Performance Analysis of Adaptive LMS Filter through Regression Analysis using SystemC

Hyeong-Geon Lee, Jae-Young Park, Suk-ki Lee, and Jong-Tae Kim

***Abstract*—**The LMS adaptive filter has several parameters which can affect their performance. From among these parameters, most papers handle the step size parameter for controlling the performance. In this paper, we approach three parameters: step-size, filter tap-size and filter form. The regression analysis is used for defining the relation between parameters and performance of LMS adaptive filter with using the system level simulation results. The results present that all parameters have performance trends in each own particular form, which can be estimated from equations drawn by regression analysis.

***Keywords*—**System level model, adaptive LMS FIR filter, regression analysis, systemC.

I. Introduction

THE LMS adaptive filter is widely used in applications such as noise filter, echo canceller, channel equalization and other DSP units because of its simplicity and low calculation cost. We can check the performance of LMS adaptive filter with two indices. First one is steady-state error which means error value when the changes of error reach steady-state. Second one is convergence speed which is the time until error attenuation reaches threshold point. In this paper, we make the model of LMS adaptive filter in system level with systemC language and measure the performance at each parameter set. We use three parameters such as step-size, filter tap-size and filter form. Values of each parameter used in this paper are seen at the Table I.

TABLE I
VALUE F PARAMETERS

Filter form	DF(direct form), TF(transposed form)
Step-size	0.005, 0.003, 0.001, 0.0008, 0.0005
Tap-size	320, 280,256,232, 200

Filter forms are two typical type of FIR filter which are used widely for digital signal processing applications. Algorithm types used in this paper is LMS algorithm and normalized version LMS algorithm which can protects the divergent of converging operation by unexpected input value. In this paper, we apply a regression algorithm mainly to LMS algorithm which can draw stable performance trends. Simulation results based on various parameter sets are adopted to using the regression analysis for finding a relationship between parameters and performances. Equations derived from regression analysis can tell us the trend of performances with change of parameters value.

Hyeong-Geon Lee is with the Department of IT Convergence, Sungkyunkwan University, Suwon city, Korea. (e-mail: dr2no3@ skku.edu).

Jae-Young Park with College of Information & Communication Engineering, Sungkyunkwan University, Suwon city, Korea. (e-mail: jyp8389@gmail.com).

Suk-ki Lee is with the Department of IT Convergence, Sungkyunkwan University, Suwon city, Korea. (e-mail:secross@skku.edu).

Jong-Tae Kim is with College of Information & Communication Engineering, Sungkyunkwan University, Suwon city, Korea. (e-mail: jtkim@ skku.edu).

II. LMS Adaptive Filter

A. Structure

LMS adaptive filter has a two canonical FIR filter structure. In this paper we use direct form and transposed form FIR filter structures, illustrated in Fig. 1 and 2, for adaptive FIR filter, where X[n] is a input signal and Y[n] is a output signal.

In direct form structure, multiplications can be done in parallel and accumulation of each product results need tap-size-1 number of adder which can make the delay of output latency. On the other hand, transposed form has a worst-case propagation path with only one adder and one multiplication which is much smaller than accumulation in direct form structure [1],[2]. This means that one adder and one multiplication in each tap is a critical path of this structure and performed simultaneously. So, it is well-known for its advantages for parallel behavior.

But there is a solution for direct form's accumulation delay, where they use a adder-tree for accumulation and delay unit for parallel implementation [3]. For using this solution, LMS algorithm should be modified because many buffers are inserted in each calculation component for parallel operation. They verify that speed-up of convergence is higher in direct form structure than in transposed form when parallel implementation is done.

