



HAL
open science

Dynamic variable step size LMS adaptation algorithms—Application to adaptive feedforward noise attenuation

Tudor-Bogdan Airimitoiaie, Bernard Vau, Dariusz Bismor, Gabriel Buche,
Ioan Doré Landau

► To cite this version:

Tudor-Bogdan Airimitoiaie, Bernard Vau, Dariusz Bismor, Gabriel Buche, Ioan Doré Landau. Dynamic variable step size LMS adaptation algorithms—Application to adaptive feedforward noise attenuation. ECC 2024 - 22nd European Control Conference, Jun 2024, Stockholm, Sweden. hal-04509676

HAL Id: hal-04509676

<https://hal.science/hal-04509676v1>

Submitted on 19 Mar 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - ShareAlike 4.0 International License

Dynamic variable step size LMS adaptation algorithms—Application to adaptive feedforward noise attenuation*

Tudor-Bogdan Airimitoiaie^a, Bernard Vau^b, Dariusz Bismor^c, Gabriel Buche^d, and Ioan Doré Landau^d

Abstract—The paper explores in detail the use of dynamic adaptation gain/step size (DAG) for improving the adaptation transient performance of variable step-size LMS (VS-LMS) adaptation algorithms. A generic form for the implementation of the DAG within the VS-LMS algorithms is provided. Criteria for the selection of the coefficients of the DAG filter which is required to be a strictly positive real transfer operator are given. The potential of the VS-LMS adaptation algorithms using a DAG is then illustrated by experimental results obtained on a relevant adaptive active noise attenuation system.

I. INTRODUCTION

The modern development of adaptation techniques in automatic control and signal processing started at the end of the fifties and beginning of the sixties (20th century). The paper [25] introduced a gradient-based adaptation algorithm in the discrete-time, later named the least mean squares (LMS). While the choice of the adaptation gain/step size for assuring the stability of the system was an open problem, interesting applications in the field of signal processing have been done. The paper [24] gives an account of the applications of the LMS algorithm up to 1975.

In automatic control, the first attempt to synthesize adaptation algorithms has been probably the paper [23], where a continuous-time formulation of a gradient type algorithm has been proposed. Unfortunately, dealing with feedback control systems to which a non-linear/time-varying loop (the adaptation loop) is added, raised crucial stability issues. The problem of the choice of the adaptation gain (step size) assuring the stability of the full system is fundamental. Discrete time adaptation algorithms assuring global asymptotic stability for any values of the adaptation gain were available since 1971 ([12], [13], [17]). The concepts of "a priori" and "a posteriori" adaptation error emerged as key points for understanding the stability issues in the discrete-time context. These algorithms, derived from stability considerations, can be interpreted in the scalar case as gradient type algorithms trying to minimize a quadratic criterion in terms of the "a posteriori" prediction error ([10], [11]).

The signal processing community has concentrated its efforts in developing variable step size/adaptation gain algorithms in a scalar context (more exactly using a diagonal matrix adaptation gain) in order to improve the performance of the LMS algorithm. An exhaustive account of the various variable step-size LMS (VS-LMS) algorithms is provided by [5], where a unified presentation is done as well as an extensive comparison of the algorithms. It turns out that the adaptation algorithms developed in control from a stability point of view can also be interpreted as "variable step size" LMS algorithms.

In using adaptive/learning recursive algorithms there is an important problem to be addressed: the compromise between alertness (with respect to environment changes—like plant or disturbance characteristics) and stationary performances when using a constant value for the adaptation gain/step size. Accelerating the adaptation transient without augmenting the value of the adaptation gain/step size is a challenging problem.

Recently, the concept of dynamic adaptation gain (DAG) has been introduced in [14], [16], [18] as means to accelerate significantly the adaptation transients without modifying the steady state (asymptotic) properties of an algorithm for a given adaptation gain/step size. The correcting term in the adaptation algorithm is first filtered before its use for the estimation of a new parameter value. With an appropriate choice of the parameters of this filter, which should be characterized by a strictly positive real (SPR) transfer function, a significant improvement of the adaptation transient is obtained. The design of this filter is well understood and design tools are available.

The main objective of this paper is to show that the DAG introduced in the context of stability based adaptation algorithms can be successfully applied to VS-LMS adaptation algorithms leading to similar significant acceleration of the adaptation transients. This will be illustrated by real-time experiments on an adaptive feedforward attenuator (a silencer).

II. INTRODUCING THE DAG-VS-LMS ALGORITHMS

The VS-LMS (variable step size least mean squares) algorithms are very popular in the field of signal processing and the field of active vibration and noise control. There is a strong similarity with some of the algorithms used in the adaptive control and recursive system identification. The VS-LMS algorithms (which are improvements of the original LMS algorithm) will be briefly reviewed in order to add the dynamic adaptation gain/step size introduced in [14].

*This work was not supported by any organization

^aTudor-Bogdan Airimitoiaie is with the Univ. Bordeaux, CNRS, Bordeaux INP, IMS, 33405 Talence, France tudor-bogdan.airimitoiaie@u-bordeaux.fr

^bBernard Vau is with Exail, 12 avenue des Coquelicots, 94385 Bonneuil sur Marne, France bernard.vau@exail.com

^cDariusz Bismor is with the Silesian University of Technology, ul. Akademicka 2A, 44-100 Gliwice, Poland dariusz.bismor@polsl.pl

^dIoan Doré Landau and Gabriel Buche are with the Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, 38000 Grenoble, France firstname.lastname@gipsa-lab.grenoble-inp.fr

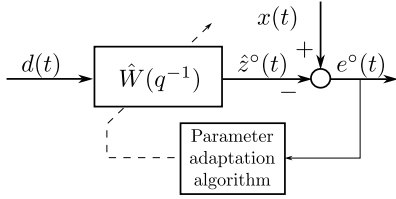


Fig. 1. Least mean squares (LMS) adaptive filtering problem.

