Efficient very large-scale integration architecture design of proportionate-type least mean square adaptive filters

Gangadharaiah Soralamavu Lakshmaiah[1], Chikkajala Krishnappa Narayanappa[2], Lakshmi Shrinivasan[3], Divya Muddenahalli Narasimhaiah[4]

[1]VTU Research Centre, Department of Electronics and Communication, M. S. Ramaiah Institute of Technology,
Visvesveraya Technological University, Belagavi, India
[2]Department of Medical Electronics, M. S. Ramaiah Institute of Technology, Visvesveraya Technological University, Belagavi, India
[3]Department of Electronics and Communication, M. S. Ramaiah Institute of Technology, Visvesveraya Technological University,
Belagavi, India
[4]School of Electronics and Communication Engineering, REVA University, Bangalore, India

Article Info

ABSTRACT

The effectiveness of adaptive filters are mainly dependent on the design techniques and the algorithm of adaptation. The most common adaptation technique used is least mean square (LMS) due its computational simplicity. The application depends on the adaptive filter configuration used and are well known for system identification and real time applications. In this work, a modified delayed μ-law proportionate normalized least mean square (DMPNLMS) algorithm has been proposed. It is the improvised version of the μ-law proportionate normalized least mean square (MPNLMS) algorithm. The algorithm is realized using Ladner-Fischer type of parallel prefix logarithmic adder to reduce the silicon area. The simulation and implementation of very large-scale integration (VLSI) architecture are done using MATLAB, Vivado suite and complementary metal–oxide–semiconductor (CMOS) 90 nm technology node using Cadence RTL and Genus Compiler respectively. The DMPNLMS method exhibits a reduction in mean square error, a higher rate of convergence, and more stability. The synthesis results demonstrate that it is area and delay effective, making it practical for applications where a faster operating speed is required.

Corresponding Author:

Gangadharaiah Soralamavu Lakshmaiah
VTU Research Centre, Department of Electronics and Communication
M. S. Ramaiah Institute of Technology, Visvesveraya Technological University
Belagavi 590018, Karnataka, India
Email: gdhar75@gmail.com

1. INTRODUCTION

In the digital world there is a need for higher level of intelligence and accuracy. Digital circuits are the basic building blocks for any smart system and signal processing plays a vital role in deciding the performance of the circuits [1], [2]. In signal processing, filters are most usual circuits we can find. The need of filtering is remarkable; hence filtering is having lot of importance because of the noise presence [3].

Any noise can enter to the circuit in any means and can degrade the circuit performance. In order to make the circuit less sensitive to the noise, an efficient filter needs to be designed. The basic principle of filter is to filter any unwanted signal which should provide only the desired signal [4] which is required for circuit operation. The response of the filter with different noise is considered while evaluating a filter design. There are two types of filters, namely, finite impulse response (FIR) filters and infinite impulse response (IIR) filters.

As the name suggests, the output of FIR filter is finite and becomes zero after some time period, while for an IIR filter, the output response is infinite [4], [5].

Any filter should adapt to the change in its operating environment, the filter which can adapt to the changes in operating environment is an adaptive filter. Adaptive filters are realized either with IIR and FIR filter where the coefficients of filter can be updated in order to get the desired signal. Hence, by varying the coefficients of the FIR filters according to the change in operating condition we can make the filter to adapt to the change in operating condition [6].

The key component of the adaptive filter is an algorithm, which updates the filter coefficients iteratively with respect to the changes in environment conditions. Least mean square (LMS) is one such algorithm which is used to mimic the response of the desired filter by estimating the filter coefficients that can produce the LMS of the error signal, where error signal is the difference between desired signal and input signal with noise [7]. LMS algorithm suffers fixed step size (μ) parameter which leads to gradient noise amplification problem and it has weak convergence. To overcome these problems, normalized least mean square (NLMS) algorithm is used. NLMS algorithm offers normalized step size and modification of weight update with small positive number (ε) which makes the NLMS performance better than LMS algorithm [8]. Many other algorithms such as least mean logarithmic square (LMLS) which combines the advantages of both LMS and least mean fourth (LMF) algorithms, least logarithmic absolute difference (LLAD) algorithm which offers advantages of LMS and sign LMS (SLMS) algorithm [7], [8].

