REDUCED-COMPLEXITY PROPORTIONATE NLMS EMPLOYING BLOCK-BASED SELECTIVE COEFFICIENT UPDATES

*James D. Gordy**, *Tyseer Aboulnasr, and Martin Bouchard*
School of Information Technology and Engineering, University of Ottawa
{jdgordy, aboulnas, bouchard}@site.uottawa.ca

ABSTRACT

This paper proposes a selective coefficient update algorithm for reducing the complexity of the proportionate normalized least-mean-square (P-NLMS) class of algorithms. It is shown that an optimal subset of coefficients to update, namely those minimizing the *a posteriori* error, cannot be constructed efficiently. A sub-optimal block-based coefficient selection algorithm is presented that combines proportional weighting of the input signal vector with fast ranking methods. It is compared to existing sub-optimal algorithms with respect to complexity overhead and convergence rate. Simulations show that the proposed algorithm produces performance approaching that of the optimal subset while maintaining a low coefficient selection overhead.

Index Terms— adaptive echo cancellation, complexity reduction, IP-NLMS, selective coefficient update.

1. INTRODUCTION

Echo in telecommunications systems is typically controlled by the use of an adaptive echo canceller, which constructs an echo estimate to be subtracted from the return-path signal. The normalized least-mean-square (NLMS) adaptation algorithm is commonly employed in echo cancellers because of its simplicity and stability [1]. However, its performance degrades in the presence of colored input signals, e.g. speech. To improve the convergence rate of NLMS, the class of proportionate (P-NLMS), improved proportionate (IP-NLMS), and gradient proportionate (GP-NLMS) algorithms was recently proposed [2] – [4]. These employ variable step sizes that distribute more update energy to higher-magnitude coefficients, increasing the rate of convergence at the expense of increased complexity. Exacerbating the problem is the increasing use of hands-free terminals and wideband telephony, where long adaptive filters are required to model echo paths in acoustic environments (up to 250 ms or more in duration), and increasing the sampling rate leads to a corresponding increase in the filter length. M-Max, periodic, and block-based coefficient updates have been proposed to reduce the complexity of NLMS [5], [6]. With respect to P-NLMS, in [7] it was proposed to divide the adaptive filter into blocks and selectively update blocks of coefficients, while [8] applies M-Max update criteria to IP-NLMS.

This paper proposes an improved selective coefficient update algorithm for IP-NLMS, and is organized as follows. IP-NLMS is reviewed in Section 2, followed in Section 3 by a derivation of an optimal coefficient selection update criteria and the proposed algorithm. Section 4 evaluates the proposed and other coefficient selection algorithms with respect to performance and complexity.

* This work was funded by an NSERC post-doctoral fellowship.

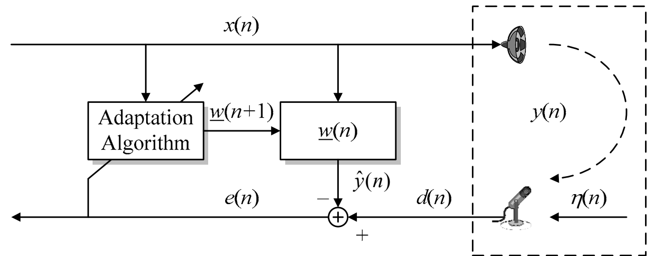

Fig. 1 – Block diagram of an adaptive acoustic echo canceller.

2. ECHO CANCELLER STRUCTURE AND PROPORTIONATE NLMS ALGORITHMS

A diagram of an echo canceller in a hands-free telephone is shown in Fig. 1. The input signal $x(n)$ is played over a loudspeaker, and the microphone signal $d(n)$ consists of echo $y(n)$ and background noise $\eta(n)$. An echo estimate is subtracted from $d(n)$ using an adaptive filter which models the echo path as a finite impulse response of length N samples:

$$e(n) = d(n) - \underline{x}^T(n)\underline{w}(n) = d(n) - \sum_{i=0}^{N-1} x(n-i)w_i(n) \qquad (1)$$

where $\underline{x}(n)$ and $\underline{w}(n)$ are the $N \times 1$ input signal and adaptive filter coefficient vectors, respectively, at time n. A control mechanism is assumed to halt adaptation during near-end speech periods.