The aim of the LMS parameter adaptation/learning algorithm (PALA) is to drive the parameters of an adjustable model in order to minimize a quadratic criterion in terms of the prediction error (difference between real data and the output of the model used for prediction).

The basic block diagram illustrating the LMS algorithm's operation is shown in Fig. 1. The adaptive filter $W(q^{-1})$ is fed with the input sequence $d(t)$. The output of the filter, $z(t)$, is compared with the desired signal, $x(t)$, to compute the error signal $e(t)$. The LMS algorithm adjusts the weights of the $W(q^{-1})$ filter to minimize the error.

Consider that the desired signal can be described by:

$$x(t) = \theta^T \phi(t), \quad (1)$$

where the *parameter vector* and the *measurement vector* are denoted by

$$\theta^T = [w_0, w_1, \dots, w_{n_W}], \quad \text{and} \quad (2)$$

$$\phi^T(t) = [d(t), d(t-1), \dots, d(t-n_W)], \quad (3)$$

respectively. The adjustable prediction model of the adaptive filter will be described by:

$$\hat{z}^o(t) = \hat{\theta}^T(t-1)\phi(t), \quad (4)$$

where $\hat{z}^o(t)$ is termed the *a priori* predicted output depending upon the values of the estimated parameter vector θ at instant $t-1$:

$$\hat{\theta}^T(t-1) = [\hat{w}_0(t-1), \hat{w}_1(t-1), \dots, \hat{w}_{n_W}(t-1)]. \quad (5)$$

It is very useful to consider also the *a posteriori* predicted output computed on the basis of the new estimated parameter vector at t , $\hat{\theta}(t)$, which will be available somewhere between t and $t+1$. The *a posteriori* predicted output will be given by:

$$\hat{z}(t) = \hat{\theta}^T(t)\phi(t) \quad (6)$$

One defines an *a priori* prediction error as:

$$e^o(t) = x(t) - \hat{z}^o(t) = [\theta - \hat{\theta}(t-1)]^T \phi(t) \quad (7)$$

and an *a posteriori* prediction error as:

$$e(t) = x(t) - \hat{z}(t) = [\theta - \hat{\theta}(t)]^T \phi(t). \quad (8)$$

The VS-LMS algorithms update the filter taps according to the formula (see [24]):

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \mu(t)\phi(t)e^o(t), \quad (9)$$

where $\mu(t)$ is the variable step-size parameter. In the standard form of the LMS algorithm, the step-size has a constant value $\mu(t) = \mu$. Large values of μ allow for fast adaptation, but also give large excess mean square error (EMSE, see [5]). Too large step-sizes may lead to the loss of stability (see [9] for a discussion). On the other hand, too small step-sizes give slow convergence, which in many practical applications is not desirable. A variable step-size $\mu(t)$ can provide a compromise. The first VS-LMS algorithm was the Normalized LMS (NLMS) algorithm proposed in 1967 independently by [3], [20], which uses the following equation for the step-size:

$$\mu(t) = \frac{\mu}{\delta + \phi(t)^T \phi(t)}, \quad (10)$$

where δ is a very small scalar value used in order to avoid division by zero (typically of the order of 10^{-16}).¹

In adaptive control, where the stability of the full adaptive control system is considered as a fundamental issue, VS-LMS algorithms have been developed from a stability point of view. However, when scalar type adaptation gain is used (i.e. diagonal matrix adaptation gain), these algorithms can be interpreted as gradient type algorithms for the minimization of a quadratic error in terms of the *a posteriori* prediction error defined in (8). See [9] for details. See also [18] for a derivation of this type of algorithm and its application to active noise control. This algorithm will be termed PLMS (to distinguish it with respect to the LMS and the other VS-LMS which are using the *a priori* prediction error). The algorithm has the form:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \mu\phi(t)e(t) \quad (11)$$

$$= \hat{\theta}(t-1) + \mu(t)\phi(t)e^o(t), \quad (12)$$

where:

$$\mu(t) = \frac{\mu}{1 + \mu\phi(t)^T \phi(t)}, \quad (13)$$

See Appendix I for a direct derivation of the algorithm.

When using the *dynamic adaptation gain/learning rate* (DAG) equation (9) of the VS-LMS algorithms will take the form:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \frac{C(q^{-1})}{D'(q^{-1})} [\mu(t)\phi(t)e^o(t)] \quad (14)$$

where² $\frac{C(q^{-1})}{D'(q^{-1})}$ is termed the “dynamic adaptation gain/learning rate” (DAG) and has the form:

$$H_{DAG}(q^{-1}) = \frac{C(q^{-1})}{D'(q^{-1})} = \frac{1 + c_1 q^{-1} + \dots + c_{n_C} q^{-n_C}}{1 - d'_1 q^{-1} - \dots - d'_{n_{D'}} q^{-n_{D'}}} \quad (15)$$

¹The case of NLMS using larger values of δ (for example $\delta = 1$) is discussed in [7].

²The complex variable z^{-1} will be used for characterizing the system's behaviour in the frequency domain and the delay operator q^{-1} will be used for describing the system's behavior in the time domain.