Proportionate LMS (PLMS) algorithms are introduced in order to track the sparse impulse response faster. PNLMS give better performance than NLMS with faster convergence and improved mean square error (MSE) [9]. Delayed μ-law proportionate normalized least mean square (DMPNLMS) is the proposed algorithm, which is an improvement over μ-law proportionate normalized least mean square (MPNLMS) algorithm. The remainder of the paper is organized as follows: section 2 describes the proposed architecture and the implementation of DMPNLMS algorithm, section 3 discusses the simulation results and section 4 is conclusion of the present work.

2. PROPOSED DMPNLMS ARCHITECTURE

The architecture of DMPNLMS filter is as shown in Figure 1. The input signal $u(n)$ is fed into tap coefficients with each having an arithmetic delay of 'X' units. To introduce this delay 'X', unit delay registers are used. The output of tap coefficient is fed into parallel prefix logarithmic adder. The output of the adder is multiplexed with desired signal, which contains some erroneous. This signal is fed into the desired function block for the DMPNLMS filter residues. The loopback path is formed for continuous 'n' number of iterations due to adaption [10]–[12]. Thus, the architecture of the designed DMPNLMS filter consists of tap coefficient, parallel prefix logarithmic adder, desired function and a desired block. In adaptive filtering, the tap coefficient is crucial. It represents the weights used to create the filter's output from various input values. In order to improve the effectiveness of the filter, these coefficients are modified during the learning process.
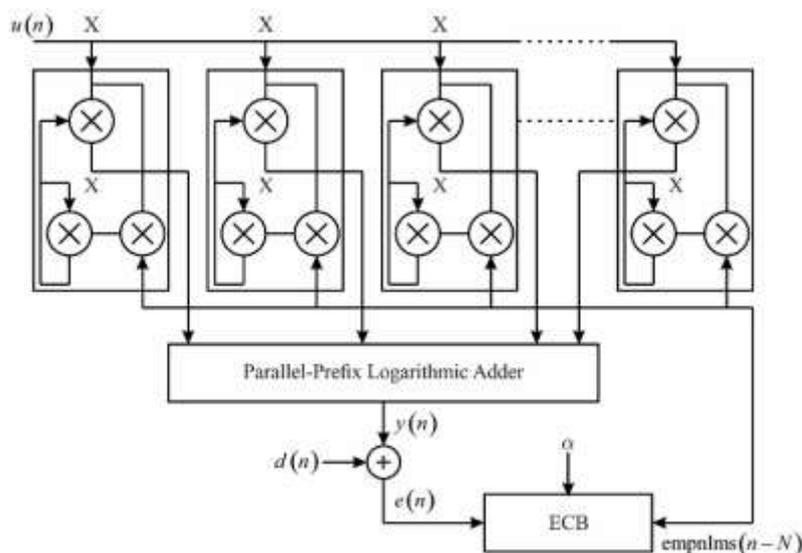
Figure 1. DMPNLMS architecture

A crucial element for effective computation inside the filter is the parallel prefix logarithmic adder. To speed up filter processes, it executes arithmetic calculations, frequently in parallel. The filter's objective is defined by the desired function. It stands in for the desired result that the filter seeks to produce [13]–[16]. Each of the subcomponents are discussed in the upcoming sub sections. The coefficient update equation of the DMPNLMS is as shown in (1) which is slightly different from NLMS with the extra step size update matrix Q as (1).

$$h(n - N + 1) = h(n - N) + \frac{Q(n-N)x(n-N+1)e(n-n+1)}{X^T(n-N+1)Q(n-N)x(n-N+1)+\delta(mpnims)} \qquad (1)$$

The diagonal matrix controls the step size and is evaluated using (2) and (3).