Normalized least-mean-square (NLMS) updates the filter coefficients using the instantaneous gradient estimate normalized by the input signal power at time n [1]. Proportionate NLMS incorporates a diagonal variable-step-size matrix $\underline{G}(n)$ that is constructed proportional to the adaptive filter coefficient magnitude [2] – [4]. The IP-NLMS variant incorporates a constant factor into $\underline{G}(n)$ to improve performance in pseudo-sparse environments such as in acoustic echo cancellation [3]:

$$w_i(n+1) = w_i(n) + \frac{\mu g_i(n)x(n-i)e(n)}{\underline{x}^T(n)\underline{G}(n)\underline{x}(n) + \delta_{IPNLMS}}, \quad 0 \le i \le N-1 \qquad (2)$$

$$\underline{G}(n) = diag\{g_0(n), g_1(n), \ldots, g_{N-1}(n)\} \qquad (3)$$

$$g_i(n) = \frac{1-\beta}{2}\frac{abs\{w_i(n)\}}{\sum_{j=0}^{N-1} abs\{w_j(n)\}} + \frac{\beta}{2N}, \quad 0 \le i \le N-1 \qquad (4)$$

where $0 \le \mu \le 2$ is the overall step size, δ_{IPNLMS} is a regularization parameter, and $0 \le \beta \le 1$ controls the proportionality of $\underline{G}(n)$. NLMS requires approximately $2N$ multiplication and addition operations per sample, whereas IP-NLMS roughly doubles the complexity with at least $2N$ additional multiplications and additions per sample.

3. SELECTIVE COEFFICIENT UPDATE ALGORITHMS

3.1. Optimal Selective Coefficient Update Criteria

Assume we want to reduce the complexity of IP-NLMS by updating a subset of $M < N$ coefficients at each sample period, thus eliminating at least $N - M$ multiplication and addition operations. Following the approach in [5], we first define the *a posteriori* error $\varepsilon(n)$ as the error produced after the update of (2) is applied:

$$\varepsilon(n) = d(n) - \underline{x}^T(n)\underline{w}(n+1) = d(n) - \sum_{i=0}^{N-1} x(n-i)w_i(n+1) \ . \quad (5)$$

Substituting (2) into (5), the squared *a posteriori* error can be written as a function of the *a priori* error $e(n)$ and a sum of terms corresponding to each adaptive filter coefficient update as follows:

$$\varepsilon^2(n) = e^2(n)\left[1 - \frac{\mu\sum_{i=0}^{N-1} g_i(n)x^2(n-i)}{\underline{x}^T(n)\underline{G}(n)\underline{x}(n) + \delta_{IPNLMS}}\right]^2 . \quad (6)$$

If only a *subset* of coefficients are updated, they should be chosen such that (6) is minimized. Since the step sizes $g_i(n)$ are positive, (6) is minimized if the coefficients correspond to the M maxima of $g_i(n)x^2(n-i)$ for $0 \le i \le N-1$. Formally, let I be an ordered set of indices i_j corresponding to the weighted input samples ranked by decreasing magnitude, and let I_M be the first M indices of the set:

$$I = \left\{ i_j \;\middle|\; \begin{array}{l} g_{i_j}(n)x^2(n-i_j) \ge g_{i_{j+1}}(n)x^2(n-i_{j+1}) \\ \text{for } 0 \le i_j \le N-1 \text{ and } 1 \le j \le N \end{array} \right\} \quad (7)$$

$$I_M = I : \{i_1, i_2, \ldots, i_M\} \ . \quad (8)$$

The selective coefficient update that minimizes the *a posteriori* error is obtained by applying (2) only for the M indices within I_M:

$$w_i(n+1) = \begin{cases} w_i(n) + \Delta w_i(n), & i \in I_M \\ w_i(n), & \textit{otherwise} \end{cases} \quad (9)$$

$$\Delta w_i(n) = \frac{\mu g_i(n)x(n-i)e(n)}{\underline{x}^T(n)\underline{G}(n)\underline{x}(n) + \delta_{IPNLMS}} \ . \quad (10)$$

Each input sample is weighted by a unique time-varying step-size parameter. As such, obtaining an optimal selection using (7) – (8) precludes the use of efficient ranking algorithms such as *SORTLINE* [5]. In contrast, finding the M maxima involves calculating a sort requiring $min\{MN, N\log_2 N\}$ operations, far more than the savings from eliminating $N - M$ updates per sample. Clearly a sub-optimal approach must be employed to select coefficients. In [7] it was proposed to divide the adaptive filter into blocks, and update a subset of blocks containing maximal weighted input signal power. As shown in Section 4, that may lead to "stalling" due to infrequent updates of blocks with low filter magnitudes. In [8] the coefficients to be updated are chosen as the M maxima of the input signal vector only, ignoring $g_i(n)$.

