Article

Journal of Vibration and Control
1–16

DOI: 10.1177/1077546315626323
jvc.sagepub.com
SAGE

Combination of map-based and adaptive feedforward control algorithms for active engine mounts

F Hausberg[1], M Plöchl[2], M Rupp[3], P Pfeffer[4] and S Hecker[5]

Abstract

Active engine mounts significantly contribute to ensure the comfort in vehicles with emission-reducing engine technologies, e.g., cylinder-on-demand (COD), downsizing or turbochargers. To control active engine mounts, either adaptive or non-adaptive feedforward control is commonly employed. Since both approaches have previously been treated separately, this study proposes methods to connect them in terms of multiple-input-multiple-output Newton/FxLMS adaptive filters with self-trained, grid-based look-up tables. The look-up tables are incorporated as parameter-maps or parallel-maps, respectively. By combining the two feedforward control strategies, their inherent advantages, i.e., the adaptivity of adaptive filtering and the direct impact as well as the tracking behavior of map-based feedforward control, are utilized. The proposed control structures are illustrated by simulation and experimentally demonstrated in a vehicle with a V8-COD engine. While both methods significantly reduce the convergence time of the adaptive filter, the parallel implementation additionally improves the tracking behavior during fast engine run-ups.

Keywords

Active engine mounts, active vibration control, adaptive feedforward control

1. Introduction

To meet the higher demand for low fuel consumption and legal restrictions to reduce emissions, modern engine technologies, e.g., cylinder-on-demand (COD), downsizing or turbochargers, are increasingly applied in today's vehicles. In combination with lightweight car bodies, it becomes a more and more challenging task to satisfy the noise, vibration and harshness (NVH) expectations of the customer. Against this background, the design of engine mounts, being the main connection between the drive train and the chassis, is of particular importance. Besides supporting the static engine weight and the isolation of high-frequency (>20 Hz) engine-induced vibrations, their main task is to prevent low-frequency (<20 Hz) engine bounce from shock excitation (Yu et al., 2001). In the past, the resulting conflict of frequency-dependent stiffness and damping properties has been solved as far as possible with more elaborate passive engine mount designs, e.g., hydraulic engine mounts (HEMs) (Marzbani et al., 2014).

An additional improvement in the acoustic and vibrational comfort of passenger cars can be achieved by the application of active engine mounts (AEMs). In contrast to switchable or semi-active engine mounts, AEMs cancel the disturbing engine vibration by introducing a signal of equal amplitude and opposite phase through an actuator located inside the AEM. To control AEMs, two approaches are widely used. In the first, adaptive feedforward control is applied to generate an appropriate control signal – in particular the use of finite impulse response (FIR) filters whose coefficients are adapted with the filtered-x least-mean-squares (FxLMS) algorithm (Riley et al., 1995; Togashi and Ichiryu, 2003; Hillis et al., 2005; Hillis,

[1]Audi AG, I/EF-35, Engine/Gearbox Mountings, Germany
[2]TU Wien, Institute of Mechanics and Mechatronics, Austria
[3]TU Wien, Institute of Telecommunications, Austria
[4]Department of Mechanical, Automotive and Aeronautical Engineering, Munich University of Applied Sciences, Germany
[5]Department of Electrical Engineering and Information Technology, Munich University of Applied Sciences, Germany

Received: 27 January 2015; accepted: 16 December 2015

Corresponding author:
M Plöchl, Institute of Mechanics and Mechatronics, Getreidemarkt 9, 1060 Wien, Austria.
Email: manfred.ploechl@tuwien.ac.at

2011) or one of its variants (Aoki et al., 1999; Nakaji et al., 1999; Fursdon, 2006) is common practice. Since adaptive feedforward control employs an accelerometer on the chassis, it guarantees full vibration compensation, despite any changes in the primary transfer paths or engine disturbances. However, due to its adaptive nature, adaptive feedforward control always requires a certain convergence time after its activation and may possess poor tracking behavior. A second approach, non-adaptive feedforward control, has been proposed for the control of AEMs. In this case, data maps obtained from prior measurements (Matsuoka et al., 2004) or an analytical engine model (Lee and Lee, 2009) are used to generate an appropriate control signal. Besides its general advantage that no error sensor is necessary, map-based feedforward control exhibits no convergence time until full vibration compensation is achieved. However, as map-based feedforward control is non-adaptive, it is unable to track any changes in the transfer paths. In addition, extensive measurements are necessary to identify appropriate data maps or engine models for feedforward control.

This article proposes methods that combine adaptive and map-based feedforward control in order to overcome their respective disadvantages. First, the available AEM and the COD-technology of the test vehicle are described in Section 2. Afterwards, the Newton/FxLMS algorithm, a variant of the narrowband-FxLMS algorithm, is extended with online-adapted, grid-based look-up tables in Sections 3 and 4. Two control structures are presented and their inherent properties are discussed. A simulation example illustrates the proposed control structures in Section 5. Finally, the developed methods are experimentally validated in Section 6. Section 7 closes the paper with some concluding remarks.

2. Active engine mounts

Figure 1 shows a schematic diagram of the AEM that has been employed for this study. Comparable engine mounts have been used in the studies of Hillis (2011), Mansour et al. (2010), Mansour et al. (2011), Fakhari and Ohadi (2012), Fakhari et al. (2013), and Hosseini et al. (2013); also, recently a thorough experimental and analytical study of its dynamic characteristics through the use of a mechanical model has been published by one of the authors (Hausberg et al., 2015). The upper fluid chamber is bounded by the main rubber spring, which supports the static engine weight. When the main rubber spring is compressed, fluid is forced through the fluid channel into the lower chamber. The fluid's inertia acts like a tuned mass damper and provides additional damping to counteract low frequency (<20 Hz) engine vibrations induced by road disturbances. A diaphragm

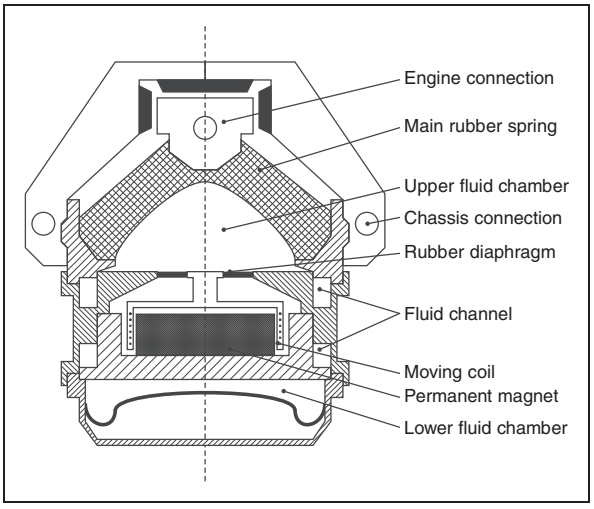

Figure 1. Schematic diagram of an active engine mount.

separates the upper from the lower fluid chamber. A moving coil actuator is attached to the diaphragm. Through the control of the actuator, the pressure in the upper chamber can be influenced, generating a force to cancel high frequency engine disturbances (>20 Hz).

