrical Engineering, National Tsing Hua University, Hsinchu, Taiwan, Republic of China.

IEEE Log Number 9101331.

[10]–[12]

$$r_{ML}(k) = r_{MV}(k) = E[r(k)|z] = \sigma_r^2 q(k) f_k. \tag{11}$$

Let us define SNR and the normalized autocorrelation function $\gamma(k)$ of $v(k)$ for easy later use. SNR is defined as the ratio [1], [13] of the signal power to noise power, i.e.,

$$\text{SNR} = \frac{E[(\mu(k) * v(k))^2]}{\sigma_n^2} = \frac{\lambda \sigma_r^2 \varphi(0)}{\sigma_n^2} = \lambda F \tag{12}$$

where $F = \sigma_r^2 \varphi(0)/\sigma_n^2$ and

$$\varphi(k) = v(k) * v(-k) = \sum_{j=0}^{\infty} v(j) v(j+k) \cong \boldsymbol{v}_i' \boldsymbol{v}_{i+k} \tag{13}$$

is the autocorrelation function of $v(k)$. $\gamma(k)$ is defined as

$$\gamma(k) = \frac{\varphi(k)}{\varphi(0)}. \tag{14}$$

Note that the $\gamma(0) = 1$, $\gamma(k) = \gamma(-k)$, and $|\gamma(k)| \le 1$. Finally, we assume that $F = \text{SNR}/\lambda \gg 1$ because λ is generally very small.

III. Performance Analysis

Let $P_r(k' = k)$ denote the probability of $k' = k$ where $k \in \{1, 2, \cdots, N\}$. The performance of the SMLR algorithm can be predicted from the value of $P_r(k')$. However, the derivation of $P_r(k')$ is extremely difficult if not impossible. The analysis below is based on a heuristic approach that the mean value of $\Lambda(k', \boldsymbol{q}_r)$, $E[\ln \Lambda(k', \boldsymbol{q}_r)]$, which is then computable and is a function of both SNR and wavelet characteristics, is used to analyze the performance dependence of the SMLR algorithm on both SNR and wavelet characteristics. The larger $E[\ln \Lambda(k', \boldsymbol{q}_r)]$ indicates that the corresponding case such as a correct detection, a false alarm and removal of a false alarm occurs more likely.

The true $q(k)$ for a small λ is a sparse spike train which basically consists of isolated single spikes and pairs of two close spikes. Various aspects of performance including false alarms, correct detections and the resolution can be unravelled by considering two major cases. The first one (case I below) is that the true $\boldsymbol{q}$, denoted $\boldsymbol{q}_T$, consists of a single spike located at time point $k = k_1$, i.e., $q_T(k) = \delta(k - k_1)$. The second one (case II below) is that $q_T(k)$ consists of two spikes located at time points k_1 and k_2, i.e., $q_T(k) = \delta(k - k_1) + \delta(k - k_2)$. At each iteration the SMLR algorithm only makes a single change at $k = k'$. The local region centered at $k = k'$ can be thought of as one of these two cases when λ is small. Three specific reference sequences (I-A through I-C and II-A through II-C below) in each of the two cases are considered because they are most usual cases during the operation of the SMLR algorithm by our experience. However, even when λ is small, a $q(k)$ could also include a group of three close spikes. Although this case is not considered here, this case occurs much less likely than the two major cases and the conclusions obtained from the following analysis should still be correct for the rest part of $q(k)$. Furthermore, to perform the analysis, we need the following lemma:

Lemma 1: Assume that $\boldsymbol{q}$ in which $q(k) = 1$ only for $k = k_1, k_2, \cdots, k_L$, includes all true spikes as well as some false alarms. Then, the error variances of $r_{ML}(k_i)$, for all $1 \le i \le L$, approach zero as SNR approaches infinity.

The proof of Lemma 1 is given in Appendix A.