3.2. Proposed Selective Coefficient Update Algorithm

Our goal is to combine proportionate weighting and fast ranking algorithms into the coefficient selection process while avoiding a sort of the weighted input signal vector. This is achieved by first dividing the input signal and adaptive filter coefficient vectors into $B = N / L$ equal-sized blocks of length L:

$$\underline{x}(n) = [\underline{x}_1^T(n) \quad \underline{x}_2^T(n) \quad \cdots \quad \underline{x}_B^T(n)]^T \quad (11)$$

$$\underline{w}(n) = [\underline{w}_1^T(n) \quad \underline{w}_2^T(n) \quad \cdots \quad \underline{w}_B^T(n)]^T \quad (12)$$

where $\underline{x}_k(n) = [x(n-(k-1)L) \quad \cdots \quad x(n-kL+1)]^T$ and $\underline{w}_k(n) = [w_{(k-1)L}(n) \quad \cdots \quad w_{kL-1}(n)]^T$. For each block $\underline{w}_k(n)$, $M_k \le L$ coefficients are updated at each sample period and chosen as the M_k maxima of the input signal block $\underline{x}_k(n)$. More formally, let I_k be an ordered set of indices i_j within input block k corresponding to a ranking of the L samples by decreasing magnitude. The adaptive filter coefficients are updated using (9) – (10), where the indices are obtained by selecting the first M_k indices from each set I_k so that the total number of updates is equal to M:

$$I_k = \left\{ i_j \;\middle|\; \begin{array}{l} abs\{x(n-i_j)\} \ge abs\{x(n-i_{j+1})\}, \\ \text{for } (k-1)L \le i_j \le kL+1 \text{ and } 1 \le j \le L \end{array} \right\} \quad (13)$$

$$I_M = \bigcup_{k=1}^{B} I_k : \{i_1, \ldots, i_{M_k}\} \ \text{ such that } \sum_{k=1}^{B} M_k = M \ . \quad (14)$$

During convergence periods, a logical approach is to distribute updates to blocks in proportion to the step sizes $g_i(n)$ so that more updates are applied in blocks containing a higher proportion of adaptive filter coefficient energy. A uniform distribution of updates (M / B per block) should be applied afterwards to allow more adaptation in blocks with lower adaptive filter magnitudes. In our approach, the updates per block are allocated using an integer programming approach. Let $M_k(n)$ represent the time-varying number of coefficient updates assigned to block k, and let $p_k(n)$ be the sum of proportionate step-size parameters within the block:

$$p_k(n) = \sum_{i=(k-1)L}^{kL-1} g_i(n) \ . \quad (15)$$

Allocation is performed to maximize the weighted sum of updates allocated to each block, constrained by the total number of updates and bound on the number of updates per block:

$$\max_{M_k(n)} \sum_{k=1}^{B} p_k(n)M_k(n) \ \text{ subject to} \quad (16)$$

$$\sum_{k=1}^{B} M_k(n) = M \ \text{ and } \ M_{MIN,k}(n) \le M_k(n) \le M_{MAX,k}(n) \quad (17)$$

where the bounds $M_{MIN,k}(n)$ and $M_{MAX,k}(n)$ are automatically increased for periods of high variability and decreased during steady-state periods as a function of $\sigma_k^2(n)$, an estimate of the variance of adaptive filter coefficients in each block:

$$\sigma_k^2(n) = \frac{\Delta\underline{w}_k^T(n)\Delta\underline{w}_k(n)}{\underline{w}_k^T(n)\underline{w}_k(n)} \quad (18)$$

$$\Delta\underline{w}_k(n) = \underline{w}_k(n) - \underline{w}_k(n-1) \quad (19)$$

$$M_{MIN,k}(n) = [\tfrac{M}{B} - \sigma_k^2(n)(\tfrac{M}{B} - 1)] \quad (20)$$

$$M_{MAX,k}(n) = [\tfrac{M}{B} + \sigma_k^2(n)(L - \tfrac{M}{B})] \quad (21)$$

where [·] denotes the rounding operator. Equations (20) and (21) ensure that the bounds are limited by the block size L and by M / B, a uniform distribution of updates among all blocks.