The effective implementation of the algorithm given in (14) leads to:

$$\begin{aligned} \hat{\theta}(t) = & d_1 \hat{\theta}(t-1) + d_2 \hat{\theta}(t-2) + \dots + d_{n_D} \hat{\theta}(t-n_D) \\ & + \mu(t) \phi(t) e^\circ(t) + c_1 \mu(t-1) \phi(t-1) e^\circ(t-1) + \\ & + \dots + c_{n_C} \mu(t-n_C) \phi(t-n_C) e^\circ(t-n_C) \end{aligned} \quad (16)$$

where ($n_D = n_{D'} + 1$):

$$d_i = (d'_i - d'_{i-1}) \quad ; i = 1, \dots, n_D; d'_0 = -1, d'_{n_D} = 0 \quad (17)$$

To implement the algorithm, one needs a computable expression for $e^\circ(n)$. This is obtained by computing $\hat{z}^\circ(n)$ in (7) as:

$$\hat{z}^\circ(t) = \hat{\theta}_0^T(t-1) \phi(t), \quad (18)$$

where

$$\begin{aligned} \hat{\theta}_0^T(t-1) = & d_1 \hat{\theta}(t-1) + d_2 \hat{\theta}(t-2) + \dots + d_{n_D} \hat{\theta}(t-n_D) \\ & + c_1 \mu(t-1) \phi(t-1) e^\circ(t-1) + \\ & + \dots + c_{n_C} \mu(t-n_C) \phi(t-n_C) e^\circ(t-n_C). \end{aligned} \quad (19)$$

A. Relations with other algorithms

Many algorithms have been proposed for accelerating the speed of convergence of the adaptation algorithms derived using the "gradient rule". The algorithm of (14) is termed ARIMA (Autoregressive with Integrator Moving Average). As discussed in [16, Section 8], a number of well known algorithms are particular cases of the ARIMA algorithm. The various algorithms described in the literature are of MAI form or ARI form. The MAI form includes "Integral+Proportional" algorithm [9], [1] ($c_1 \neq 0, c_i = 0, \forall i > 1, d'_i = 0, \forall i > 0$), "Averaged gradient" ($c_i, i = 1, 2, \dots, d'_i = 0, \forall i > 0$) [22], [21]. The ARI form includes "Conjugate gradient" and "Nesterov" algorithms [19], [6] ($c_i = 0, i = 1, 2, \dots, d'_1 \neq 0, d'_i = 0, i > 1$) as well as the "Momentum back propagation" algorithm [8] which corresponds to the conjugate gradient plus a normalization of μ by $(1-d'_1)^3$. A particular form of the ARIMA algorithms termed "ARIMA2" ($c_1, c_2, c_i = 0, \forall i > 2, d'_1 \neq 0, d'_i = 0, \forall i > 1$) will be studied subsequently and evaluated experimentally.⁴

III. DESIGN OF THE DYNAMIC ADAPTATION GAIN/LEARNING RATE

A. Performance issues

The dynamic adaptation gain/learning rate will introduce a phase distortion on the gradient depending upon the frequency. Assume that the algorithms should operate for all frequencies in the range: 0 to $0.5f_s$ (f_s is the sampling frequency). Assume that the gradient of the criterion to be minimized contains a single frequency. In order to minimize the criterion, the phase distortion introduced by the dynamic adaptation gain/learning rate should be less than 90° at

³There are very few indications how to choose the various weights in the above mentioned algorithms.

⁴The algorithms mentioned above can be viewed as particular cases of the ARIMA2 algorithm.

all the frequencies. In other terms, the transfer function $\frac{C(z^{-1})}{D'(z^{-1})}$ should be strictly positive real (SPR) (see [9] for a detailed definition). In order that a transfer function be strictly positive real, it must first have its zeros and poles inside the unit circle. One has the following property:

Lemma 3.1: Assume that the polynomials $C(z^{-1})$ and $D'(z^{-1})$ have all their zeros inside the unit circle, then:

$$I = \int_0^\pi \log \left(\left| \frac{C(e^{-i\omega})}{D'(e^{-i\omega})} \right| \right) d\omega = 0. \quad (20)$$

The proof relies on the Cauchy Integral formula (see [16]).

This result allows to conclude that the average gain over the frequency range 0 to $0.5f_s$ is 0 dB, i.e. on the average this filter will not modify the adaptation gain/step size. It is just a frequency weighting of the adaptation gain/step size. To be more specific, Figure 2 shows the frequency characteristics of various DAG's that will be subsequently used in the experimental section (the VS-LMS algorithm corresponds to the axis 0 dB). It can be observed first that the phase is within the range $\pm 90^\circ$, i.e. they are SPR. Then one can observe that effectively the average gain over the frequency range 0 to $0.5f_s$ ($f_s = 2500$ Hz) is 0 dB. Now examining the magnitude, one observes that all are low pass filters amplifying low frequencies. This means that if the frequency content of the gradient is in the low frequency range, the effective adaptation gain/learning rate will be larger than μ , which should have a positive effect upon the adaptation/learning transient. In particular, the DAG which has a larger gain in low frequencies (ARIMA2) should provide the best performance (which is indeed the case—see Section IV).