$$Q(n - N) = diag\{q_0(n - N), q_1(n - N), \ldots, q_{L-1}(n - 1)\} = \begin{bmatrix} q_0(n) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & q_{L-1}(n - 1) \end{bmatrix} \qquad (2)$$

The control matrix elements can be expressed as (3):

$$q_i(n) = \frac{k_i(n)}{\frac{1}{L}\sum_{i=0}^{L-1} k_i(n)} \qquad (3)$$

where,

$$k_1(n) = max\{p * F(|^\wedge h_{L-1}(n - N)|)\}, F(|h(n - N)|)$$

$$F(|h(n - N)|) = \frac{ln(1+\mu|h(n\_N)|)}{ln(1+\mu)}$$

$$(|h(n)|) < 1 \text{ and } \mu = \frac{1}{\varepsilon} \qquad (4)$$

the negative infinity at the initial stage is overcome by inserting a constant 1 in the logarithm function. The denominator function $ln(1 + \mu)$ normalizes $F(|^\wedge h1(n)|)$ in the range [0, 1]. The value of ε is a small positive number, and should be chosen such that it supports the background. $\varepsilon = 0.001$ is a good choice as the echo below -60 dB is negligible. The general design methodology used in the current work is summarized as [17]–[20]:
−   The convergence rate and stability is done using a MATLAB code simulation. This solidifies the concept for the current and previous works. Using MATLAB simulation, the algorithm is verified for the correct functionality.
−   The field programmable gate array (FPGA) synthesis is carried out using Vivado Kintex-7 to implement this digital system.
−   Application specific integrated circuit (ASIC) synthesis is also carried out with area, timing and power parameter information.

2.1. Tap coefficient
The tap coefficient is the primary block of the DMPNLMS filter. It consists of an adder which is liable for adding the input values with error control block's value, so as to boost the signal. The output of this adder is fed to an AND gate that performs "AND" function of the loop backed error control block output with the output of the adder [21]–[23]. The output of AND circuit is "OR" ed to introduce a delay. *N* number of cascaded OR gates are used to produces *N* delay unit. The input is then "AND" ed with the delayed output of the OR gate to provide tap coefficient output [24], [25].

2.2. Ladner-Fischer logarithmic adder
In the current work, Ladner-Fischer adder is being used. It consists of black cell, gray cell and AO (AND-XOR) block. The black cell is accountable for generation and propagation. The gray cell is liable for generation alone. The black cell is the combination of two AND cells and one OR cell. It gives out two outputs, one from the AND gate which is the propagation signal and the other is from the OR gate, which is the generate signal. It is the combination of AND gate and OR gate. The output is solely the generate signal. Table 1 shows the comparison between the logic levels, area, fan out and wire length of different types of parallel prefix logarithmic adders.

Table 1. Comparison of different types of parallel prefix adders

Types	Logic level	Area	Fan out	Wirelength
Kogge-Stone	$log_2 n$	$N\, log_2 n - 1 + 1$	2	$\frac{n}{2}$
Brent-Kung	$2\, log_2 n - 1$	$2n - log_2 n - 2$	2	1
Ladner-Fischer	$log_2 n + 1$	$\left(\frac{n}{4}\right) log_2 n + \frac{3n}{4-1}$	$\frac{n}{4} - 1$	1
Han-Carlson	$log_2 n$	$\left(\frac{n}{2}\right) log_2 n$	2	$\frac{n}{4}$

2.3. Desired function

The pivotal arrangement of the filter's processing sequence positions the "desired function" immediately following the parallel prefix Ladner-Fischer logarithmic adder. Its primary role revolves around conducting subtraction operations, involving the deduction of the logarithmic adder's output from the initial input signal. This subtraction process forms the bedrock of the filter's adaptation mechanism by quantifying the disparity or deviation between the expected output, as characterized by the desired function, and the present output produced by the filter.