The proposed algorithm requires maintaining B ranked lists of input signal magnitudes corresponding to each block. Since the input samples are not weighted by the individual step-size parameters, the lists of (13) can be efficiently maintained using the *SORTLINE* algorithm, which in this case carries a total overhead of $B\lfloor 2\log_2(N / B)+2 \rfloor$ operations per sample [5]. A more expensive task is allocating the number of updates to each block using (15) – (21), requiring on the order of $2N + B^2$ operations. As shown in Section 4, in practice it is possible to almost completely amortize this cost by performing the allocation periodically.

4. SIMULATION RESULTS

4.1. Experimental Setup

An echo path N = 1024 samples long was obtained from a small office at f_s = 8 kHz, and is shown in Fig. 2. A coloured input signal was constructed from Gaussian noise filtered by a lowpass transfer function $H(z) = 0.5/(1–0.5z^{-1})$, with white noise added to produce a signal-to-noise ratio of 40 dB. Convergence was assessed using the system distance (error norm) in decibels [1]. Performance was assessed for NLMS, IP-NLMS, the full-complexity selective coefficient IP-NLMS of (7) – (10), denoted SC-IP-NLMS, the proposed algorithm of (13) – (21), denoted VSC-IP-NLMS, the block-update algorithm of [7], denoted BSC-IP-NLMS, and the magnitude-only algorithm of [8], denoted MSC-IP-NLMS. Common parameters of μ = 0.5 and $\delta = 10^{-4}$ were employed to achieve the same steady-state error. IP-NLMS variants used β = 0.5, and VSC-IP-NLMS re-allocated updates every 100 ms.

4.2. Results and Discussion

As M decreases the performance of selective coefficient updates degrades relative to IP-NLMS; for the given environment it was found that updating ⅜ of the coefficients (M = 384) offered a reasonable performance tradeoff compared to IP-NLMS. Results with this parameter are shown in Figs. 3 – 5. Fig. 3 shows the performance of NLMS, IP-NLMS, SC-IP-NLMS and MSC-IP-NLMS. Fig. 4 shows the performance of SC-IP-NLMS compared to the proposed VSC-IP-NLMS algorithm for B = 4, 8, and 16 blocks. Fig. 5 shows the performance of VSC-IP-NLMS (B = 16) compared to BSC-IP-NLMS with B = 32, 64, and 128 blocks. A number of observations can be made from these results. The overall performance of VSC-IP-NLMS is very close to that of full-complexity SC-IP-NLMS, and does not vary significantly as the number of blocks increases. MSC-IP-NLMS has the slowest rate of convergence, which is expected because it does not incorporate any proportional weighting into the coefficient selection. In contrast, BSC-IP-NLMS exhibits a faster initial convergence than VSC-IP-NLMS, but then "stalls". For BSC-IP-NLMS, blocks with higher adaptive filter coefficient magnitude are updated frequently even after they have converged. VSC-IP-NLMS adapts to the convergence of each block over time, avoiding a similar effect. The coefficient selection algorithms are related in terms of function and performance by appropriate choice of parameters: VSC-IP-NLMS becomes equivalent to MSC-IP-NLMS as $B \to 1$, and BSC-IP-NLMS is equivalent to SC-IP-NLMS as $B \to N$.

Table I compares coefficient selection overhead, defined as the sum of operations whose results cannot be re-used for the IP-NLMS update of (2) – (4). For simplicity it is assumed that each such operation is equivalent to a multiplication and addition operation required by (2). Within this framework, Fig. 6 shows a complexity / performance comparison of the algorithms for ⅛ and ⅜ coefficient updates (M = 128 and 384, respectively) per sample period. MSC-IP-NLMS is included as a special case of VSC-IP-NLMS for B = 1. Performance was assessed as error signal energy measured over 12.5 seconds of convergence (less error is better). Complexity was assessed with coefficient selection overhead as a percentage of selective coefficient update savings. All sub-optimal algorithms showed a decrease in performance as the overhead decreased. However, the degradation of VSC-IP-NLMS is much more gradual than BSC-IP-NLMS as overhead or M decreases. This is owing to the fact that VSC-IP-NLMS allows a more flexible allocation of updates among blocks than BSC-IP-NLMS.