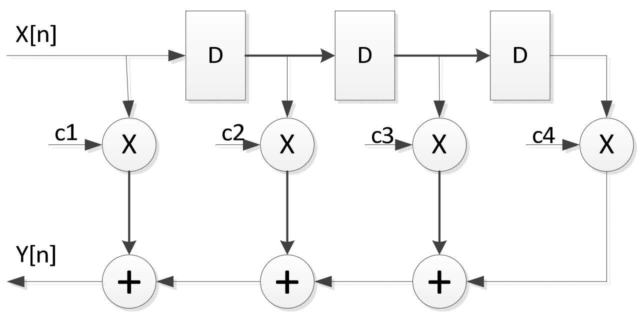

Fig. 1 Direct Form FIR Filter

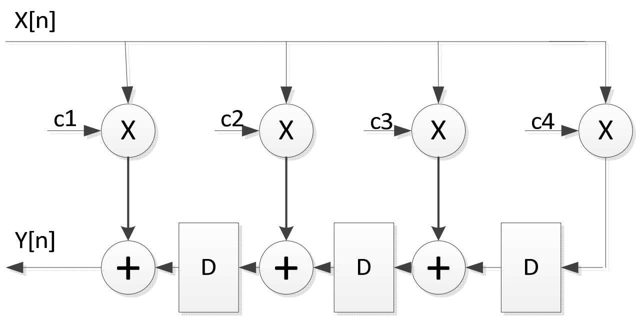

Fig. 2 Transposed Form FIR Filter

In case of echo canceller, adaptive LMS filter works to make the FIR filter follow the echo system which makes echo signal with input X[n] by changing the coefficients of filter. In these situations we suppose that the real echo system has a FIR filter form with length of echo path. For example, when we using the hands-free in the car, the length of echo path is normally 32 msec and it can be changed to 256-tap FIR filter system. The whole structure of adaptive LMS FIR filter is shown in Fig. 3, where the d[n] is a output of real echo system which is mentioned before.

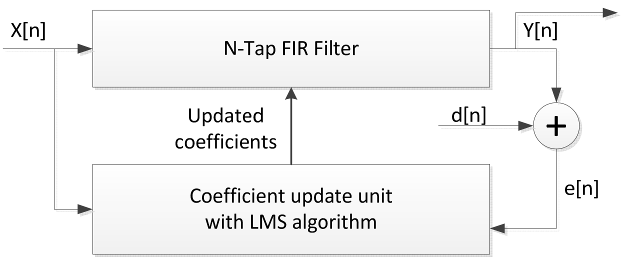

Fig. 3 Adaptive LMS filter structure

Error signal e[n], which means difference between filter output Y[n] and desired signal d[n], decrease with progress of coefficients update [4]. Goal of this filter is make the same virtual system with real, in this situation, real system is the echo system. In step of implementation, we use 256-tap FIR filter for echo system.

B. LMS Algorithm

LMS algorithm is described in (1), where u is a step-size which controls convergence speed and accuracy. In general, size of u is between 0 and 1.

$$\mathrm{w(n+1)} = \mathrm{w(n)} + ue(n)x(n) \tag{1}$$

w(n) is a coefficient of each filter tap at current cycle. w(n) and x(n) is a vector value with length of tap-size. This vector values are performed in parallel for each filter tap’s coefficients update. As shown in this equation, coefficients are updated at every cycle of adaptive filter and make the FIR filter to real echo system.

NLMS algorithm uses modified step-size, where divide it by power of input signal, defined in (2)

$$\text{modified u} = \frac{u}{X[n]\times X[n]+offset} \tag{2}$$

Offset is needed for prevent divergence of step-size when input signal become zero. Step-size of NLMS is changed by input signal at every time, so it is hard to predict the convergence speed, but it can guarantee stable work in any input signal.

Including these filter structures, we can use three parameters as shown in Table I.

III. System Level Modeling

In the abstraction layer of system design flow, we can design the adaptive LMS FIR filter with systemC language. Before make the RTL level design model, we can select optimal parameters value, in system level modeling, so that we can save the effort and time. Five cases of systemC design model are suggested in [5]. We choose model 3 among five cases to design the adaptive LMS FIR filter.

Block diagram we used in this paper is shown in Fig. 4, where input buffer and output buffer is .mat type file which can be used and made in Matlab mathematical program. Because Matlab tool is easy to make the virtual signal and analyze with graph, all signals used in this model are generated by Matlab.