2.4. Desired block and error control block

The desired block is used to extract error. It performs the subtraction operation of the output of desired function block and the output of the parallel prefix Ladner-Fischer logarithmic adder. This is the block where the algorithm resides. It is responsible for the formation of loop back in the system [26]–[28].

3.   RESULTS AND DISCUSSIONS

Figure 2 shows the rate of convergence for SLMS, LLAD, LMLS, quantized kernel LMS (QKLMS), and NLMS filter algorithms. From the MATLAB simulations using Ladner-Fischer adder, the MSE of SLMS, LLAD, LMLS, QKLMS and NLMS filters are 12.24 dB, -36.91 dB, -43.26 dB, -34.38 dB and -42.52 dB respectively. The filter length used is 64 and the number of iterations ran are 4000. Figure 3 shows the rate of convergence for delayed-LMS (DLMS), delayed-LLAD (DLLAD), delayed-NLMS (DNLMS) and proportionate normalized LMS (PNLMS) filter algorithms, for a filter length of 64 and 4000 iterations, from the MATLAB simulations using the Ladner-Fischer adder, the MSE of DLMS, DLLAD, DNLMS, PNLMS filter are -39.14 dB, -45.18 dB, -48.07 dB and -52.84 dB respectively.

Figure 4 shows the rate of convergence for delayed μ-law proportionate LMS (DMPLMS), DMPLLAD, DMPLMLS and DMPNLMS filter algorithms, for a filter length of 64 and 4000 iterations. From the MATLAB simulations using the Ladner-Fischer adder, the MSE of DMPLMS, DMPLLAD, DMPLMLS, DMPNLMS filters are -57.01 dB, -59.12 dB, -63.27 dB and -67.24 dB respectively and Figure 5 shows the ASIC synthesized netlist for DMPNLMS architecture [26]–[30].

Table 2 shows the improvement seen in the current work with respect to MSE when compared to the different previous works DLMS [25], DLLAD [5], DNLMS [23], PNLMS [23], DMPLMS [5], DMPLLAD [5], DMPLMLS [5], DMPNLMS [5], and Table 3 shows the delay, area and power reports for filters with length 32 bit and 64 bit. The output response of the adaptive filter $y(n)$ can be observed, the output of desired function $e(n)$ is the result of subtraction of the output $y(n)$ from desired output $d(n)$ [31]–[34].
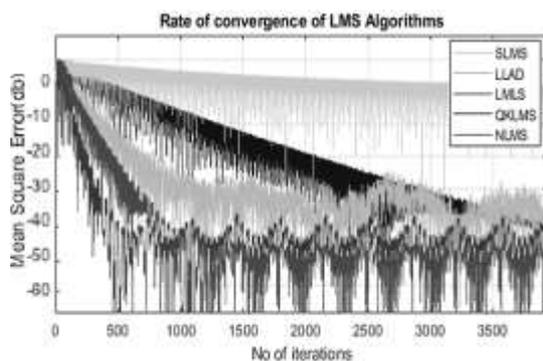
Figure 2. Rate of convergence of SLMS, LLAD, LMLS, QKLMS, and NLMS algorithms
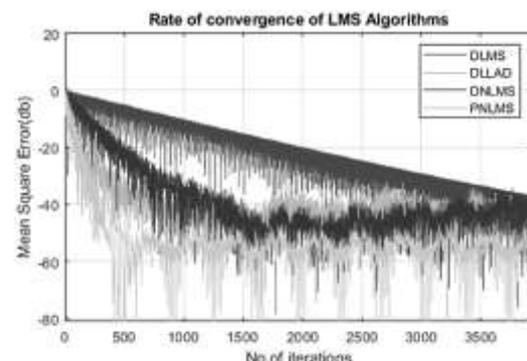
Figure 3. Rate of convergence of DLMS, DLLAD, DNLMS, and PNLMS algorithms
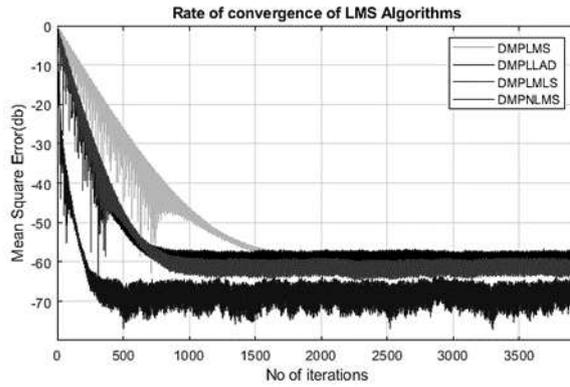
Figure 4. Rate of convergence of DMPLMS, DMPLLAD, DMPLMLS, and DMPNLMS algorithm
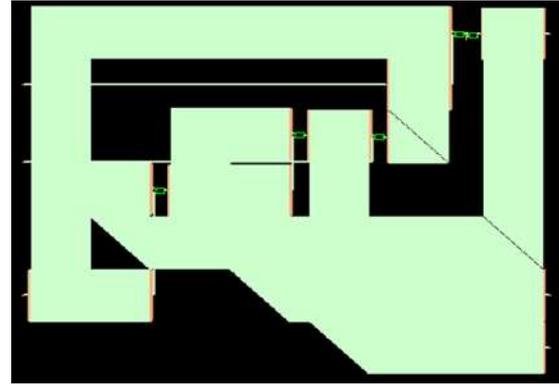
Figure 5. ASIC synthesized netlist for DMPNLMS architecture