The passenger car that has been used for this study is equipped with two AEMs and two accelerometers located on the chassis. The vehicle's V8-COD engine deactivates half of its cylinders in driving situations where only low torque is necessary. Engine vibrations normally consist of several narrowband components, so-called engine orders, whose frequencies are directly related to the engine speed ω. The partial deactivation of the cylinders results in a change of the dominant engine order as well as the excitation level of the disturbing engine vibration. This is illustrated by the spectrograms of the vertical acceleration measured at the left engine mount's chassis side in Figure 2. When half of the cylinders are deactivated, the number of ignitions per crankshaft revolution is halved and thus the second engine order dominates. Furthermore, an overall increase in excitation amplitude can be observed. This can be attributed to the shift of the dominant engine order to lower frequencies and the increased gas pressure in the four remaining active cylinders. The AEMs are activated to compensate for these changes. Due to the frequent activation and deactivation of the employed control algorithm, a short convergence time is necessary.

3. Adaptive feedforward control algorithm

A very popular adaptive feedforward algorithm in active vibration control is the FxLMS algorithm.

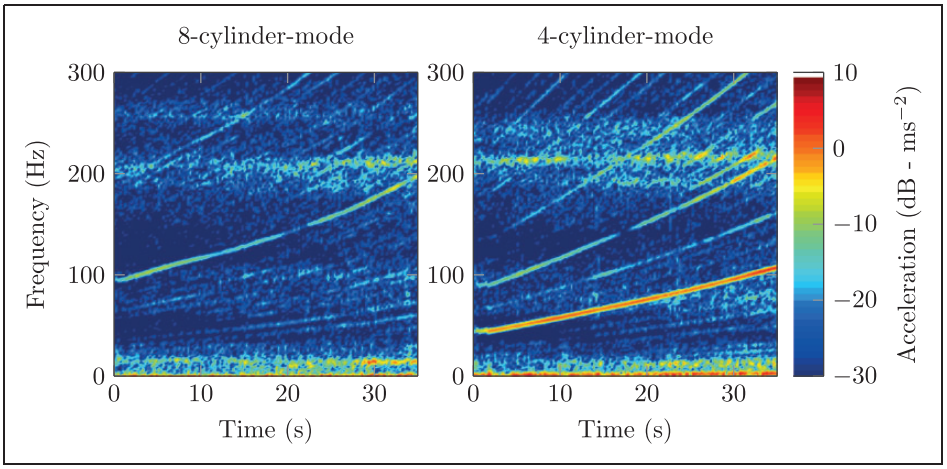

Figure 2. Spectrograms of the vertical acceleration measured at the chassis side of the left AEM during acceleration in fifth gear with full and half cylinder mode.

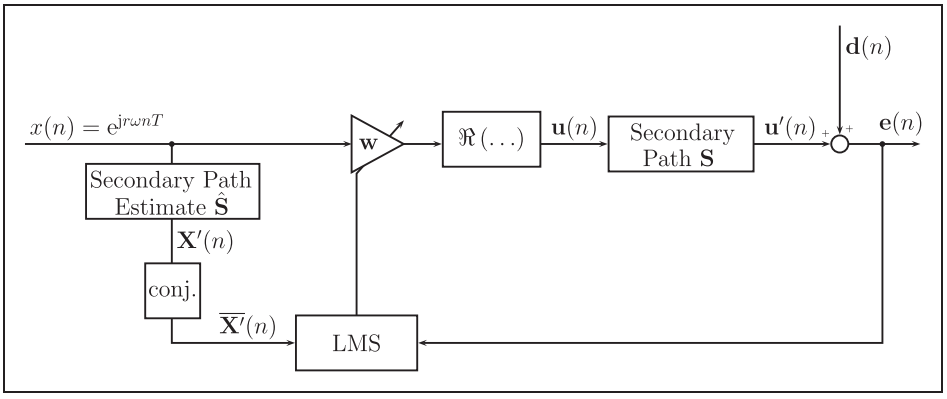

Figure 3. Block diagram of the complex Narrowband-FxLMS algorithm.

It was initially proposed by Morgan (1980) and has been independently developed for adaptive inverse control and active noise control by Widrow et al. (1981) and Burgess (1981), respectively. In this study, a multiple-input-multiple-output (MIMO) complex, narrowband variant of the FxLMS algorithm is employed. A similar approach has already been implemented, e.g., by Johansson (2000), and its convergence has been analyzed by Rupp and Hausberg (2014). A general block diagram of the algorithm is shown in Figure 3.

The m-th entry of the vector $\mathbf{e}(n)$ of error signals measured at the chassis side can be written as

$$e_m(n) = d_m(n) + \sum_{k=1}^{K} \sum_{l=0}^{L-1} s_{mk}(l) u_k(n-l), \quad \text{for } m = 1, 2, \ldots, M \tag{1}$$

where d_m is the m-th entry of the vector $\mathbf{d}(n)$ of disturbance signals to be canceled, u_k is the k-th entry of the control output vector $\mathbf{u}(n)$, and $s_{mk}(l)$ is the l-th coefficient of the impulse response of the secondary path's transfer function $S_{mk}(z)$ between the k-th control signal and the output of the m-th accelerometer.

A complex reference signal is generated with the engine speed ω obtained from a crankshaft tachometer and the engine order r to be canceled. Since the synthesized reference signal is generated internally, an undesired feedback loop from the actuator's output back to the reference sensor is avoided (Kuo and Morgan, 1999). The complex reference signal has to be filtered with a transfer function matrix $\hat{\mathbf{S}}(z)$ of secondary path estimates $\hat{S}_{mk}(z)$, in order to compensate for the dynamics of the secondary path $\mathbf{S}(z)$. The computational effort of generating the filtered reference signals is reduced by expressing the secondary path estimates by their respective amplitude $A_{mk}(r\omega)$ and phase angle $\varphi_{mk}(r\omega)$ at the current operating frequency of the adaptive feedforward control algorithm.

In this case, the matrix of complex filtered reference signals is given by

$$\mathbf{X}'(n) = \hat{\mathbf{S}}^T(r\omega)x(n) = \hat{\mathbf{S}}^T(r\omega)\mathrm{e}^{\mathrm{j}r\omega n\mathrm{T}} \tag{2}$$

where

$$\hat{S}_{mk}(r\omega) = A_{mk}(r\omega)\mathrm{e}^{\mathrm{j}\varphi_{mk}(r\omega)}, \quad \text{for } m = 1, 2, \ldots, M \text{ and } k = 1, 2, \ldots, K \tag{3}$$

is the mk-th entry of $\hat{\mathbf{S}}(r\omega)$. In this study, the entries of $\hat{\mathbf{S}}(r\omega)$ are identified experimentally using stepped sine test signals. Figure 4 shows the experimentally identified amplitude and phase angle of the secondary path $\hat{S}_{11}$ between the left active engine mount and the accelerometer located at the left engine mount's chassis side. The secondary path has been identified in steps of 5 Hz in the frequency range of interest between 25 Hz and 250 Hz.