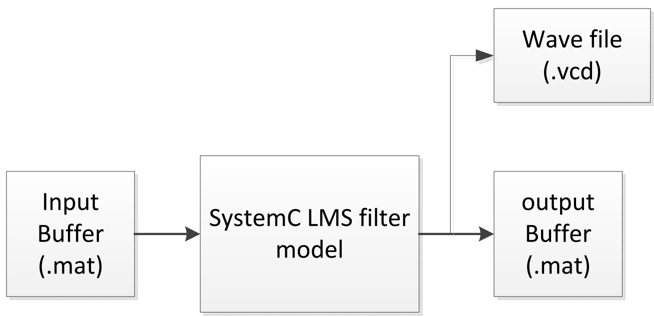

Fig. 4 System-level model

In system level model, we can design abstract layer hardware model of adaptive LMS FIR filter, which can estimate the performance with various parameter values easily. This system behavior is based on clock edge and we can check the wave form of simulation result using GTKwave program which can trace every signals, including inner signal, in GUI environment. Output files which will be saved in output buffer with mat type file are e(n) and y(n). We can analyze the performance with e(n) file in the Matlab by calculation convergence speed and steady-state error in dB level.

IV. Regression Analysis

Regression analysis is used for determining the dependency between parameters and performances with some equations. For example, general theory about step-size in LMS algorithm shows that if step-size is increased, convergence speed is also increased, but steady-state error grows worse. And other side is a contrast to this side.

But we cannot know exact trend of convergence speed and steady-state error with changing of step-size because increase means not just linear function. And there are more parameters which can have an influence to these performances. So, when we choose the parameter value before designing the RTL level model, we can regulate the equations that show the relation between parameters and performances, so we can choose best parameter values.

We control three parameters in this paper, but we consider direct-form mainly to be a filter form parameter because direct form FIR filter has better performance in both convergence speed and steady-state error than transposed form [3]. So filter form need not used in parameter in regression analysis, but we simulation transposed form LMS FIR filter just for apply the regression analysis in various situation and verify the application of regression analysis in adaptive LMS FIR filter system. In this paper, direct form FIR filter will be called case 1 and transposed form FIR filter will be called case 2. Step-size is a first parameter which will be used in parameter x and tap-size which means FIR filter order will be expressed in parameter y. There are various regression analysis equation model including quadratic model which uses main effects, iteration and square term of parameter x, y and z [6],[7]. But if we need, we can use square of 3 for more accurate regression equation model, but it will need much more time to calculate the coefficients of that equation.

V. Simulation Result

Table II shows the simulation results which will be used in extracting regression equation at case 1. Convergence speed is checked by number of iterations when the error dB value is decreased to -27dB. Steady-state error is a average of error values in dB scale when system reach the steady-state. Desired signal is made from Maltab using 256-tap FIR filter system, so we use tap-size value range from 200 to 320. And the simulation result in Table II shows that all-around performance of NLMS algorithm is worse than LMS algorithm, so we need not include the result of NLMS algorithm in the regressive equation, but just compare LMS with NLMS algorithm in the change of parameters.

TABLE II
Simulation Result with Various Parameters in Case 1

Algorithm	Tap -size (y)	Step -size (x)	Convergence speed (iteration)	Steady state error (dB)
LMS	200	0.005	550	-31.6365
LMS	256	0.003	736	-32.4507
LMS	280	0.0005	2163	-33.7220
LMS	320	0.0008	1603	-33.4970
NLMS	232	0.003	1371	-33.2948
NLMS	256	0.001	3400	-33.5093
NLMS	280	0.005	1032	-32.6718