The three specific reference sequences considered below include $q_r(k) = 0$ (no spikes in $\boldsymbol{q}_r$) for (I-A) and (II-A), $q_r(k) = \delta(k - k_1)$ (a true spike in $\boldsymbol{q}_r$) for (I-B) and (II-B), and $q_r(k) = \delta(k - m)$, $m \ne k_1$, $m \ne k_2$ (a false alarm in $\boldsymbol{q}_r$) for (I-C) and (II-C). We now analyze what determines the performance of the SMLR algorithm for (I-A) and (II-A). The other cases can be similarly performed for completeness and thus are omitted.

Case I: $q_T(k) = \delta(k - k_1)$:

I-A: $q_r(k) = 0$, no spikes in $\boldsymbol{q}_r$:

The measurement vector z for this case can be easily seen from (4) to be

$$z = r(k_1) \boldsymbol{v}_{k_1} + \boldsymbol{n}. \tag{15}$$

One can also see, from (6), (10), and (9), that $\Omega_r = \sigma_n^2 I$, $a_k = \varphi(0)/\sigma_n^2$ and

$$f_k = \boldsymbol{v}_k' \Omega_r^{-1} z = \frac{1}{\sigma_n^2} \varphi(k - k_1) r(k_1) + \frac{1}{\sigma_n^2} \boldsymbol{v}_k' \boldsymbol{n} \tag{16}$$

from which and (8) one can show that

$$\begin{aligned} E[\ln \Lambda(k, \boldsymbol{q}_r)] &= \frac{F}{2(1+F)} [1 + F\gamma^2(k - k_1)] \\ &\quad - \frac{1}{2} \ln (1 + F) + \ln \frac{\lambda}{1 - \lambda} \\ &\cong \frac{1}{2} + \frac{1}{2} F\gamma^2(k - k_1) - \frac{1}{2} \ln F + \ln \frac{\lambda}{1 - \lambda} \\ &= A_1(k, k_1) \qquad (\text{since } F \gg 1) \end{aligned} \tag{17}$$

where we have used (17) to define $A_1(k, k_1)$. Next, we discuss the effects of $\gamma(k)$ and F on the performance based on (17).

Let us consider all possible cases about k' as follows:

(I-A-1) $k' = k_1$, a correct detection occurs;
(I-A-2) $k' \ne k_1$, a false alarm occurs.

From (17), we see that $\max \{E[\ln \Lambda(k, \boldsymbol{q}_r)]\} = E[\ln \Lambda(k_1, \boldsymbol{q}_r)]$. Let

$$\begin{aligned} \Delta(k) &= E[\ln \Lambda(k_1, \boldsymbol{q}_r)] - E[\ln \Lambda(k, \boldsymbol{q}_r)] \\ &= [1 - \gamma^2(k - k_1)] F/2. \end{aligned} \tag{18}$$

It is easy to see that $0 \le \Delta(k) \le F/2$ and that $\Delta(k)$ increases as F increases and $\gamma^2(k)$ decreases but has nothing to do with the length of $\gamma(k)$ which is about twice the length of $v(k)$. When the mainlobe of $\gamma(k)$ is narrow (i.e., $\gamma^2(k) \ll 1$ for $k \ne 0$), $\Delta(k) \approx (F/2)$ for $k \ne k_1$. However, when the mainlobe of $\gamma(k)$ is broad (i.e., $\gamma^2(k) \approx 1$, $|k| \le W$ for some W) it is then possible that $\Delta(k)$ for $|k - k_1| \le W$ is very small when F is not large enough. In other words, a false alarm associated with the true spike located at $k = k_1$ could occur near $k = k_1$ when the mainlobe of $\gamma(k)$ is broad and F is not large. Finally, $\Delta(k)$ for $k \ne k_1$ can be made arbitrarily large by increasing F or SNR. Therefore, we conclude that the performance is better for larger F and $\gamma(k)$ with a narrower mainlobe.

Case II: $q_T(k) = \delta(k - k_1) + \delta(k - k_2)$, $k_1 \ne k_2$:

II-A: $q_r(k) = 0$, no spikes in $\boldsymbol{q}_r$:

The mean of log-likelihood ratio $\ln \Lambda(k, \boldsymbol{q}_r)$ can be shown to be

$$E[\ln \Lambda(k, \boldsymbol{q}_r)] = A_1(k, k_1) + \frac{1}{2} F\gamma^2(k - k_2). \tag{19}$$

Let us consider the following situations for k':

(II-A-1) $k' = k_1$, a correction detection occurs:
(II-A-2) $k' = k_2$, a correction detection occurs;
(II-A-3) $k' \ne k_1$, $k' \ne k_2$, a false alarm occurs.