According to Widrow et al. (1975), the complex conjugate $\overline{\mathbf{X}}'(n)$ of the reference signal matrix $\mathbf{X}'(n)$ has to be applied in the weight update equation of a complex-valued LMS algorithm. The complex filter weight vector $\mathbf{w}(n)$ is updated according to

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu\hat{\mathbf{S}}^H(r\omega)\mathrm{e}^{-\mathrm{j}r\omega n\mathrm{T}}\mathbf{e}(\mathrm{n}), \tag{4}$$

in order to minimize the mean square of the error signal vector $\mathbf{e}(n)$. The choice of the step-size parameter μ determines the convergence rate and stability of the algorithm. Finally, the control output vector $\mathbf{u}(n)$ is obtained by multiplying the complex filter weight vector with the complex reference signal and taking the real part of the signal (Rupp and Hausberg, 2014):

$$\mathbf{u}(n) = \Re\{\mathbf{w}(n)x(n)\} \tag{5}$$

According to Elliott (2001), the convergence properties of the FxLMS algorithm in equation (4) are determined by the eigenvalue spread of the matrix product $\hat{\mathbf{S}}^H(r\omega)\mathbf{S}(r\omega)$, i.e., the ratio of the largest to the smallest eigenvalue. Due to the dynamic characteristics of the secondary paths, the entries of $\hat{\mathbf{S}}^H(r\omega)$ and $\mathbf{S}(r\omega)$ vary in the operating frequency range of the adaptive algorithm, resulting in unequal convergence rates at different operating frequencies. Therefore, in order to obtain an equal convergence rate in the frequency range of interest, the following Newton/FxLMS update equation is introduced in this work:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu\hat{\mathbf{S}}^{-1}(r\omega)\mathrm{e}^{-\mathrm{j}r\omega n\mathrm{T}}\mathbf{e}(\mathrm{n}) \tag{6}$$

Substitution of equation (1) into equation (6) and application of the frequency domain analysis proposed in Elliott (2001), leads to $\hat{\mathbf{S}}^{-1}(r\omega)\mathbf{S}(r\omega)$ for the relevant matrix product, whose eigenvalues determine the convergence properties of the algorithm. This matrix product becomes the identity matrix provided a perfect

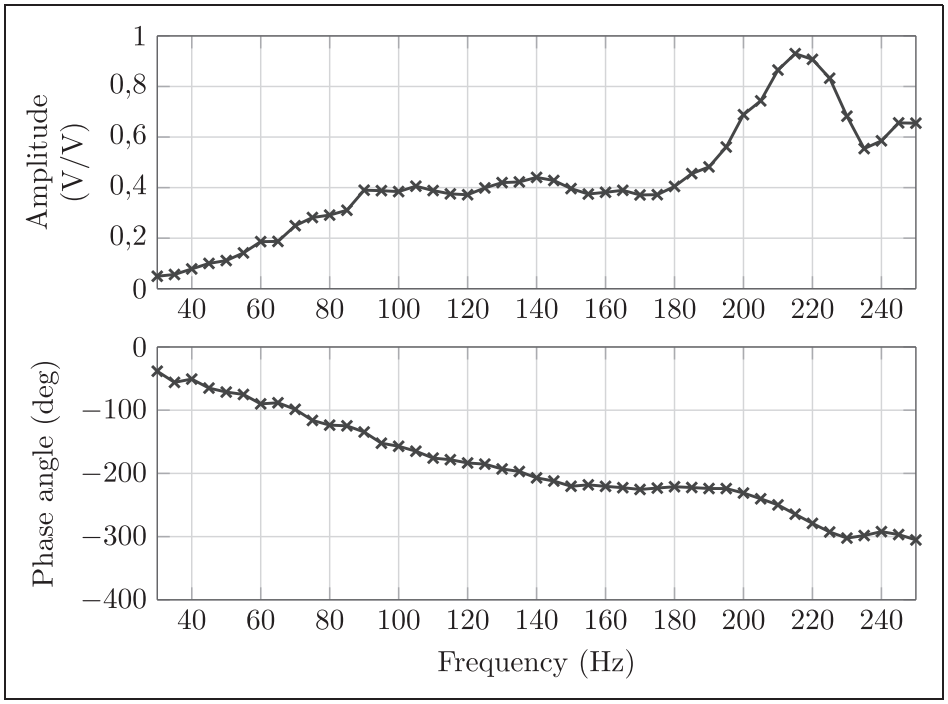

Figure 4. Experimentally identified secondary path $\hat{S}_{11}$ between the left active engine mount and the accelerometer located at the left engine mount's chassis side.

estimation of the secondary paths exists. Thus, the convergence rate is independent of the current operating frequency, since the eigenvalues are always equal to one. A more detailed convergence analysis and a comparison to the conventional FxLMS algorithm is given in the appendix. According to equation (6), in the block diagram of Figure 3 the conjugation is only applied to the reference signal $x(n)$ and then multiplied by $\hat{\mathbf{S}}^{-1}$, which substitutes the secondary path estimate $\hat{\mathbf{S}}$ in the conventional FxLMS algorithm.

Figure 5 shows the convergence of the normalized total control effort $\mathbf{u}^T(n)\mathbf{u}(n)/(\mathbf{u}_{\text{opt}}^T(r\omega)\mathbf{u}_{\text{opt}}(r\omega))$ for the FxLMS algorithm in equation (4) and the Newton/FxLMS algorithm in equation (6) activated at two different engine rotation speeds. Herein, $\mathbf{u}_{\text{opt}}(r\omega)$ describes the steady-state solution, which minimizes the mean square error after convergence of the adaptive algorithm. As expected, the Newton/FxLMS algorithm shows an equal rate of convergence in both cases, while the FxLMS algorithm exhibits a significantly slower convergence at low engine speeds compared to higher engine speeds.

Although the matrix inversion $\hat{\mathbf{S}}^{-1}$ in equation (6) has been realizable in the rapid control prototyping unit used for this study, its computational load and numerical requirements might be too large for practical applications in mass production processors. In this case, several methods are applicable:

- If the secondary paths do not change significantly over time, their inverse could be determined *a priori* offline and stored into look-up tables (Fraanje, 2004). The look-up table values would be scheduled with the current engine rotation speed. However, if secondary path changes occur during vehicle operation, due to temperature or engine load, it has to be permanently adapted, in order to guarantee the stability of the adaptive algorithm. In this case, the matrix inversion has to be calculated online as well.
- Instead of applying the matrix inversion $\hat{\mathbf{S}}^{-1}$, a frequency-dependent normalized step-size $\tilde{\mu}(r\omega)$ can be introduced in the adaptation law of the conventional FxLMS algorithm in equation (4). Johansson (2000) proposes an actuator-individual normalization

$$\tilde{\mu}(r\omega) = \mu_0 \begin{pmatrix} \frac{1}{\kappa + \sum_{m=1}^{M} \hat{A}_{m1}^2(r\omega)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\kappa + \sum_{m=1}^{M} \hat{A}_{mk}^2(r\omega)} \end{pmatrix}, \quad \text{for } k = 1, 2, \ldots, K \tag{7}$$

where μ_0 is a global, frequency-independent step-size and κ is a constant value to limit the maximum overall step-size. In this case equation (4) becomes:

$$\mathbf{w}(n+1) = \mathbf{w}(n) - \mu_0 \cdot \begin{pmatrix} \frac{1}{\kappa + \sum_{m=1}^{M} \hat{A}_{m1}^2(r\omega)} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\kappa + \sum_{m=1}^{M} \hat{A}_{mk}^2(r\omega)} \end{pmatrix} \hat{\mathbf{S}}^H(r\omega) \mathrm{e}^{-\mathrm{j}r\omega\mathrm{n}\mathrm{T}} \mathbf{e}(\mathrm{n}) \tag{8}$$

In this case, one tries to compensate the phase factors for each frequency component, but the length