TABLE III
Simulation Result with Various Parameters in Case 2

Algorithm	Tap -size (y)	Step -size (x)	Convergence speed (iteration)	Steady state error (dB)
LMS	200	0.005	5000	-27.7257
LMS	256	0.003	1045	-31.5192
LMS	280	0.0005	2259	-33.7384
LMS	320	0.0008	1895	-33.5015
NLMS	232	0.003	1026	-31.8495
NLMS	256	0.001	1377	-33.4295
NLMS	280	0.003	2191	-28.4905

Result of case 2 is also deducted and presented in Table III. In first parameter set in Table III, convergence speed is over 5000 iterations which mean the unstable working. Moreover, other results show worse performance than case 1 in overall parameter set. We will define 4 sets of relation with parameters and performances for regression analysis in below:

1. x, y and convergence speed.
2. x, y and steady state error.
3. y and convergence-speed, steady-state error.
4. x and convergence-speed, steady-state error.

At each set, linear model and quadratic model of regression equation will be applied. At first, in case of set 3 and set 4, quadratic model is chosen for best model which have highest accuracy. It means that the step-size increase has not just lineal relation with performance so when we choose the parameter value for more accuracy performance, using a regressive equation is needed. Equation (3) and (4) is the result of regression analysis with third and fourth set, where p means a steady-state error in (3) and convergence speed in (4). (3) is a result of simulation with 256-tap fixed filter tap-size and (4) is a result of simulation in 0.001 fixed step-size value. Both (3) and (4) use LMS algorithm only. Each equation has over 90% accuracy in the range of step-size 0.0005~0.005 and tap-size 200~320. It is credible equation when we choose the parameter value for specific performance because it has over 90% accuracy in test with various sample parameter sets. But mostly, both parameters should be considered so we need a equation which uses two parameter variables with first and second set. In the case 1 and LMS algorithm, relation with x, y and convergence speed is analyzed and equation (5) is the result of regression analysis with first set, where p means a convergence speed. Since quadratic model has a better accuracy than linear model, we choose quadratic model for best one.

Equation (6) is the result with second set, where p means a steady-state error. In this paper, we present the steady-state error with dB unit and convergence speed with number of iterations until error is declined to under the -27 dB.

$$p = 85544.4444x^2 + 199.48333x - 33.629203 \quad (3)$$

$$p = -0.0037y^2 + 2.3911y + 982.9769 \quad (4)$$

$$p = (-0.00295802)x^2 + 1.3940796y^2 + 7.01616168xy + 1.285839x + (-1232145.99)y + 2386.02426 \quad (5)$$

$$p = (-3.9281537)x^2 + 7.4295089y^2 + 3.7987806xy + 0.01577836x + (-727.735892)y - 35.226129 \quad (6)$$

In the NLMS algorithm, regression analysis results of each set have a similar trend with LMS algorithm. But as we forecasted, over all performances are dropped compared with LMS algorithm.

Fig. 5 indicates the relation between performance and two parameters by 3-D graph. As the graph signifying, convergence speed of near the 200 tap-size is faster than near the 320 tap-size. But under the 200 tap-size's convergence speed is much worse. Since desired signal is created by passing through 256 tap-size filter system in Matlab, adaptive LMS FIR filter's tap-size should be around 256 tap-size. Step-size is also limited to range from 0.0005 to 0.005 because, at out of this range, convergence speed is so slow or even don't work suitable.

Fig. 6 shows the trend of steady-state error with two parameters in the same range with Fig. 5, where equation (6) is used for calculation.

Table IV shows the accuracy of quadratic model with first set as contrasted with simulation results.

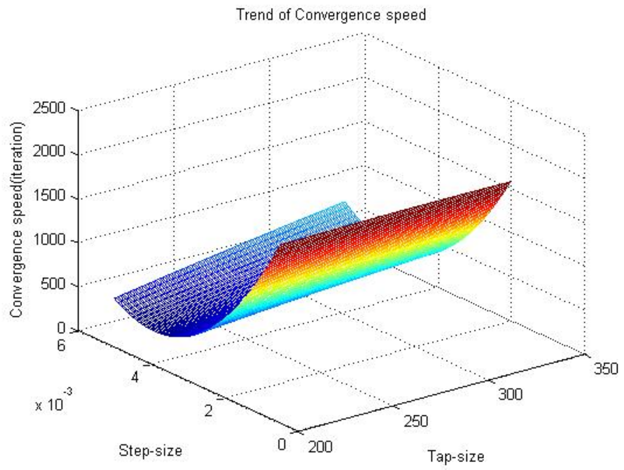

Fig. 5 3-D graph of equation (5)