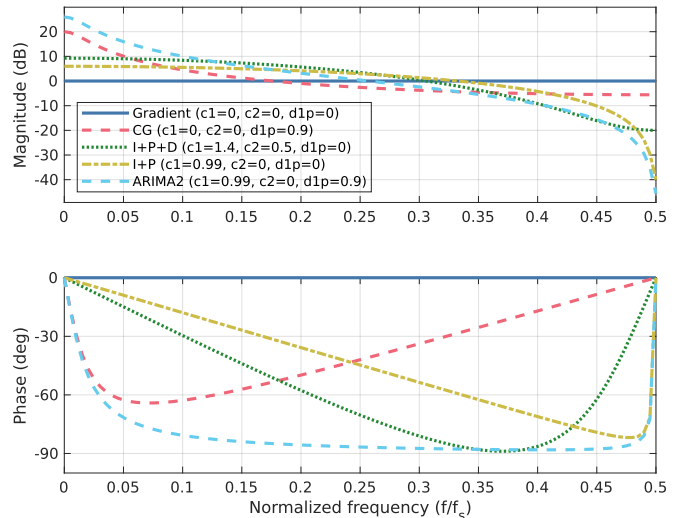


Fig. 2. Frequency characteristics of DAGs used in the experimental section (see also Table I).

Since we need to have a DAG which is SPR, we will provide subsequently the tools for design of a SPR DAG. We will consider the case of the ARIMA2 algorithm introduced in [14]. The DAG in this case will have the form:

$$H_{DAG} = \frac{C(q^{-1})}{D'(q^{-1})} = \frac{1 + c_1 q^{-1} + c_2 q^{-2}}{1 - d'_1 q^{-1}} \quad (21)$$

A criterion for the selection of c_1 , c_2 and d'_1 in order that the DAG be SPR is given below.

Lemma 3.2: The conditions assuring that $H_{DAG}(z) = \frac{1+c_1z^{-1}+c_2z^{-2}}{1-d'_1z^{-1}}$ is strictly positive real (SPR) are:

- for $c_2 \leq 0$, c_1 must be such that

$$-1 - c_2 < c_1 < 1 + c_2$$

- for $c_2 \geq 0$
 - if the following condition is satisfied

$$2(d'_1 - c_2) < \sqrt{2(c_2 - c_2^2)(1 - d_1'^2)} < 2(d'_1 + c_2)$$

the maximum bound on c_1 is given by

$$c_1 < d'_1 - 3d'_1c_2 + 2\sqrt{2(c_2 - c_2^2)(1 - d_1'^2)}$$

otherwise the maximum bound on c_1 is given by $c_1 < 1 + c_2$

- if the following condition is satisfied

$$2(d'_1 - c_2) < -\sqrt{2(c_2 - c_2^2)(1 - d_1'^2)} < 2(d'_1 + c_2)$$

the minimum bound on c_1 is given by

$$c_1 > d'_1 - 3d'_1c_2 - \sqrt{2(c_2 - c_2^2)(1 - d_1'^2)}$$

otherwise the minimum bound on c_1 is given by $c_1 > -1 - c_2$

The proof of this result is given in [16].

From the conditions of Lemma 3.2, closed contours in the plane $c_2 - c_1$ can be defined for the different values of d'_1 allowing to select c_1 and c_2 for a given value of d'_1 such that the DAG be SPR. Note that a necessary condition for the selection of the parameters c_1, c_2, d'_1 is that both the denominator and the numerator of the H_{DAG} should be asymptotically stable.

B. Stability issues

If one wants to use VS-LMS algorithms with large values of the adaptation gain μ , the stability analysis of the resulting scheme using a dynamic adaptation gain is an important issue. For the case of the PLMS algorithm, this analysis has been carried in detail (see [16], [18]). We will just indicate the main result subsequently. Equation (14) can be expressed also as:

$$(1 - q^{-1})\hat{\theta}(t+1) = + \frac{C(q^{-1})}{D'(q^{-1})} [\mu(t)\phi(t)e^o(t+1)]$$

leading to:

$$\hat{\theta}(t+1) = H_{PAA}(q^{-1})[\mu\phi(t)e(t+1)], \quad (22)$$

where $e(t+1) = e^o(t+1)(1 + \mu\phi^T(t)\phi(t))^{-1}$ and $H_{PAA} = (1 - q^{-1})^{-1}H_{DAG}$ is a MIMO diagonal transfer operator having identical terms. All the diagonal terms are described by:

$$\begin{aligned} H_{PAA}^{ii}(q^{-1}) &= \frac{1 + c_1q^{-1} + \dots + c_{n_c}q^{-n_c}}{(1 - q^{-1})(1 - d'_1q^{-1} - \dots - d'_{n_D}q^{-n_D})} \\ &= \frac{C(q^{-1})}{(1 - q^{-1})D'(q^{-1})} = \frac{C(q^{-1})}{D(q^{-1})}. \end{aligned} \quad (23)$$

The relation between the coefficients of polynomials D and D' is given in (17).

One has the following result:

Theorem 3.3: For the system described by Equations (1) through (8) using the PLMS algorithm of (14) and (15) one has $\lim_{t \rightarrow \infty} e(t+1) = 0$ for any positive adaptation gain μ and any initial conditions $\theta(0), e(0)$, if $H_{PAA}^{ii}(z^{-1})$ given in (23) is a PR transfer function with a pole at $z = 1$.

The proof is given in [16].

The adaptive/learning system considered in the Theorem 3.3 leads to an equivalent feedback representation where the equivalent feedforward path is a constant positive gain and the equivalent feedback path features the H_{PAA} (see [14]). However, in many cases the equivalent feedforward path will be a transfer operator. In such situations in addition to the PR condition upon the H_{PAA} , there will be an additional SPR condition upon the transfer operator characterizing the equivalent feedforward path.

For small values of the adaptation gains/learning rates, the passivity/stability condition can be relaxed using *averaging* [4]. In this case, the most important is that the H_{PAA}^{ii} is PR in the frequency region of operation (mainly defined by the spectrum of the input signals to the systems).