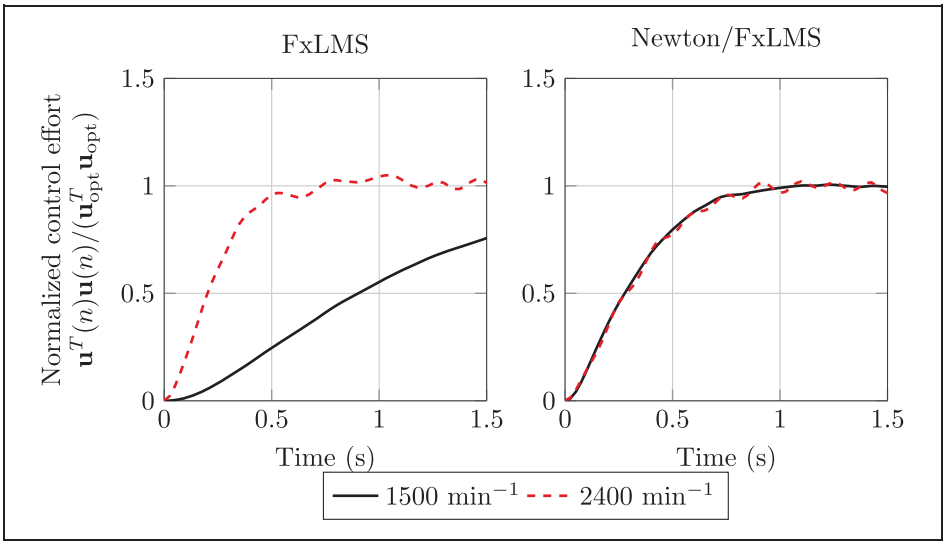

Figure 5. Normalized control effort of the conventional FxLMS algorithm and the Newton/FxLMS algorithm.

is adjusted by a common factor μ_0. As the diagonal matrix is an approximation for $\mathbf{S}^{-1}$, the eigenvalue spread is lowered only to a certain extent. Hence, the method is a compromise.

- As only the expression $\hat{\mathbf{e}}(n) = \hat{\mathbf{S}}^{-1}(n)\mathrm{e}^{-\mathrm{j}r\omega n\mathrm{T}}\mathbf{e}(\mathrm{n})$ is needed to compute equation (6), it could be determined by a second subsidiary iterative algorithm rather than explicitly computing $\hat{\mathbf{S}}^{-1}$. Practically, the complexity has to be limited and thus such iterations run only L times. The result is an approximation of the desired term $\hat{\mathbf{e}}(n)$ and L defines its precision. The subsidiary adaptation law is obtained by introducing $n = L\ l$

$$\hat{\mathbf{e}}(n, l+1) = \hat{\mathbf{e}}(n, l) + \mu_{\mathrm{iter}}\hat{\mathbf{S}}^{H}(n) \cdot \left(\mathbf{e}(n)\mathrm{e}^{-\mathrm{j}r\omega n\mathrm{T}} - \hat{\mathbf{S}}(n)\hat{\mathbf{e}}(n, l)\right), \quad \text{for } l = 1, 2, \ldots, L \tag{9}$$

For the initial condition $\hat{\mathbf{e}}(n, 1)$, *a priori* knowledge, e.g. $\mathbf{e}(n-1)$ or $\hat{\mathbf{e}}(n-1, L)$ can be used. However, convergence of equation (9) has to be ensured, in order to guarantee stability of the Newton/FxLMS algorithm in equation (6). Using the iterative approach in equation (9), a low-complexity solution has been proposed that is numerically less demanding and very suitable for fixed-point DSPs as they are employed today in automobiles.

4. Incorporation of adaptive grid-based look-up tables

The convergence rate of the Newton/FxLMS algorithm introduced in the preceding section is mainly determined by its step-size μ. A larger step-size generally leads to a reduced convergence time. However, the step-size is limited by stability constraints and a larger step-size reduces the algorithm's robustness to errors in the secondary path estimate (Elliott, 2001). In addition, Snyder and Tanaka (1993) showed that a large step-size may lead to an undesired amplification of uncorrelated signal components, passing from the error signals to the control outputs.

To overcome these drawbacks, this section introduces two control structures to improve the convergence and tracking behavior of the Newton/FxLMS algorithm while keeping its step-size parameter at moderate values. The proposed methods incorporate grid-based look-up tables in adaptive feedforward algorithms. The look-up tables are trained online in order to avoid any prior measurements.

In general, the weight adaptation in equation (6) starts with the initial condition $\mathbf{w}(0) = \mathbf{0}$. However, the convergence time could be significantly reduced if prior information of the final weight values was available. Engine vibrations arise from oscillating parts and the combustion process. It is a valid assumption that their amplitude and phase depend on the current engine speed and torque. Assuming that the engine vibration's amplitude and phase angle reach approximately constant values for steady-state engine operation, then the filter weight vector of the adaptive algorithm will also converge to constant values. By using the crankshaft angle $\varphi_{\mathrm{CS}}(n)$ instead of discrete integration of the engine's rotational frequency ω to generate the reference signal in equation (5) and (6), a fixed relationship between the reference signal $x(n)$ and the disturbing engine vibration $\mathbf{d}(n)$ is obtained. In this case, the steady-state filter weight vector values are reproducible for constant engine operation points.

The preceding assumptions are illustrated by Figure 6, which has been obtained in a full vehicle test rig measurement. The figure shows the activation of the left AEM in order to cancel the second engine order at a steady-state engine operating point in fifth gear with an engine torque $M_{\mathrm{eng}} = 200\,\mathrm{Nm}$ and an engine speed $n_{\mathrm{eng}} = 1500\,\mathrm{min}^{-1}$. The step-size μ of the Newton/FxLMS algorithm has been chosen intentionally low in order to suppress disturbances or short-time-scale fluctuations of the filter weights. As expected, the real and imaginary parts of the complex filter weight converge to constant values. Hence, it would be sufficient to determine the control signal's amplitude and phase angle for all engine operating points. Omitting the error sensors and the adaptation algorithm, these values could be applied to a map-based feedforward control algorithm. However, map-based feedforward control is unable to track any changes in the excitation spectrum and the transfer paths. Therefore, the subsequent section introduces methods to combine adaptive look-up tables with adaptive feedforward control.

4.1. Control structures

The relationship between the filter weights and the combustion process that has been described lays the foundations for the following control structures, which incorporate grid-based look-up tables in the Newton/FxLMS algorithm of Section 3. The independent dimensions of the two-dimensional look-up tables are the engine speed $n_{\mathrm{eng}}(n)$ and the engine torque $M_{\mathrm{eng}}(n)$. Since an estimate of the engine torque is generally available on the vehicle bus system, an additional torque sensor is unnecessary. The look-up tables contain the real and imaginary parts of the complex filter weight vector $\mathbf{w}(n)$ for compensating the vibrations of the particular engine order at the current engine operating point. In order to compensate for any changes in the transfer paths or engine disturbance during vehicle