From (19) and (17), we see that $\max \{E[\ln \Lambda(k, \boldsymbol{q}_r)]\} = E[\ln \Lambda(k_1, \boldsymbol{q}_r) = E[\ln \Lambda(k_2, \boldsymbol{q}_r)]$ when the mainlobe of $\gamma(k)$ is narrow. We can infer that when the mainlobe of $\gamma(k)$ is narrow, $E[\ln \Lambda(k_1, \boldsymbol{q}_r)] = E[\ln \Lambda(k_2, \boldsymbol{q}_r)] \gg E[\ln \Lambda(k, \boldsymbol{q}_r)]$ for $k \ne k_1$ and $k \ne k_2$. How-

ever, max $\{E[\ln \Lambda(k, q_r)]\}$ could happen at some $\bar{k}$ where $k_1 < \bar{k} < k_2$ when k_1 is close to k_2 and the mainlobe of $\gamma(k)$ is broad. In other words, two close spikes could lead to a false alarm located between $k = k_1$ and $k = k_2$ when the mainlobe of $\gamma(k)$ is broad. This also implies that the resolution is better for $\gamma(k)$ with a narrow mainlobe than for $\gamma(k)$ with a broad mainlobe.

From the performed analyses we obtain the following conclusions:

(R1) The performance is better for a larger SNR and $\gamma(k)$ with a narrower mainlobe;
(R2) the resolution is better for $\gamma(k)$ with a narrower mainlobe;
(R3) the performance is not dependent on the wavelet length;
(R4) the performance can be infinitely improved by increasing SNR no matter when the mainlobe of $\gamma(k)$ is broad or narrow;
(R5) when the mainlobe of $\gamma(k)$ is broad, false alarms cannot be removed by increasing SNR, but their amplitudes tend to be smaller for a larger SNR by Lemma 1;
(R6) for the same performance, a higher SNR is required for $\gamma(k)$ with a broad mainlobe than for $\gamma(k)$ with a narrow mainlobe.

IV. Simulation Examples

In order to illustrate the analytic results presented in Section III, we selected two different wavelets $v_1(k)$ (solid line) and $v_2(k)$ (dashed line) shown in Fig. 1(a). The associated normalized autocorrelation functions $\gamma_1(k)$ (solid line) and $\gamma_2(k)$ (dashed line) are shown in Fig. 1(b) which apparently indicates the different mainlobe widths of $\gamma_1(k)$ and $\gamma_2(k)$. The synthetic data was generated with parameters $\lambda = 0.07$ and $\sigma_r^2 = 0.0225$. $q(k) = 0$ for all k was used to initialize the SMLR algorithm. The deconvolved results associated with $v_1(k)$ and $v_2(k)$ are shown in Figs. 2 and 3, respectively, where *'s denote true spikes and bars denote estimated ones.

From Fig. 2, where SNR = 10 dB, we see that the deconvolved results are very good in spite of two false alarms and five missing spikes whose amplitudes are too small to be detected since SNR is not high enough. Note that the two close spikes located at $k = 55$ and 57 were correctly detected, and the two close spikes located at $k = 262$ and 263 were also correctly detected. These results are consistent with the predicted results (R1) and (R2) since the mainlobe of $\gamma_1(k)$ is narrow.

From Fig. 3(a) where SNR = 10 dB, we see that only one spike at $k = 288$ was correctly detected. The other bars in this figure are all false alarms. As analyzed in case I-A, each false alarm is associated with an isolated spike in its vicinity. Note that the two spikes at $k = 55$ and 57 were converted into a false alarm at $k = 56$, and the two spikes at $k = 57$ and $k = 66$ led to a false alarm at $k = 61$. These observations are consistent with (R2).