Table 2. Improvement on MSE for different algorithms

Algorithm	MSE (dB)	Improvement (dB)
DLMS [25]	-39.14	4.94
DLLAD [5]	-45.18	8.28
DNLMS [23]	-48.07	5.57
PNLMS [23]	-52.84	9.34
DMPLMS [5]	-57.01	17.87
DMPLLAD [5]	-59.12	13.18
DMPLMLS [5]	-63.27	12.02
DMPNLMS [5]	-67.24	14.04
DMPNLMS	-67.54	14.10

Table 3. Delay, area, and power reports for different algorithms with 32-bit and 64-bit filter lengths

Design	Filter length (bits)	No. of cells	Delay (ns)	AD	Total area (μm^2)	ADP ($\mu m^{2*}ns$)	leakage power (mW)	Dynamic power (mW)	Total power (mW)
DNLMS	32	1,396	2.536	5	6,026	15,281	0.0324	0.0983	0.1307
	64	3,427	2.416	5	17,998	43,478	0.0422	0.1950	0.2372
DMPLMS	32	1,432	13.256	6	9,248	122,073	0.0534	0.1215	0.1749
	64	18,265	22.563	6	19,445	437,512	0.1739	2.1385	2.3123
DMPLLAD	32	4,526	23.985	7	8,628	582,758	0.0423	0.3225	0.3648
	64	18,924	46.731	7	109,614	5,405,820	0.1528	4.0814	4.2342
DMPLMLS	32	5,428	24.635	6	41,626	603,999	0.0432	0.9925	1.0357
	64	20,735	48.409	6	116,122	5,620,304	0.1750	4.6315	4.8065
DMPNLMS	32	5,201	20.96	7	26,268	549,001	0.0523	0.3012	0.3535
	64	17,056	38.916	7	110,462	4,296,971	0.1772	3.9056	4.0828

4. CONCLUSION

The DMPNLMS algorithm shows a improvement in MSE, convergence rate and greater stability. The synthesis results show that it is area efficient and delay efficient, hence it becomes viable for applications where higher speed of operation is required. Proportional-type adaptive algorithms offer a substantial enhancement in the convergence performance of sparse adaptive filters when compared to the traditional LMS algorithm. Nevertheless, the significant computational burden associated with these algorithms presents a formidable challenge for their implementation in VLSI. In response to this challenge, we have put forth a number of modifications aimed at simplifying the original proportionate-type normalized LMS (Pt-NLMS) algorithms. We have also introduced efficient VLSI designs tailored to these modified algorithms. Among our proposals, the DMPNLMS stands out as a robust VLSI solution. We believe that our research will serve as a catalyst for other researchers to explore more efficient hardware solutions, thus advancing the capabilities of sparse adaptive filter architectures through the use of streamlined arithmetic circuits.