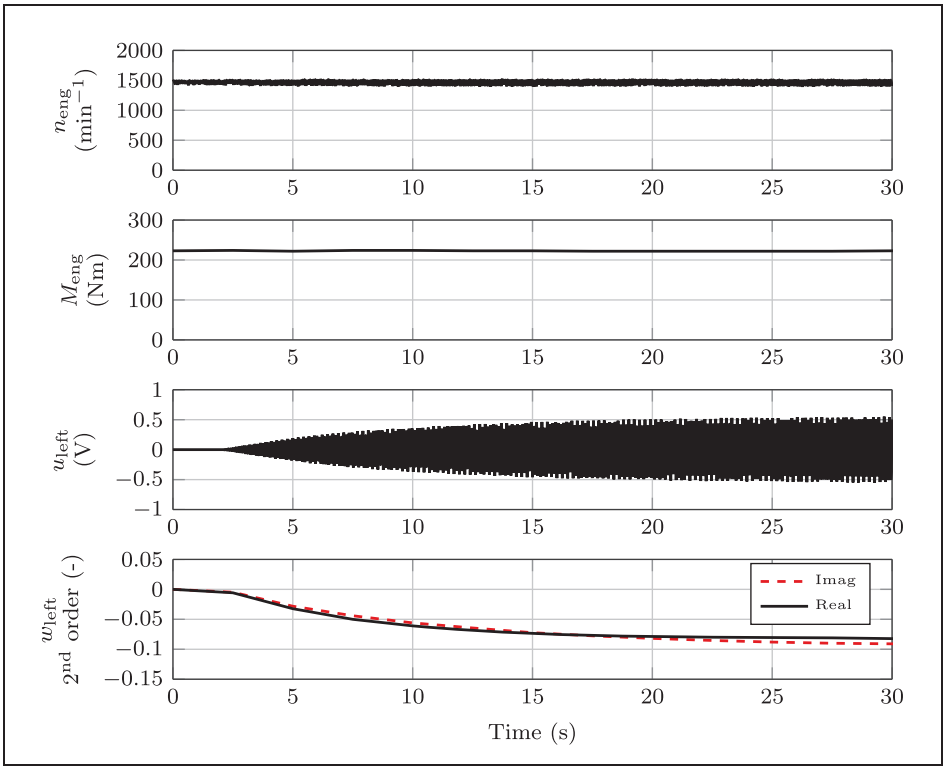

Figure 6. Time series of the real and imaginary part of the filter weight in order to compensate the second engine order at a steady-state engine operating point.

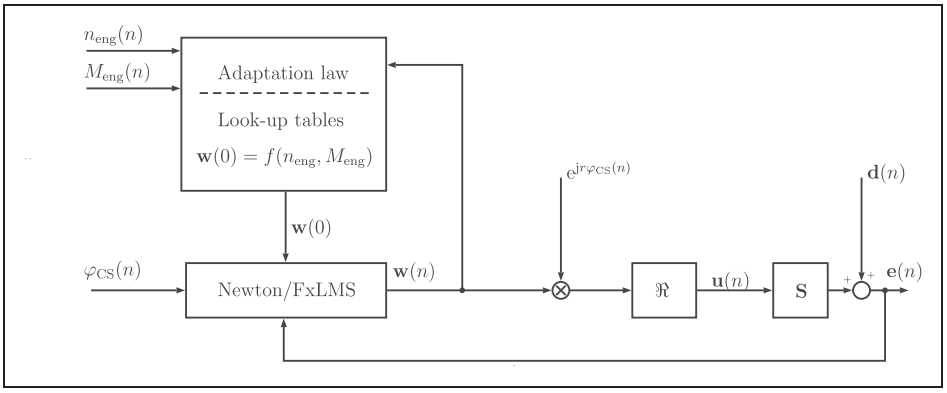

Figure 7. Newton/FxLMS algorithm with parameter-map.

operation, the look-up tables are extended by an adaptation process which is described in Section 4.2.

According to the classification of Heiss (1995), two control structures are possible for the present application. The control structure of Figure 7 has already been proposed in Hausberg et al. (2013). The look-up tables are parameter-maps, which store the real and imaginary parts of the complex filter weight vector as initial conditions for the adaptive feedforward control algorithm. Prior to the activation of the adaptive feedforward control algorithm, the initial conditions $\mathbf{w}(0)$ of the current engine operating point are passed from the look-up tables to the adaptive control algorithm. Since the look-up tables' information is only accessed before the activation of the adaptive feedforward algorithm, this control structure improves the convergence behavior while the tracking performance remains unchanged. During the operation of the adaptive feedforward control algorithm, the current complex filter weight vector is used as a training signal for

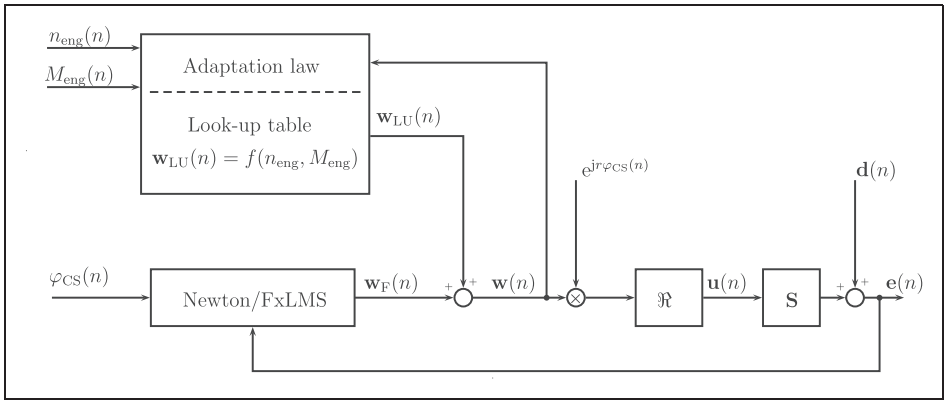

Figure 8. Newton/FxLMS algorithm with parallel-map.

the adaptation process of the look-up tables. In this control structure, map access and map adaptation are never carried out simultaneously.

An alternative structure to incorporate grid-based look-up tables into adaptive feedforward control is shown in Figure 8. The look-up table is applied in parallel with the adaptive algorithm and their respective complex outputs $\mathbf{w}_{\mathrm{LU}}(n)$ and $\mathbf{w}_{\mathrm{F}}(n)$ are summed to a superimposed complex filter weight $\mathbf{w}(n)$. The control signal is obtained by multiplication with the complex reference signal. This control structure is comparable to the method proposed in Shin (2007). However, in the present case the look-up tables are adaptable. The complex filter weight vector $\mathbf{w}(n)$ resulting from the overall superposition is used as a training signal for the adaptation of the look-up tables. Map access and map adaptation are carried out simultaneously in this control structure. As will be shown in Section 4.3, this is crucial for the choice of the step-size parameters of both adaptation algorithms. Map-based feedforward control supports the adaptive filter at switching points between full and half cylinder engine operation. Hence the convergence properties are improved. In addition, by the superposition of the look-up table output with the output of the adaptive filter, an improvement in the tracking behavior can be expected, e.g., during gear changes or fast acceleration. During steady-state engine operation, the adaptive filter compensates for modeling errors in the map-based feedforward control.

4.2. Online-adaptation of grid-based look-up tables

Grid-based look-up tables are the most common type of nonlinear static models in practice. In modern combustion engine control in particular, a vast number of one- and two-dimensional look-up tables is employed (Isermann, 2005). The two-dimensional look-up tables employed in this paper contain a set of scalar data

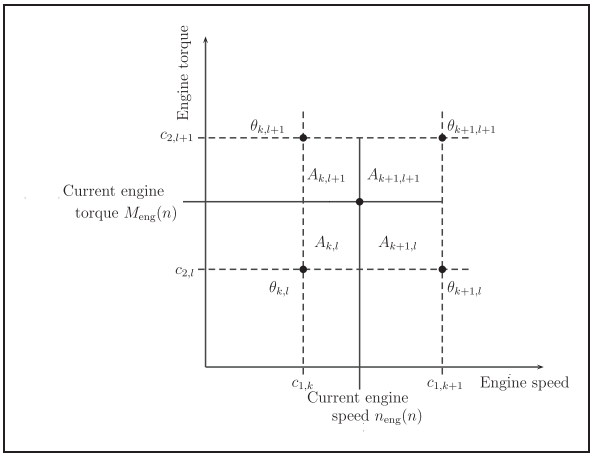

Figure 9. Interpolation within a two-dimensional look-up table (Schmitt, 1995; Vogt et al., 2004).

points approximating the nonlinear dependency between the complex filter weight vector $\mathbf{w}(n)$ and the look-up table's input signals, which are engine torque $M_{\mathrm{eng}}(n)$ and engine speed $n_{\mathrm{eng}}(n)$, respectively. For each engine mount and engine order to be canceled, two look-up tables are necessary: to store the real and imaginary part of the corresponding entry of $\mathbf{w}(n)$.