TABLE I – COMPARISON OF COEFFICIENT SELECTION OVERHEAD.

Algorithm	Operations per Sample
VSC-IP-NLMS	$B\lfloor 2\log_2(N / B) + 2 \rfloor$
BSC-IP-NLMS ([7])	$min\{ B^2(M / N), B\log_2 B \}$
MSC-IP-NLMS ([8])	$\lfloor 2\log_2 N + 2 \rfloor$

5. CONCLUSIONS

A selective coefficient update algorithm was proposed for IP-NLMS, which when compared to existing algorithms showed an improvement in performance with low computational overhead. Future work could investigate applying similar techniques to IP-AP (affine projection) and GP-IP-NLMS algorithms [4].

REFERENCES

[1] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Upper Saddle River, NJ: Prentice-Hall, 1996.

[2] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancellers," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 508 – 518, Sep. 2000.

[3] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in *Proc. IEEE ICASSP*, May 2002, vol. 2, pp. 1881 – 1884.

[4] O. Hoshuyama, R. A. Goubran and A. Sugiyama, "A generalized proportionate variable step-size algorithm for fast changing acoustic environments," in *Proc. IEEE ICASSP*, May 2004, vol. 4, pp. 161 – 164.

[5] T. Aboulnasr and K. Mayyas, "Complexity reduction of the NLMS algorithm via selective coefficient update," *IEEE Trans. Signal Process.*, vol. 47, no. 5, pp. 1421 – 1424, May 1999.

[6] S. C. Douglas, "Adaptive filters employing partial updates," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 44, no. 3, pp. 209 – 216, Mar. 1997.

[7] O. Tanrikulu and K. Dogancay, "Selective-partial-update proportionate normalized least-mean-squares algorithm for network echo cancellation," in *Proc. IEEE ICASSP*, May 2002, vol. 2, pp. 1889 – 1892.

[8] H. Deng and M. Doroslovacki, "New sparse adaptive algorithms using partial update," in *Proc. IEEE ICASSP*, May 2004, vol. 2, pp. 845 – 848.

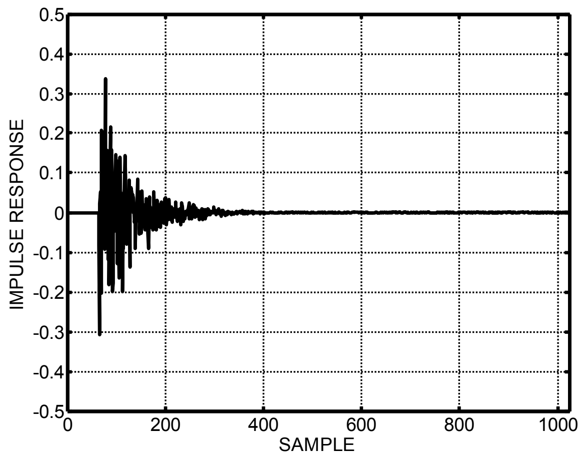

Fig. 2 – Acoustic echo path of length $N = 1024$ samples from a small office.

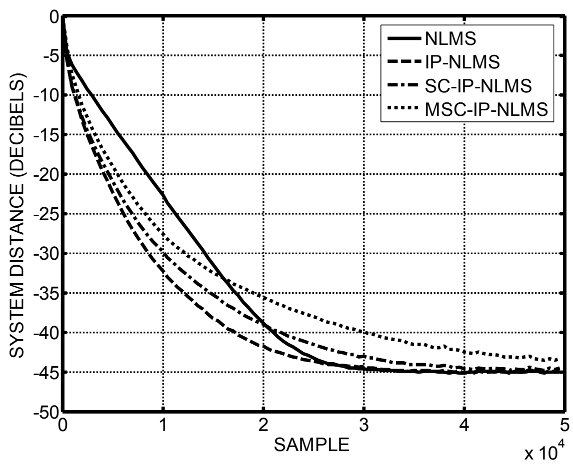

Fig. 3 – Convergence of NLMS, IP-NLMS, SC-IP-NLMS, and MSC-IP-NLMS ($M = 384$ updates per sample).