REFERENCES

[1] K. Wagner, M. D. Cki, and H. Deng, "Convergence of proportionate-type LMS adaptive filters and choice of gain matrix," in *2006 Fortieth Asilomar Conference on Signals, Systems and Computers*, Oct. 2006, pp. 243–247, doi: 10.1109/ACSSC.2006.356624.

[2] K. Wagner and M. Doroslovacki, "Probability density of weight deviations given preceding weight deviations for proportionate-type LMS adaptive algorithms," *IEEE Signal Processing Letters*, vol. 18, no. 11, pp. 667–670, Nov. 2011, doi: 10.1109/LSP.2011.2168816.

[3] S. Haykin and B. Widrow, Eds., *Least-mean-square adaptive filters*. Wiley, 2003.

[4] C.-H. Lee, B. D. Rao, and H. Garudadri, "Proportionate adaptive filters based on minimizing diversity measures for promoting sparsity," in *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, Nov. 2019, pp. 769–773, doi: 10.1109/IEEECONF44664.2019.9048716.

[5] V. C. Gogineni, S. Mula, R. L. Das, and M. Chakraborty, "Performance analysis of proportionate-type LMS algorithms," in *2016 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA)*, Sep. 2016, pp. 177–181, doi: 10.1109/SPA.2016.7763608.

[6] K. T. Wagner and M. I. Doroslovacki, "Joint conditional and steady-state probability densities of weight deviations for proportionate-type LMS algorithms," in *2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, Nov. 2011, pp. 1775–1779, doi: 10.1109/ACSSC.2011.6190326.

[7] S. Ciochina, C. Paleologu, J. Benesty, and S. L. Grant, "An optimized proportionate adaptive algorithm for sparse system identification," in *2015 49th Asilomar Conference on Signals, Systems and Computers*, Nov. 2015, pp. 1546–1550, doi: 10.1109/ACSSC.2015.7421405.

[8] B. Jelfs, D. P. Mandic, and A. Cichocki, "A unifying approach to the derivation of the class of PNLMS algorithms," in *2007 15th International Conference on Digital Signal Processing*, Jul. 2007, pp. 35–38, doi: 10.1109/ICDSP.2007.4288512.

[9] S. C. Chan and Y. Zhou, "Improved generalized-proportionate stepsize LMS algorithms and performance analysis," in *2006 IEEE International Symposium on Circuits and Systems*, pp. 2325–2328, doi: 10.1109/ISCAS.2006.1693087.

[10] G. Rani and S. Kumar, "Delay and area analysis of parallel-prefix adders," *International Journal of Science and Research (IJSR)*, vol. 3, no. 6, pp. 2339–2342, 2014.

[11] H. El-Razouk, "Input-latency free versatile bit-serial GF(2 m) polynomial basis multiplication," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 30, no. 5, pp. 589–602, May 2022, doi: 10.1109/TVLSI.2022.3155611.

[12] K. R. Borisagar, B. S. Sedani, and G. R. Kulkarni, "Simulation and performance analysis of LMS and NLMS adaptive filters in non-stationary noisy environment," in *2011 International Conference on Computational Intelligence and Communication Networks*, Oct. 2011, pp. 682–686, doi: 10.1109/CICN.2011.148.

[13] M. H. Costa, L. R. Ximenes, and J. C. M. Bermudez, "Statistical analysis of the LMS adaptive algorithm subjected to a symmetric dead-zone nonlinearity at the adaptive filter output," *Signal Processing*, vol. 88, no. 6, pp. 1485–1495, Jun. 2008, doi: 10.1016/j.sigpro.2007.12.008.

[14] M. H. Costa and J. C. M. Bermudez, "A noise resilient variable step-size LMS algorithm," *Signal Processing*, vol. 88, no. 3, pp. 733–748, Mar. 2008, doi: 10.1016/j.sigpro.2007.09.015.