It is interesting to see intersections of the contours assuring the SPR of the H_{DAG}^{ii} with the contours assuring that H_{PAA}^{ii} is PR. Such an intersection is shown in Fig.3 for the case of the H_{DAG} given in (21). From this figure one can conclude that not all the SPR H_{DAG} will lead to a PR H_{PAA} . In such cases, the performance is improved for low adaptation gains, but one can not guarantee asymptotic stability for large values of the adaptation gain. Fig. 3 shows also that there is a region where despite that H_{PAA} is PR, H_{DAG} is not SPR. For such configurations, large adaptation gains can be used but the adaptation transient is slower than for the basic gradient algorithm.

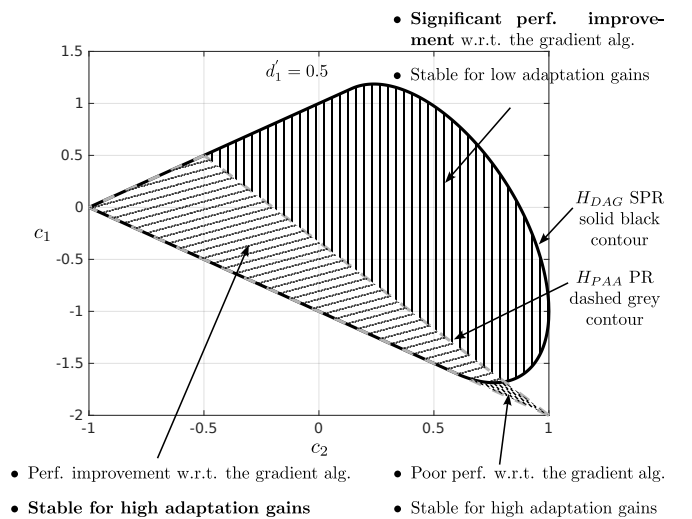


Fig. 3. Intersection in the plane $c_1 - c_2$ of the contour $H_{PAA} = PR$ with the contour $H_{DAG} = SPR$ for $d'_1 = 0.5$.

IV. EXPERIMENTAL RESULTS

The objective of this section is to show that the dynamic adaptation gain can be implemented on various VS-LMS algorithms and this will lead to a significant acceleration of the adaptation transient. Specifically for this paper the LMS, NLMS and PLMS algorithms have been implemented and tested experimentally on an active noise control test-bench (adaptive feedforward noise attenuation). Figure 4 shows the view of the test-bench used for experiments. Detailed description can be found in [16]. The speakers and the microphones were connected to a target computer with the Simulink Real-Time[®]. A second computer is used for development and operation with Matlab/Simulink. The sampling frequency was $f_s = 2500$ Hz.

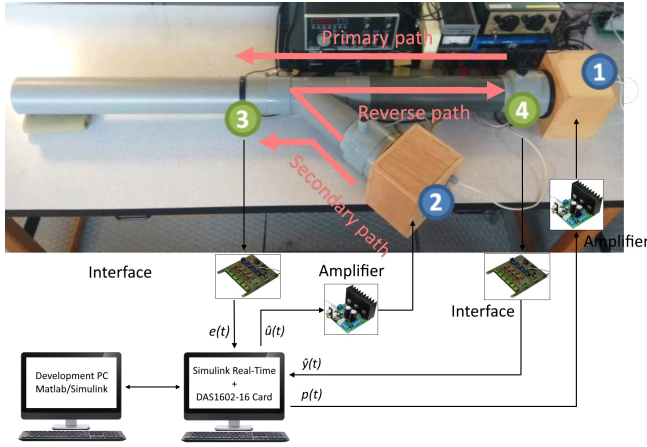


Fig. 4. Duct active noise control test-bench .

The models of the various paths in Figure 4 are characterized by the presence of many pairs of very low damped poles and zeros. These models have been identified experimentally. The orders of the various identified models are: $n_G = 33$, $n_M = 27$ and $n_D = 27$.

The objective is to attenuate an incoming unknown broadband noise disturbance. The corresponding block diagram for the adaptive feedforward noise compensation using FIR Youla-Kucera (FIR-YK) parametrization of the feedforward compensator (introduced in [15] for active vibration control and in [2] for active noise control) is shown in Figure 5.

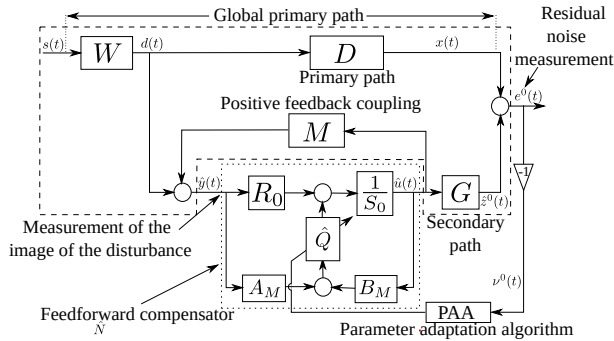


Fig. 5. Feedforward AVC with FIR-YK adaptive feedforward compensator.