Next, let us compare Fig. 2 with Fig. 3(a) where both SNR's are equal to 10 dB. Although the wavelet lengths of $v_1(k)$ and $v_2(k)$ are about the same, the mainlobe widths of $\gamma_1(k)$ and $\gamma_2(k)$ are very different. Obviously, the results shown in Fig. 2 are much better than those shown in Fig. 3(a). Again, this is consistent with (R1) and (R3).

Figs. 3(b) and (c) show the deconvolved results for SNR equal to 30 and 40 dB, respectively. From Fig. 3(b), one can see that there are more correct detections but there are still many false alarms. These results are consistent with (R5). From Fig. 3(c), again, one can see that there are still many false alarms in spite of increase of SNR. Nevertheless, false alarm amplitudes and amplitude estimation errors of detected spikes decrease as SNR increases. Again, these results are consistent with the previous conclusions (R4) and (R5).

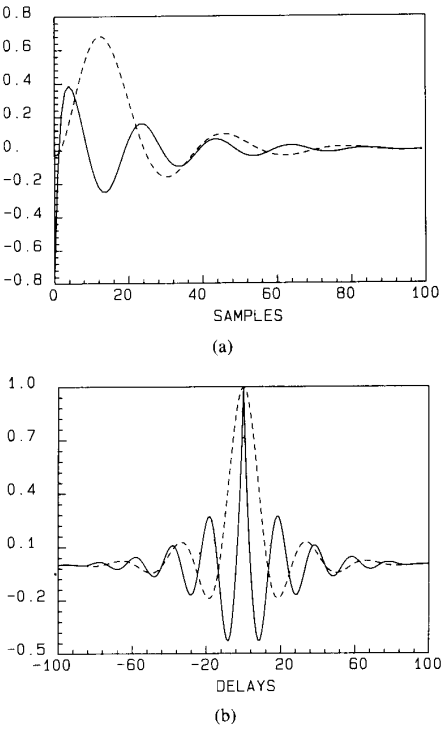

Fig. 1. (a) Wavelets $v_1(k)$ (solid line) as well as $v_2(k)$ (dashed line) and (b) the associated normalized correlation functions $\gamma_1(k)$ (solid line) as well as $\gamma_2(k)$ (dashed line).

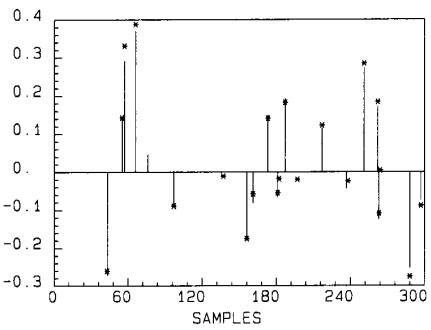

Fig. 2. Deconvolved results associated with $v_1(k)$ for SNR = 10 dB. *'s denote true spikes and bars denote estimates.

Finally, comparing Fig. 2 where SNR = 10 dB with Fig. 3(c) where SNR = 40 dB, we see that their performances are comparable but SNR's are very different. This is also consistent with our conclusion (R6).

V. Conclusions

In this correspondence, we have presented an analysis based on a heuristic approach for the performance of a well-known suboptimal iterative MLD algorithm, the SMLR algorithm, for B-G processes assuming that λ, σ_r^2, and σ_n^2 and $v(k)$ were given *a priori*. From this analysis, we obtained six main conclusions (R1) through (R6) summarized at the end of Section III with regard to the per-