Figure 9 shows a sector of a look-up table in order to illustrate the method of calculating the table's output value for the current operating point, initially proposed in Schmitt (1995). After determining the four surrounding nodes of the current operating point, the table's output is calculated by weighting the data values at the interpolation nodes with the corresponding opposite areas $A_{k,l}$, $A_{k,l+1}$, $A_{k+1,l}$, $A_{k+1,l+1}$, divided by the total area A of one cell:

$$w_{LU} = \theta_{k,l}\frac{A_{k+1,l+1}}{A} + \theta_{k+1,l}\frac{A_{k,l+1}}{A} + \theta_{k,l+1}\frac{A_{k+1,l}}{A} + \theta_{k+1,l+1}\frac{A_{k,l}}{A} \tag{10}$$

with the areas

$$\begin{aligned} A_{k+1,l+1} &= (c_{1,k+1} - n_{eng}(n))(c_{2,l+1} - M_{eng}(n)) \\ A_{k,l+1} &= (n_{eng}(n) - c_{1,k})(c_{2,l+1} - M_{eng}(n)) \\ A_{k+1,l} &= (c_{1,k+1} - n_{eng}(n))(M_{eng}(n) - c_{2,l}) \\ A_{k,l} &= (n_{eng}(n) - c_{1,k})(M_{eng}(n) - c_{2,l}) \end{aligned} \tag{11}$$

The look-up table data values are adapted online according to the method described in Vogt et al. (2004). Converting equation (10) into a general basis function formulation leads to:

$$w_{LU} = \sum_{k=1}^{K} \sum_{l=1}^{L} \theta_{k,l} \Phi_{k,l}(\mathbf{y}, \mathbf{c}) \tag{12}$$

The vector of input signals (engine torque $M_{\mathrm{eng}}(n)$, engine speed $n_{\mathrm{eng}}(n)$) is expressed by $\mathbf{y}$ and the positions of the interpolation nodes $c_{1,1}$, $c_{1,k},\ldots,c_{2,1},c_{2,l},\ldots$ is described by $\mathbf{c}$. For each input vector, only four basis functions $\Phi_{k,l}(\mathbf{y},\mathbf{c})$ are nonzero. The vector of table data values at the interpolation nodes

$$\mathbf{v} = [v_1, v_2, \ldots, v_M]^T = \left[\theta_{1,1}, \theta_{1,2}, \ldots, \theta_{K,L}\right]^T \tag{13}$$

is adapted online with the normalized LMS (NLMS) algorithm (Vogt et al., 2004):

$$v_i(n+1) = v_i(n) + \mu_{LU} e_{LU}(n) \frac{\Phi_i(\mathbf{y}(n), \mathbf{c})}{\sum_{j=1}^{K \cdot L} \Phi_j^2(\mathbf{y}(n), \mathbf{c})}, \quad \text{for } i = 1, 2, \ldots, K \cdot L \tag{14}$$

The error $e_{\mathrm{LU}}(n)$ is expressed by

$$e_{LU}(n) = w_{LU}(n) - w(n), \tag{15}$$

where, according to equation (12), $w_{\mathrm{LU}}(n)$ is the look-up table's output at the current engine operating point. The corresponding correct value of the training signal is denoted by $w(n)$. The step-size μ_{LU} is in general different to the step-size μ of the adaptive feedforward algorithm in equation (6).

With respect to the additional computational effort of the map adaptation in equation (14), it should be noted that only the four data values immediately surrounding the current engine operation points are updated in each adaptation step. In the present application, the current engine operation point is the input vector $\mathbf{c}$ to all implemented data maps. Therefore, the calculation of the basis functions is independent of the number of maps and needs to be performed only once for each adaptation step. Finally, the map adaptation does not necessarily have to be carried out with the same sample rate as the adaptive feedforward algorithm.

The adaptation of the look-up tables is limited to steady-state phases of the combustion engine. This means that the engine speed and engine torque are constant for a certain amount of time. Hence, the current operating point remains in the same cell of the look-up tables. Equation (11) requires the determination of the four surrounding nodes of the current engine operating point. Therefore, in practice, a stationary phase can be identified if these surrounding nodes have not changed for a predefined amount of time. By limiting the map adaptation to steady-state phases of the combustion engine, it is ensured that the adaptive filter weights have actually reached their steady-state values before they are used for map-adaptation.

4.3. Choice of step-sizes

As a consequence of the incorporation of adaptive look-up tables in adaptive feedforward algorithms, two step-size parameters μ and μ_{LU} have to be chosen. This section describes some guidelines for the appropriate choice of these step-sizes. In addition the differences between the two proposed control structures are shown.

Both control structures employ the filter weight vector $\mathbf{w}(n)$ as a training signal for the map-adaptation. Since the look-up tables should only be adapted to long-term changes, temporary variations in the filter weight vector have to be suppressed. Generally, LMS algorithms become insensitive to disturbances through the choice of low step-sizes. However, a low value of μ degrades the convergence and tracking behavior of the vibration compensation algorithm. A step-size $\mu_{\mathrm{LU}} < 0.0005$ is chosen for the map-adaptation in order to suppress short-term variations of the training signal. As will be shown in Section 6.1, the resulting slow adaptation of the look-up table data proves to be unproblematic, since sufficiently complete look-up table data can be attained within short time periods. The step-size parameter of the adaptive Newton/FxLMS algorithm is chosen to be $\mu = 0.001$.

In contrast to the control structure of Figure 7, where map access and map adaptation are never carried out simultaneously, in the parallel structure of Figure 8 the step-size parameters μ and μ_{LU} cannot be chosen independently. This is illustrated in the simulation example of Figure 10 where the Newton/FxLMS algorithm with a parallel-map is employed to compensate a stationary single-frequency disturbance in a single-input-single-output (SISO) system. The figure shows the real and imaginary parts of filter weight $w_{\mathrm{F}}(n)$ and look-up table output $w_{\mathrm{LU}}(n)$, control signal $u(n)$ and