The adjustable filter \hat{Q} has the FIR structure:

$$\hat{Q}(q^{-1}) = \hat{q}_0 + \hat{q}_1 q^{-1} + \dots + \hat{q}_{n_Q} q^{-n_Q} \quad (24)$$

and the parameters \hat{q}_i will be adapted in order to minimize the residual noise. The algorithm can be summarized as follows. One defines:

$$\theta^T = [q_0, q_1, q_2, \dots, q_{n_Q}] \quad (25)$$

$$\hat{\theta}^T = [\hat{q}_0, \hat{q}_1, \hat{q}_2, \dots, \hat{q}_{n_Q}] \quad (26)$$

$$\phi^T(t) = [v(t+1), v(t), \dots, v(t-n_Q+1)] \quad (27)$$

where:

$$v(t+1) = B_M \hat{y}(t+1) - A_M \hat{u}(t+1) = B_M^* \hat{y}(t) - A_M \hat{u}(t+1) \quad (28)$$

One defines also the regressor vector (a filtered observation vector) as:

$$\phi_f(t) = L(q^{-1})\phi(t) = [v_f(t+1), v_f(t), \dots, v_f(t-n_Q+1)] \quad (29)$$

where

$$v_f(t+1) = L(q^{-1})v(t+1) \quad (30)$$

Using $R_0 = 0$ and $S_0 = 1$, the poles of the internal positive closed loop are defined by A_M and they will remain unchanged. The filter used in (30) becomes $L = \hat{G}$ and the associated linear transfer operator appearing in the equivalent feedforward path is

$$H(q^{-1}) = \frac{G(q^{-1})}{\hat{G}(q^{-1})} \quad (31)$$

(the algorithm uses an approximate gradient). The transfer function associated to $H(q^{-1})$ should be SPR in order to assure asymptotic stability in the case of perfect matching. This is a very mild condition as long as a good experimental identification of the models is done.

The VS-LMS algorithms which were used are of the form given in (9) where $e^\circ(n)$ is given by (7) with z° given in (18), where $\hat{\theta}$ is given by (26) and ϕ is replaced by ϕ_f given in (29)⁵. The adjustable filter \hat{Q} had 60 parameters.

The values of the adaptation gain μ for the three algorithms were tuned such that in the absence of the DAG, the performance was close for the three algorithms. A specificity of this application is also the low value of the average of $\phi(n)^T \phi(n)$ ($\ll 1$). This means that the LMS and PLMS show a very close behavior for a given μ and that the adaptation gain μ for the NLMS should be much lower in order to get similar performances. The choices of μ are 0.2 for LMS, 0.0002 for NLMS and 0.22 for the PLMS.

A broad-band disturbance 70–170 Hz was used as an unknown disturbance acting on the system. The steady state and transient attenuation⁶ were evaluated for the various values of the parameters c_1 , c_2 and d_1^i given in Table I.

⁵In signal processing literature when using a filtering of the regressor, the algorithms are termed FU-VSLMS.

⁶The attenuation is defined as the ratio between the variance of the residual noise in the absence of the control and the variance of the residual noise in the presence of the adaptive feedforward compensation. The variance is evaluated over an horizon of 3 seconds.

TABLE I

PARAMETERS OF ARIMA2 DYNAMIC ADAPTATION GAIN.

Algorithm	H_{PAA-PR}	$H_{DAG-SPR}$	c_1	c_2	d'_1
Integral (gradient)	Y	Y	0	0	0
Conj.Gr/Nest..	N	Y	0	0	0.9
I+P+D	N	Y	1.4	0.5	0
I+P	Y	Y	0.99	0	0
ARIMA 2	N	Y	0.99	0	0.9

The system was operated in open-loop during the first 15 s. Figure 6 shows the time response of the system as well as the evolution of the global attenuation when using the standard NLMS algorithm (top) and when the ARIMA2 DAG is incorporated (bottom) with $c_1 = 0.99$, $c_2 = 0$, $d'_1 = 0.9$ (last row of Table I). One observes a significant acceleration of the adaptation transient. The acceleration obtained is equivalent to that obtained with an adaptation gain 25 times higher using the standard NLMS algorithms. Similar results were obtained for the LMS and the PLMS algorithms.

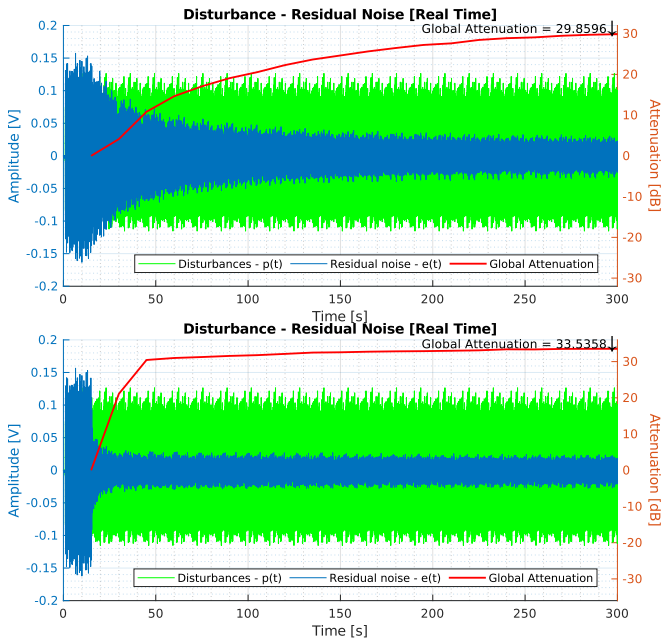


Fig. 6. Time evolution of the residual noise using the standard NLMS adaptation algorithm (top) and using the NLMS + ARIMA2 algorithm (bottom), for $\mu = 0.0002$.

Figures 7, 8, and 9 give the time evolution of the attenuation for the LMS, NLMS and PLMS algorithms and for the various DAG given in Table I. As one can observe, the effect of the DAG is similar for the three algorithms.