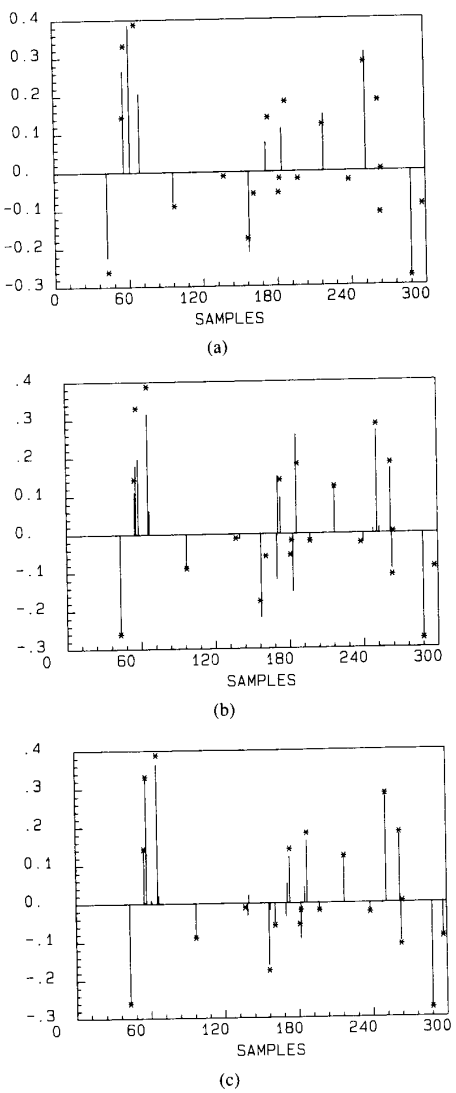

Fig. 3. Deconvolved results associated with $v_2(k)$ for (a) SNR = 10 dB, (b) SNR = 30 dB, and (c) SNR = 40 dB, respectively. *'s denote true spikes and bars denote estimates.

formance dependence of the SMLR algorithm upon both SNR and the mainlobe width of the normalized autocorrelation function $\gamma(k)$ of $v(k)$. The performance of the linear minimum-variance deconvolution (MVD) filter [13], [14] with $\mu(k)$ treated as a white noise instead of a B-G signal is determined by the bandwidth or length of $v(k)$ instead of the mainlobe width of $\gamma(k)$ on which the performance of the SMLR algorithm depends. We finally showed some simulation results to support the presented analysis.

Appendix A
Proof of Lemma 1

The estimation error of $r_{\mathrm{ML}}(k)$ for $q(k) = 1$ is defined as

$$e(k) = r(k) - r_{\mathrm{ML}}(k). \tag{A1}$$

The vector model for z given by (4) can also be expressed as

$$z = Wu + n \tag{A2}$$

where $u = (r(k_1), r(k_2), \cdots, r(k_L))'$ and $W = (v_{k_1}, v_{k_2}, \cdots, v_{k_L})$. It is well known that the maximum-likelihood estimate u is equal to the minimum-variance estimate since z and u are jointly Gaussian as follows:

$$u_{\mathrm{ML}} = u_{\mathrm{MV}} = E[u|z] = \sigma_r^2 W'(\sigma_r^2 WW' + \sigma_n^2 I)^{-1} z \tag{A3}$$

which is surely consistent with (11). It is also well known that [12]

$$\begin{aligned} E[ee'] &= \sigma_r^2 I - \sigma_r^4 W'[\sigma_r^2 WW' + \sigma_n^2 I]^{-1} W \\ &= \sigma_r^2 I - \sigma_r^4 W' \left\{ \frac{1}{\sigma_n^2} I - \frac{1}{\sigma_n^4} W \left[\frac{W'W}{\sigma_n^2} + \frac{I}{\sigma_r^2} \right]^{-1} W' \right\} W \\ &= \sigma_r^2 I - \sigma_r^4 W' \left\{ \frac{1}{\sigma_n^2} I - \frac{1}{\sigma_n^2} W [I + (\sigma_n^2/\sigma_r^2)(W'W)^{-1}]^{-1} \right. \\ &\qquad \left. \cdot (W'W)^{-1} W' \right\} W. \end{aligned} \tag{A4}$$

From (A4), we have that

$$\begin{aligned} \lim_{\sigma_n^2 \to 0} E[ee'] &= \sigma_r^2 I - \lim_{\sigma_n^2 \to 0} \sigma_r^4 W' \left\{ \frac{1}{\sigma_n^2} I - \frac{1}{\sigma_n^2} \right. \\ &\qquad \left. \cdot W [I - (\sigma_n^2/\sigma_r^2)(W'W)^{-1}] (W'W)^{-1} W' \right\} W \\ &= \sigma_r^2 I - \sigma_r^2 I = 0. \end{aligned} \tag{A5}$$

In deriving (A5), we have used the first-order Taylor series approximation. From (12), we see that SNR is inversely proportional to σ_n^2. Therefore, (A5) leads to Lemma 1.