[15] M. Thirumoorthi, M. Heidarpur, M. Mirhassani, and M. Khalid, "An optimized M-Term Karatsuba-like binary polynomial multiplier for finite field arithmetic," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 30, no. 5, pp. 603–614, May 2022, doi: 10.1109/TVLSI.2022.3148207.

[16] O. J. Tobias, J. C. M. Bermudez, and N. J. Bershad, "Mean weight behavior of the filtered-X LMS algorithm," *IEEE Transactions on Signal Processing*, vol. 48, no. 4, pp. 1061–1075, Apr. 2000, doi: 10.1109/78.827540.

[17] M. H. Costa, J. C. M. Bermudez, and N. J. Bershad, "Stochastic analysis of the LMS algorithm with a saturation nonlinearity following the adaptive filter output," *IEEE Transactions on Signal Processing*, vol. 49, no. 7, pp. 1370–1387, Jul. 2001, doi: 10.1109/78.928691.

[18] M. H. Costa, J. C. M. Bermudez, and N. J. Bershad, "Stochastic analysis of the filtered-X LMS algorithm in systems with nonlinear secondary paths," *IEEE Transactions on Signal Processing*, vol. 50, no. 6, pp. 1327–1342, Jun. 2002, doi: 10.1109/TSP.2002.1003058.

[19] D. Alex, V. C. Gogineni, S. Mula, and S. Werner, "Novel VLSI architecture for fractional-order correntropy adaptive filtering algorithm," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 30, no. 7, pp. 893–904, Jul. 2022, doi: 10.1109/TVLSI.2022.3169010.

[20] Z. Chu, C. Shang, T. Zhang, Y. Xia, L. Wang, and W. Liu, "Efficient design of majority-logic-based approximate arithmetic circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 30, no. 12, pp. 1827–1839, Dec. 2022, doi: 10.1109/TVLSI.2022.3210252.

[21] J. Shao, W. Chen, Y. Zhang, F. Yu, and J. Chang, "Adaptive multikernel size-based maximum correntropy cubature Kalman filter for the robust state estimation," *IEEE Sensors Journal*, vol. 22, no. 20, pp. 19835–19844, Oct. 2022, doi: 10.1109/JSEN.2022.3202972.

[22] S. Mula, V. C. Gogineni, and A. S. Dhar, "Algorithm and architecture design of adaptive filters with error nonlinearities," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 9, pp. 2588–2601, Sep. 2017, doi: 10.1109/TVLSI.2017.2702171.

[23] S. Mula, V. C. Gogineni, and A. S. Dhar, "Algorithm and VLSI architecture design of proportionate-type LMS adaptive filters for sparse system identification," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 9, pp. 1750–1762, Sep. 2018, doi: 10.1109/TVLSI.2018.2828165.

[24] R. B. Shivashankar, S. L. Gangadharaiah, and C. K. Narayanappa, "FPGA based optimized LMS adaptive filter using distributed arithmetic," in *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, May 2018, pp. 1863–1867, doi: 10.1109/RTEICT42901.2018.9012288.

[25] P. K. Meher and S. Y. Park, "Critical-path analysis and low-complexity implementation of the LMS adaptive algorithm," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 3, pp. 778–788, Mar. 2014, doi: 10.1109/TCSI.2013.2284173.

[26] S. Y. Park and P. K. Meher, "Low-power, high-throughput, and low-area adaptive FIR filter based on distributed arithmetic," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 60, no. 6, pp. 346–350, Jun. 2013, doi: 10.1109/TCSII.2013.2251968.

[27] H. Jiang, L. Liu, P. P. Jonker, D. G. Elliott, F. Lombardi, and J. Han, "A high-performance and energy-efficient FIR adaptive filter using approximate distributed arithmetic circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 1, pp. 313–326, Jan. 2019, doi: 10.1109/TCSI.2018.2856513.