V. CONCLUSION

The paper emphasizes the potential of the dynamic adaptation/step size (DAG) for improving the adaptation transients of VS-LMS adaptation algorithms. The main point is that the DAG should be characterized by a *strictly positive real* (SPR) transfer function if we would like to operate correctly for any frequency in the range 0 to 0.5 of the sampling frequency. Experimental results on a relevant adaptive active

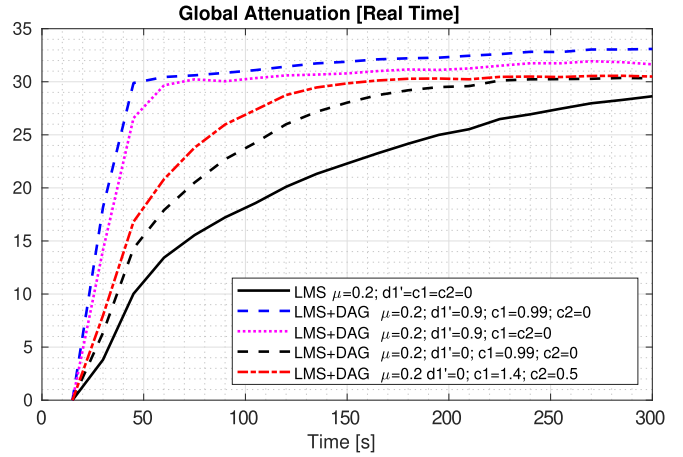


Fig. 7. Time evolution of the global attenuation for the LMS algorithm with various DAGs, using $\mu = 0.2$.

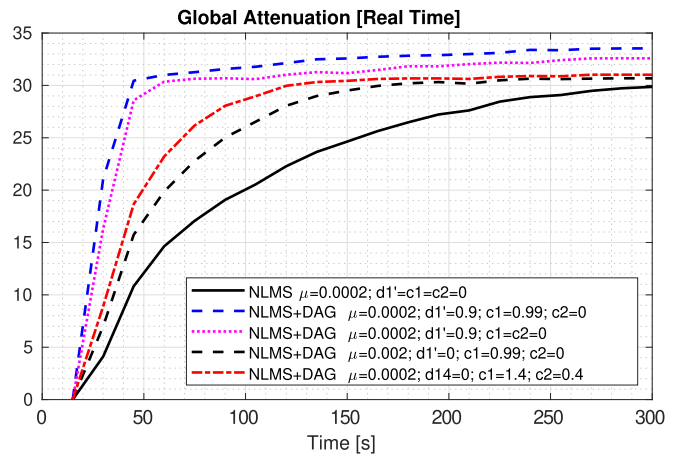


Fig. 8. Time evolution of the global attenuation for the NLMS algorithm with various DAGs, using $\mu = 0.0002$.

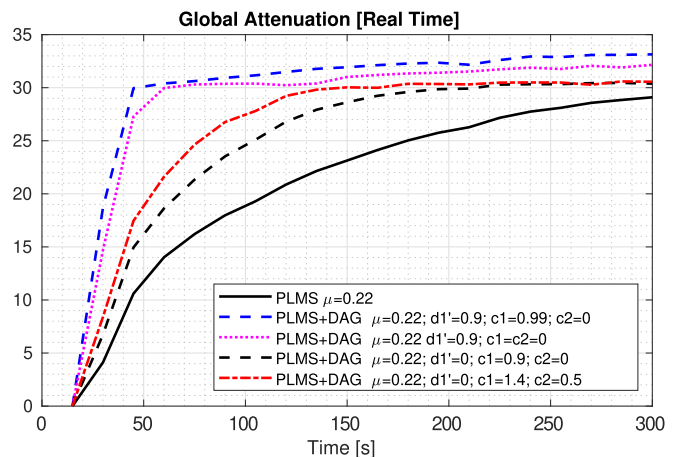


Fig. 9. Time evolution of the global attenuation for the PLMS algorithm with various DAGs, using $\mu = 0.22$.

noise control system have illustrated the feasibility and the performance improvement achieved using a DAG on VS-

LMS adaptation algorithms.

APPENDIX I
DERIVATION OF THE PLMS ALGORITHM

The objective is to minimize the criterion:

$$\min_{\hat{\theta}(t)} J(t) = \mathbf{E}\{[e(t)^2]\} \quad (32)$$

In order to apply the gradient rule one has to estimate the gradient of the criterion with respect to $\theta(n)$.

$$\nabla_{\theta} J(t) = 2\mathbf{E} \left\{ \frac{\partial e(t)}{\partial \hat{\theta}(t)} \cdot e(t) \right\} \quad (33)$$

but from (8) it results that

$$\frac{\partial e(t)}{\partial \hat{\theta}(t)} = -\phi(t) \quad (34)$$

and one obtains the algorithm:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \mu \mathbf{E}\{\phi(t)e(t)\} \quad (35)$$

$\mathbf{E}\{\phi(t)e(t)\}$ can be approximated by

$$\mathbf{E}\{\phi(t)e(t)\} = \frac{1}{N} \sum_{i=0}^{N-1} \phi(t-i)e(t-i) \quad (36)$$

and taking $N = 1$, the algorithm of (35) becomes⁷:

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \mu \phi(t)e(t); \quad \mu > 0 \quad (37)$$

We have to give now an implementable expression of this algorithm. Observe that (8) can be re-written as (using (37))

$$e(t) = [\theta - \hat{\theta}(t)]^T \phi(t) = e^{\circ} - \mu \phi(t)^T \phi(t) e(t) \quad (38)$$

yielding

$$e(t) = \frac{e^{\circ}}{1 + \mu \phi(t)^T \phi(t)} \quad (39)$$

and the adaptation algorithm takes the form given in (12) and (13).