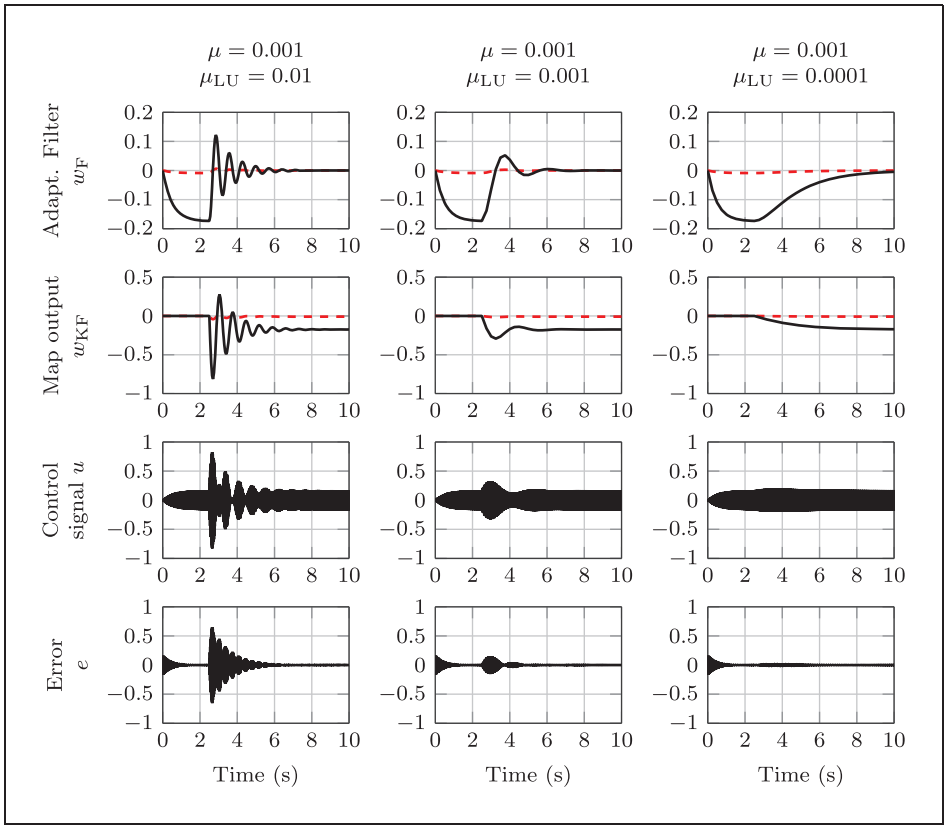

Figure 10. Comparison of the choice of step-sizes for the adaptive feedforward algorithm with parallel-map shown in Figure 8 (- -: real part; —: imaginary part).

error signal $e(n)$ for three different choices of μ and μ_{LU}. In each case, the filter weight of the adaptive feedforward algorithm converges to its steady-state value before the adaptation of the corresponding look-up table entry is activated. The simulation illustrates that the step-size μ_{LU} has to be chosen considerably lower than μ, in order to avoid any oscillating behavior resulting from the simultaneous map access and map adaptation.

5. Simulation

In this section, simulation results are presented to illustrate the control structures that have been introduced in the preceding section. The focus is on the extension with parameter-maps shown in Figure 7. Differences to the control structure with parallel-maps (Figure 8) are pointed out where appropriate. The simulations were performed in MATLAB/Simulink.

Figure 11 shows the simulation results in which the Newton/FxLMS algorithm with parameter-map has been used to cancel the second engine order in a SISO-system. First, the engine is in a steady-state with an engine torque of 175 Nm and an engine speed of 1500 min^{-1}. The relatively low engine load and speed allow the deactivation of half of the engine's cylinders at approximately 8 seconds. The resulting stronger vibration excitation requires the activation of the active engine mounts. Since the adaptation starts for the first time at the current operating point, no prior information about the complex filter weight's final value is available. A significantly long convergence time can be observed in the adaptation of the real and imaginary parts of the complex filter weight $w(n)$, the control signal $u(n)$ as well as the error signal $e(n)$. The error signal $e(n)$ has been simplified in this simulation example to a sinusoidal signal with the frequency of the second engine order and a constant amplitude.

Seven seconds after the cylinder deactivation, the engine is still in a steady-state and the adaptation of the look-up table begins. The complex output value $w_{LU}(n)$ of the look-up table converges to the current real and imaginary parts of the complex filter weight $w(n)$. As mentioned above, the step-size parameter μ_{LU} of the look-up table adaptation is chosen to be small to suppress measurement noise and small fluctuations of the complex filter weight. Thus, a considerable convergence time can be observed in the adaptation of the current look-up table output values. In contrast to an implementation of the look-up tables as parallel-maps,

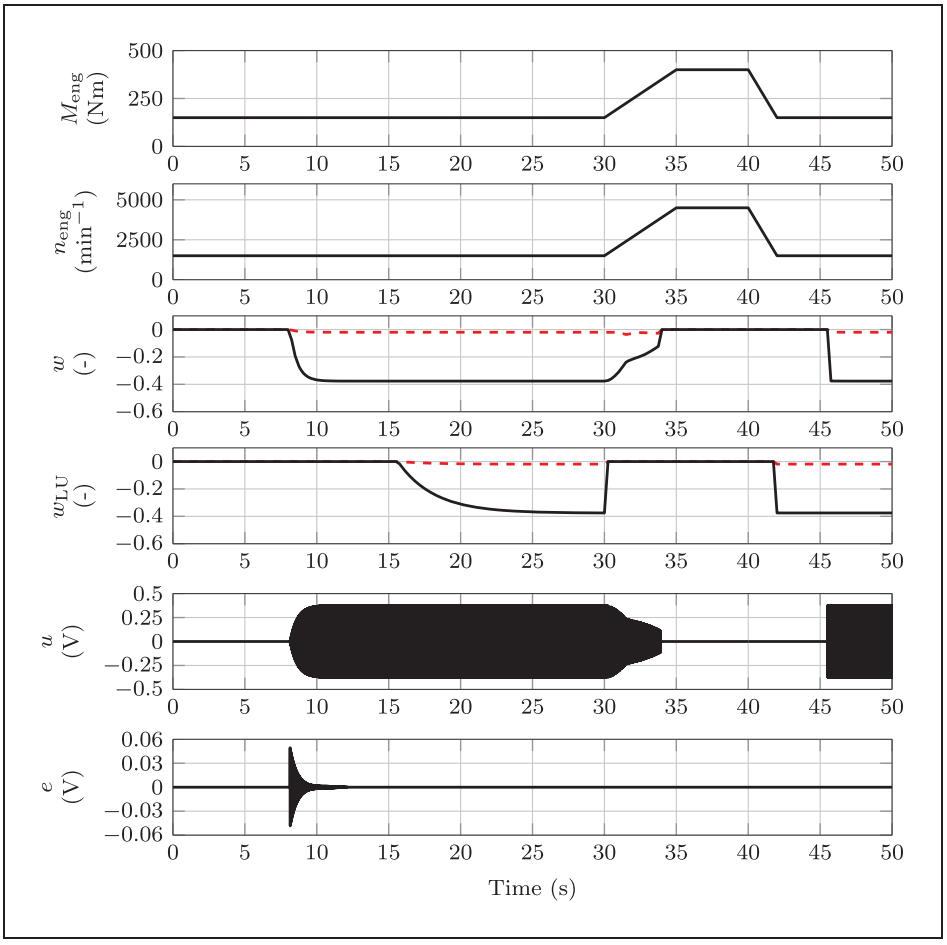

Figure 11. Simulation example to illustrate the proposed Newton/FxLMS algorithm with parameter-maps shown in Figure 7 (- -: real part; —: imaginary part).

in the incorporation of parameter-maps the complex filter weight $w(n)$ maintains its steady-state values even after the adaptation of the look-up tables. This is due to the fact that the map output $w_{\mathrm{LU}}(n)$ and the complex filter weight $w(n)$ are not superimposed. The look-up table output values are only passed as initial conditions $w(0)$ to the Newton/FxLMS algorithm.