[28] N. J. Bershad, J. C. M. Bermudez, and J.-Y. Tourneret, "An affine combination of two LMS adaptive filters—transient mean-square analysis," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1853–1864, May 2008, doi: 10.1109/TSP.2007.911486.

[29]   S. Mula, V. C. Gogineni, and A. S. Dhar, "Robust proportionate adaptive filter architectures under impulsive noise," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 5, pp. 1223–1227, May 2019, doi: 10.1109/TVLSI.2019.2892383.

[30]   G. S. Lakshmaiah, C. K. Narayanappa, D. M. Narasimhaiah, M. Nagabhushanam, N. P. Venkatesh, and B. D. S. Shobhavathi, "Delay-power efficient VLSI architecture design for robust proportionate adaptive filter," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 26, no. 1, pp. 67–74, Apr. 2022, doi: 10.11591/ijeecs.v26.i1.pp67-74.

[31]   M. M. A. da Rosa, G. Paim, P. U. L. da Costa, E. A. C. da Costa, R. I. Soares, and S. Bampi, "AxPPA: approximate parallel prefix adders," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 1, pp. 17–28, Jan. 2023, doi: 10.1109/TVLSI.2022.3218021.

[32]   Z. Habibi and H. Zayyani, "Markovian adaptive filtering algorithm for block-sparse system identification," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 8, pp. 3032–3036, Aug. 2021, doi: 10.1109/TCSII.2021.3069879.

[33]   B. Jelfs, S. Sun, K. Ghorbani, and C. Gilliam, "An adaptive all-pass filter for time-varying delay estimation," *IEEE Signal Processing Letters*, vol. 28, pp. 628–632, 2021, doi: 10.1109/LSP.2021.3065889.

[34]   C. K. Jha, A. Nandi, and J. Mekie, "Single exact single approximate adders and single exact dual approximate adders," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 7, pp. 907–916, Jul. 2023, doi: 10.1109/TVLSI.2023.3268275.

BIOGRAPHIES OF AUTHORS

Gangadharaiah Soralamavu Lakshmaiah obtained his M.Tech. in digital electronics and advanced communication from KREC, Surathkal. Currently, he is pursuing Ph.D. in the area of VLSI signal processing. Presently he is working as assistant professor in the Department of Electronics and Communication Engineering, M. S. Ramaiah Institute of Technology, Bengaluru. His areas of interest are analog VLSI, digital VLSI, VLSI signal processing, and machine learning. He can be contacted at email: gdhar75@gmail.com.

Chikkajala Krishnappa Narayanappa received Ph.D. from VTU, Belagavi. He is currently working as associate professor at the Department of Medical Electronics, M.S. Ramaiah Institute of Technology, Bengaluru. His research interests include signal and image processing and control systems. He is the member of ISTE, IETE, and BMESI. He is also a fellow at The Institution of Engineers (India). He can be contacted at email: c_k_narayanappa@msrit.edu.

Lakshmi Shrinivasan holds a Ph.D. degree from Jain University, India in 2018. She also received her B.E. from Shivaji University, Maharashtra and M.Tech. (electronics) from VTU, Karnataka, India in 1999 and 2007, respectively. She is currently working as an associate professor in Department of Electronics and Communication at Ramaiah Institute of Technology, Bangalore. Her research interests are artificial intelligence, embedded system design, IoT and robotics. She had published several research papers in international, national conferences and indexed journals. She has 17 years of academic teaching and 2 years of industry experience. She is member of several professional bodies like MIEEE, Fellow MIETE, LMISTE, and MIAENG. She has guided several PG and UG projects. She can be contacted at email: Lakshmi.s@msrit.edu.

Divya Muddenahalli Narasimhaiah obtained M.Tech. from VTU Belgaum in 2007. Presently, she is working as assistant professor in the School of Electronics and Communication Engineering, Reva University Bengaluru. Her areas of interest are aerospace electronics, signal processing, and machine learning. She can be contacted at email: draophd@gmail.com.