REFERENCES

- [1] T.-B. Airimitoiaie and I. D. Landau, "Improving adaptive feedforward vibration compensation by using integral+proportional adaptation," *Automatica*, vol. 49, no. 5, pp. 1501–1505, 2013.
- [2] T.-B. Airimitoiaie, I. D. Landau, R. Melendez, and L. Dugard, "Algorithms for Adaptive Feedforward Noise Attenuation—A Unified Approach and Experimental Evaluation," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 5, pp. 1850–1862, Sept. 2021.
- [3] A. E. Albert and L. A. Gardner, *Stochastic approximation and nonlinear regression*. Cambridge, Mass.: M.I.T. Press, 1967, oCLC: 973152.
- [4] B. Anderson, R. Bitmead, C. Johnson, P. Kokotovic, R. Kosut, I. Mareels, L. Praly, and B. Riedle, *Stability of adaptive systems*. Cambridge Massachusetts, London, England: The M.I.T Press, 1986.
- [5] D. Bismor, K. Czyz, and Z. Ogonowski, "Review and Comparison of Variable Step-Size LMS Algorithms," *The International Journal of Acoustics and Vibration*, vol. 21, no. 1, 2016.

⁷Alternatively one can consider the following approximation:

$$\mathbf{E}\{\phi(t)e(t)\} = \frac{1}{N+1} \sum_{i=-N/2}^{N/2} \phi(t+i)e(t+i)$$

and one takes $N = 0$.

- [6] J. E. Gaudio, T. E. Gibson, A. M. Annaswamy, and M. A. B. E. Lavretsky, "Connections between adaptive control and optimization in machine learning," in *Proceedings of the IEEE CDC Conference, Dec. 2019, Nice, France*, 2019, pp. 4563–4568.
- [7] G. Goodwin and K. Sin, *Adaptive filtering prediction and control*. N. J.: Prentice Hall, 1984.
- [8] R. Jacobs, "Increased rates of convergence through learning rate adaptation," *Neural Networks*, vol. 27, no. 1, pp. 295–307, Winter 1988.
- [9] I. D. Landau, R. Lozano, M. M'Saad, and A. Karimi, *Adaptive control*, 2nd ed. London: Springer, 2011.
- [10] I. Landau, *Identification et commande des systèmes*, 1st ed., ser. Série Automatique. Paris: Hermès, 1988.
- [11] —, *System Identification and Control Design: Using P.I.M.+ Software*, ser. Information and System Sciences Series. Prentice Hall, 1990.
- [12] I. D. Landau, "Synthesis of hyperstable discrete model reference adaptive systems," in *Fifth Asilomar Conference on Circuits and Systems*, Asilomar, California, USA, Nov. 1971.
- [13] —, "Design of discrete model reference adaptive systems using the positivity concept," in *Proceedings 3rd IFAC Symp. on Sensitivity, Adaptivity and Optimality*, Ischia, Italy, June 1973, pp. 307–314.
- [14] I. D. Landau and T.-B. Airimitoiaie, "Does a general structure exist for adaptation/learning algorithms?" in *CDC 2022 - 61st IEEE Conference on Decision and Control*, Cancun, Mexico, Dec. 2022. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-03738223>
- [15] I. D. Landau, T.-B. Airimitoiaie, and M. Alma, "A Youla–Kučera parametrized adaptive feedforward compensator for active vibration control with mechanical coupling," *Automatica*, vol. 48, no. 9, pp. 2152 – 2158, 2012.
- [16] I. D. Landau, T.-B. Airimitoiaie, B. Vau, and G. Buche, "On a general structure for adaptation/learning algorithms - stability and performance issues," *Automatica*, vol. 156, p. 111193, Mar. 2023.
- [17] I. D. Landau and G. Béthoux, "Algorithms for discrete time model reference adaptive systems," in *6th IFAC World Congress (IFAC 1975)*, Boston, USA, August 1975, pp. 682–692.
- [18] I. D. Landau, B. Vau, T.-B. Airimitoiaie, and G. Buche, "Improving performance of adaptive feedforward noise attenuators using a dynamic adaptation gain," *Journal of Sound and Vibration*, vol. 560, p. 117790, 2023.
- [19] I. Livieris and R. Pintelas, "A survey on algorithms for training artificial neural networks," *University of Patras Report*, vol. 455, no. TR08-01, 2014.
- [20] J. Nagumo and A. Noda, "A learning method for system identification," *IEEE Transactions on Automatic Control*, vol. 12, no. 3, pp. 282–287, 1967.
- [21] S. Pouyanfar, S. Sadiq, Y. Yan, and H. Tian, "A survey on deep learning: Algorithms, techniques, and applications," *ACM Computing Surveys*, Vol. 1, No. 1, Article 1. Publication date: January 2017., vol. 1, January 2017.
- [22] M. Schmidt, "Stochastic average gradient," *University of British Columbia.*, Winter 2018.
- [23] H. P. Whitaker, J. Yarmon, and A. Keezer, *Synthesis of model reference adaptive control systems by Lyapunov's second method*, Cambridge, USA, September 1958.
- [24] B. Widrow, J. Glover, J.R., J. McCool, J. Kaunitz, C. Williams, R. Hearn, J. Zeidler, J. Eugene Dong, and R. Goodlin, "Adaptive noise cancelling: Principles and applications," *Proceedings of the IEEE*, vol. 63, no. 12, pp. 1692 – 1716, dec. 1975.
- [25] B. Widrow and M. E. Hoff, "Adaptive switching circuits," in *1960 IRE WESCON Convention Record, Part 4*. New York: IRE, 1960, pp. 96–104.