Acknowledgment

The research described in this paper was performed at the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan.

References

[1] J. M. Mendel, *Optimal Seismic Deconvolution: An Estimation-Based Approach*. New York: Academic, 1983.
[2] J. Kormylo, "Maximum-likelihood seismic deconvolution," Ph.D. dissertation. Dep. Elec. Eng., Univ. Southern California, Los Angeles, CA, 1979.
[3] J. Kormylo and J. M. Mendel, "Maximum-likelihood detection and estimation of Bernoulli-Gaussian processes," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 482-488, 1982.
[4] J. M. Mendel, *Maximum-Likelihood Deconvolution: A Journey into Model-Based Signal Processing*. New York: Springer, 1990.
[5] C.-Y. Chi, J. M. Mendel, and D. Hampson, "A computationally fast approach to maximum-likelihood deconvolution," *Geophysics*, vol. 49, pp. 550-565, 1984.
[6] H. Kwakernaak, "Estimation of pulse heights and arrival times," *Automatica*, vol. 16, pp. 367-377, 1980.
[7] A. K. Mahalanabis, S. Prasad, and K. P. Mohandas, "Recursive decision directed estimation of reflection coefficients for seismic data deconvolution," *Automatica*, vol. 18, pp. 721-726, 1982.
[8] Y. Goussard and G. Demoment, "Recursive deconvolution of Bernoulli-Gaussian processes using a MA representation," *IEEE Trans. Geosci. Remote Sensing*, vol. 27, no. 4, pp. 384-394, July 1989.
[9] C.-Y. Chi and J. M. Mendel, "Viterbi algorithm detector for Bernoulli-Gaussian processes," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, no. 3, pp. 511-519, June 1985.
[10] C.-Y. Chi, "A fast maximum-likelihood estimation and detection algorithm for Bernoulli-Gaussian processes," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 1636-1639, 1987.
[11] J. Kormylo and J. M. Mendel, "Maximum-likelihood deconvolution," *IEEE Trans. Geosci. Remote Sensing*, vol. GE-21, pp. 72-82, 1983.
[12] J. M. Mendel, *Lessons in Digital Estimation Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[13] J. M. Mendel, "Minimum-variance deconvolution," *IEEE Trans. Geosci. Remote Sensing*, vol. GE-19, no. 3, pp. 161-171, July 1981.
[14] C.-Y. Chi and J. M. Mendel, "Performance of minimum-variance deconvolution filter," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 1145-1153, 1984.

On the Efficiency of Parallel Pipelined Architectures

Luciano da Fontoura Costa and Jan Frans Willem Slaets

***Abstract*—This correspondence describes an approach to help the design of efficient dedicated parallel pipelined architectures. Based on a previous publication where conditions for determining the most efficient mapping of digital signal processing algorithms are proposed, we develop a new approach that eliminates the restrictions and deficiencies in that paper. As an example of the presented approach, we design an efficient parallel pipelined architecture for the "butterfly" of a fast Fourier transform algorithm using operators with different execution rates.**

I. Introduction

The use of dedicated computer architectures is growing rapidly in digital signal processing (DSP) and image processing. Considerable efforts have been made to enhance the performance of these systems not only by optimizing the algorithms, such as trying to minimize arithmetic operations (e.g., [2]), but also through the design of special hardware such as dedicated parallel pipelines architectures. Several DSP (e.g., FFT, filtering, and convolution) and image processing algorithms (e.g., border detection, filtering, correlation, Hough transform), usually with regular and deterministic processing sequences, are suitable for implementation in such architectures ([1] and [3]).

This correspondence presents a methodology to help the synthesis of dedicated parallel pipelined architectures in order to optimize the utilization of the hardware resources. This subject has already been worked by Siomalas and Bowen [1], however, that paper has been prejudiced by the following deficiencies:

1) The definition of t_i (1) in Section II-C is not appropriate:

$$t_i = \left\lceil \frac{N_i}{r p_i} \right\rceil = \text{an integer.} \tag{1}$$

For example, for a given i, if $N_i = 2$ and $r = 10^6$ ops (operations per second), we have that $t_i = \lceil 2 \times 10^{-6}/p_i \rceil = 1$ for $p_i \geq 2 \times 10^{-6}$ (please refer to [1] for the meaning of t_i, N_i, r, and p_i).