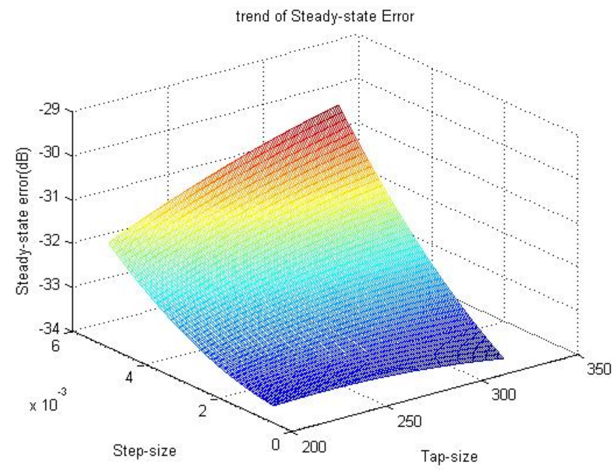

Fig. 6 3-D graph of equation (6)

TABLE IV
COMPARISON RESULT OF SIMULATION AND EQUATION (5)

Tap -size	Step -size	Simulation Result	Result of equation(5)	Accuracy (%)
210	0.005	558	586	95.2
240	0.003	711	587	82.5
270	0.002	833	989	81.3
290	0.001	1354	1620	81.1
310	0.003	771	711	92.2
250	0.002	806	966	80.1
220	0.001	1314	1587	79.2

As shown in Table III, almost the whole results of quadratic model have not over 90% accuracy in given parameter values. But it follows the trend of relation between two parameters and convergence speed. So, although we cannot predict the exact performance with each parameters value, we can define the relation between parameters and performances through quadratic equations which are drawn from regression analysis.

VI. CONCLUSION

In this paper, we design an adaptive LMS FIR filter in system-level using systemC language to adopt regression analysis at the simulation results, to define the relationship between parameters and system performance. Our experiment is done with three parameters, but especially in direct filter form and LMS algorithm in most cases because performance with these types are better than transposed-form and NLMS algorithm in all range of the rest parameters. The results of regression analysis show the trend of performances in the limited range of tap-size and step-size. This fact tells us that we cannot guess precise performance when parameters are given, but we can define the trend of performances with parameters in the limited range.

REFERENCES

[1] Douglas L, Jones, Learning Characteristics of Transposed-Form LMS Adaptive Filters , IEEE Transactions on Circuits and Systems-2: Analog and Digital Signal Processing , Vol.39, No. 10, October 1992

[2] Gan, W.S. The non-canonical LMS algorithm(NCLMS): Characteristics and analysis, Acoustics, Speech, and Signal processing, 1991. ICASSP-91.,1991 International Conference on. 2137-2140 vol.3

[3] Lok-Kee Ting, Virtex FPGA iplementation of a pipelined Adaptive LMS predictor for Electronic support Measures Receivers, IEEE Transaction On Vlsi Systemns, Vol. 13, No. 1, January 2005.

[4] Michael Huston Acoustic Echo Cancellation Using DigitalSignal Processing, The School of Information Technology and Electrical Engineering, The University of Queensland.

[5] J.R.Armstrong, Modeling With SystemC : A Case Study , Consultant to Motorola DSP Core Technology Center.

[6] Varuna Gupta, A Statistical Model for System Components Selection, IEEE International Systems Conference, 2011.

[7] Ankur Agarwal, A System-Level modeling Methodology for Performance-Driven Component Selection in Multicore Architectures, IEEE Systems Journal, Vol.6, No. 2, June 2012.