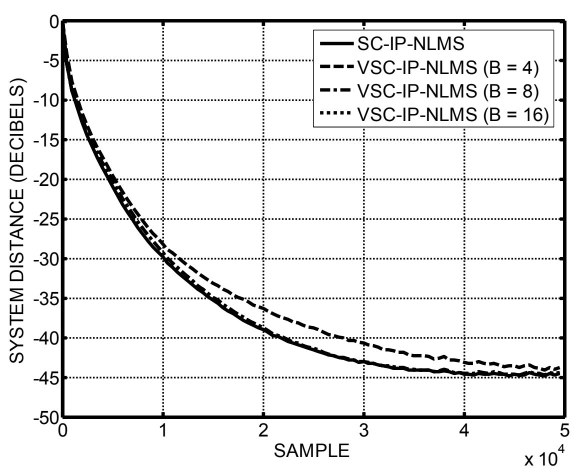

Fig. 4 – Convergence of SC-IP-NLMS and VSC-NLMS for $B = 4$, 8, and 16 blocks ($M = 384$ updates per sample).

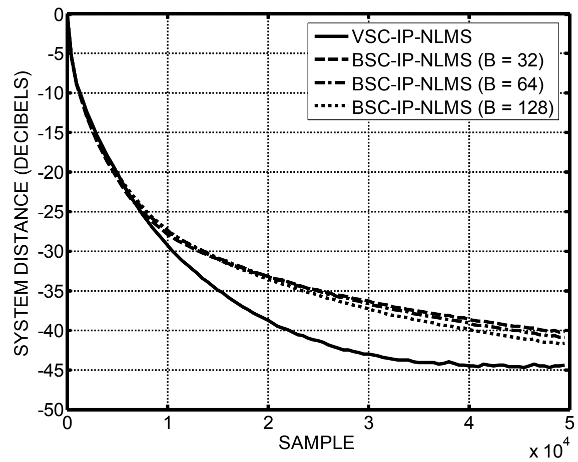

Fig. 5 – Convergence of VSC-NLMS ($B = 16$ blocks) and BSC-NLMS for $B = 32$, 64, 128 ($M = 384$ updates per sample).

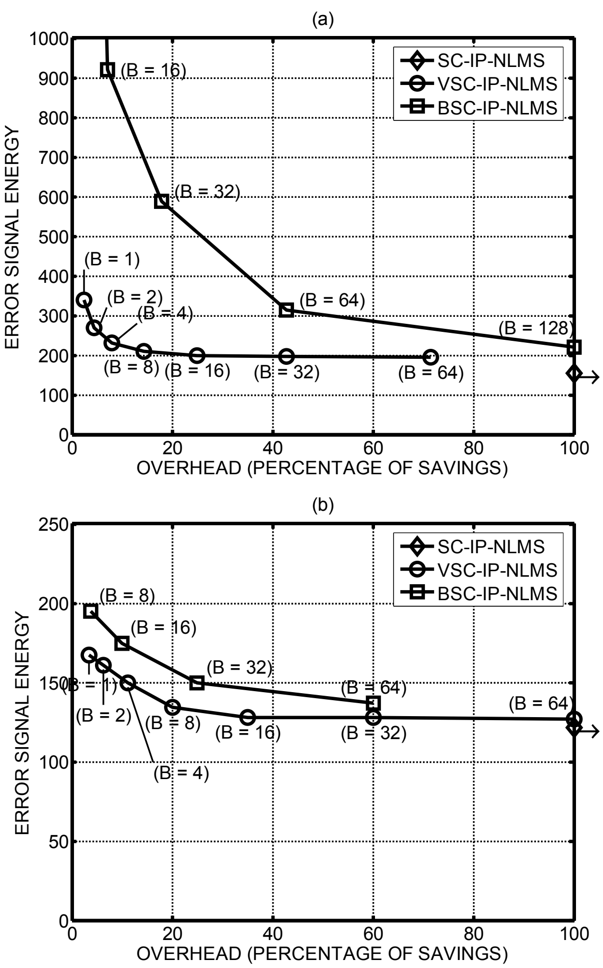

Fig. 6 – Error signal energy (performance) versus coefficient selection overhead (complexity) of SC-IP-NLMS (overhead > 100%), VSC-IP-NLMS, and BSC-IP-NLMS for (a) $M = 128$ and (b) $M = 384$ (lower energy and overhead is better).