A more appropriate definition is

$$t_i = \left\lceil \frac{N_i}{p_i} \right\rceil \frac{1}{r} = \text{a real number.} \tag{2}$$

2) The inadequate definition of t_i invalidates Theorems 1 and 2.

3) The proof of Theorems 1 and 2 considers only situations where all the operators have the same operation rate r.

Manuscript received February 17, 1988; revised August 6, 1990. This work was supported by FIPEC, FINEP, and FAPESP (Brazil).

L. da Fontoura Costa was with the Departamento de Física e Ciência dos Materiais, Instituto de Física e Química de São Carlos, Universidade de São Paulo, 13560 São Carlos, SP, Brazil. He is now with the Department of Electrical and Electronics Engineering, King's College, London, England.

J. F. W. Slaets is with the Departamento de Fisica e Ciência dos Materiais, Instituto de Física e Química de São Carlos, Universidade de São Paulo, 13560 São Carlos, SP, Brazil.

IEEE Log Number 9101322.

4) The condition for maximum efficiency with different operator rates in Section II-E is not correct. For example, if we apply that result for the first and second stage of pipeline from the optimal efficiency solution shown in Fig. 3 in Section VI of this paper, we have $N_1 = 4$; $N_2 = 4$; $P_{1,+} = 2$; $P_{2,*} = 4$; $r_{1,+} = 0.5 \times 10^6$ ops; $r_{2,*} = 10^6$ ops. Although we know that there is a solution (Fig. 3), the sixth expression in Section II-E of [1] does not hold:

$$\underbrace{\frac{4}{2 \times 0.5 \times 10^6}}_{\text{stage 1}} \neq \underbrace{\frac{4}{4 \times 10^6}}_{\text{stage 2}} \neq \text{an integer } t.$$

Besides these deficiencies, there are also unnecessary formalizations in [1]. We present a new approach which is simpler, more consistent, and allows a more practical and general synthesis of efficient parallel pipelined architectures.

II. Considerations and Definitions

A parallel pipelined architecture intended to execute a specific algorithm is here understood as a pipelined architecture with parallel operation within each of its stages, each stage corresponding to a level of the algorithm. The following definitions and assumptions are used henceforth:

1) An algorithm is defined as a finite number of operations partitioned into N levels.

2) A level in the algorithm is defined as the set of operations which can be started simultaneously. It is easy to verify that the levels, as defined, are determined from the interdependence of intermediate results in the algorithm.

3) An algorithm with the above characteristics can be implemented in a dedicated parallel pipelined architecture composed of N stages, each stage corresponding to one of the N levels of the algorithm. The operations in each stage are then performed by suitable operators.

4) The algorithm is continuously executed by the dedicated parallel pipelined architecture.

5) There are k_i types of operators in each level $i = 1, 2, \cdots, N$. These operators can be simple operators such as adders and multipliers or complex operators such as the "butterfly" of an FFT.

6) The efficiency is defined in a similar way to [1] in order to express the utilization of the hardware resources (operators).

III. Conditions for Maximum Efficiency

The efficiency of a parallel pipelined architecture is maximum (tends to 1) if and only if for each type j of operator ($j \in \{1, 2, \cdots, k_i\}$) within each stage i of the pipeline ($i \in \{1, 2, \cdots, N\}$) conditions (3) and (4) are both met:

$$T_{i,j} = \frac{F_{i,j}}{P_{i,j}} t_{i,j} = T \tag{3}$$

$$F_{i,j} \text{ is an integer multiple of } P_{i,j} \tag{4}$$

where

- $F_{i,j}$ number of operations of type j in level i of the algorithm,
- $T_{i,j}$ the time interval for execution of all operations of type j in stage i of the pipeline,
- $P_{i,j}$ number of operators of type j in stage i of the processor,
- $t_{i,j}$ execution time for one execution of operator of type j in stage i,
- T a real number representing the basic cycle time of the pipeline.