After approximately 33 seconds, the engine leaves the COD-state since a higher engine speed and torque is required to accelerate the vehicle. Due to the reduced vibration excitation, the active engine mounts are deactivated at the same time. The error signal $e(n)$ remains zero, since the second engine order is no longer excited. The adaptation of the look-up table already stops at about 30 seconds when the engine starts to leave its steady-state. Following the completion of the acceleration process, the engine returns to its former steady-state operating point. Half of the cylinders are again deactivated. However, this time the previously trained initial conditions are provided by the look-up table. Thus, no convergence time can be observed in the complex filter weight $w(n)$ and the control signal $u(n)$ at about 46 seconds. The error signal $e(n)$ is immediately canceled and therefore remains zero. At this point, another difference between the implementation of parameter- and parallel-maps exists. Since the output values of the parameter-map are only used as initial conditions to the adaptive filter, the complex filter weight $w(n)$ still switches to its former steady-state value. In contrast, in the implementation of parallel-maps the complex filter weight will initially converge to a new steady-state value, which compensates the difference between the map output and the optimal filter. If a steady-state is detected again, the corresponding look-up table values are adapted again and the complex filter weight converges to zero (see also Figure 10).

6. Experimental results

Finally, in-vehicle tests have been carried out to evaluate the performance of the proposed control structures. First, the results of the online-adaptation of the look-up tables are shown. Afterwards, the improvements in convergence and tracking behavior are illustrated.

6.1. Online-adapted look-up tables

The map-adaptation of Section 4.2 has been tested in a 30 minute driving cycle of city traffic, country roads, and highways. Figure 12 shows the resulting online adapted look-up tables to control the left and right AEM in order to achieve the compensation of the dominant second engine order. Although the complex filter weights are stored as real and imaginary parts, their amplitude and phase are shown, in order to improve the physical interpretability. The highest control amplitude can be observed at high loads and low speeds, since in this area the engine excitation is at its highest level. The control signal amplitude decreases with decreasing engine load and increasing engine speed, respectively. The control signal's phase angle increases with increasing engine speed. A possible explanation is given by the decreasing phase angle of the secondary path shown in Figure 4 which has to be compensated by the control signal.

The results of Figure 12 demonstrate that using the step-size choice of Section 4.3, a sufficiently filled look-up table can already be attained after a short drive. Hence, neither prior parametrization nor offline measurements are necessary. Since not all operating points are trained equally often, the data maps show a partially irregular shape. However, with increased driving time an increasing smoothness is expected. The shape of the data maps is independent of the extension with parallel- or parameter-maps. The vibrational behavior of the combustion engine is mainly influenced by temperature variations and aging of engine components. These changes can be accounted for by continuous adaptation of the look-up tables. In this case, the look-up table values of a previous driving cycle can be employed as initial conditions for the next driving cycle.

6.2. Convergence behavior

It has been stated in Section 4.1 that both control structures are capable of reducing convergence time without a further increase of step-size parameter μ. The diagrams of Figure 13 compare the convergence behavior of the proposed methods with that of the Newton/FxLMS algorithm for different step-sizes. The time series of the left actuator's control signal (set to cancel the second engine order) is shown. The engine operates at a steady state of 1800 rpm and half of the cylinders are deactivated. The AEMs are activated manually at 0.25 seconds in each case.

The first diagram 13 (a) shows the time response for a step-size of $\mu = 0.0005$. As expected, a very slow convergence of the control signal, and thus the attained vibration compensation, can be observed. The next

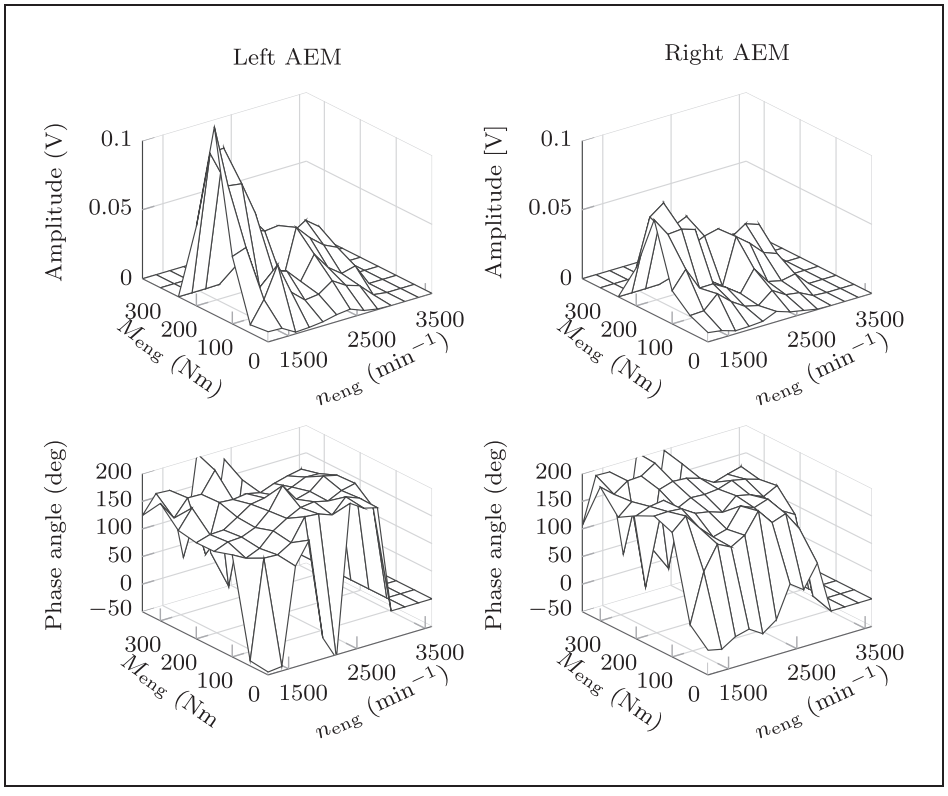

Figure 12. Online adapted map data in order to cancel the 2nd engine order.

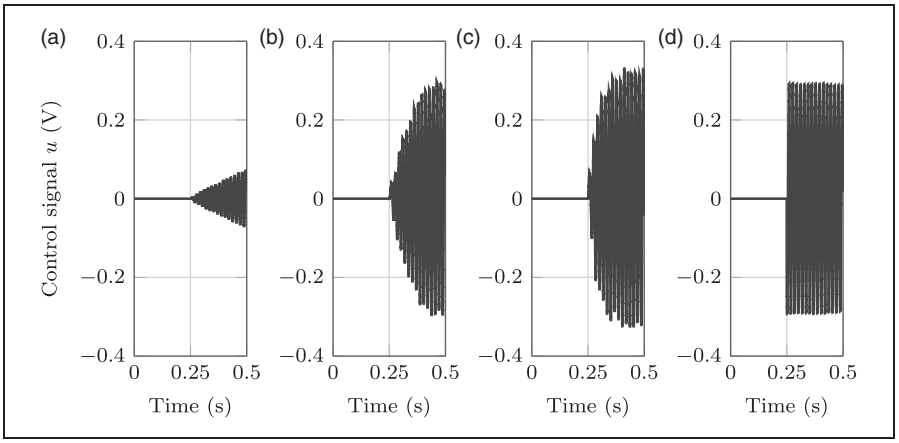

Figure 13. Convergence of the left actuator's control signal for the Newton/FxLMS in order to compensate the 2nd engine order after activation at a steady-state engine operating point. (a) $\mu = 0.0005$; (b) $\mu = 0.005$; (c) $\mu = 0.01$; (d) Extension with parallel- or parameter-map ($\mu = 0.001$).

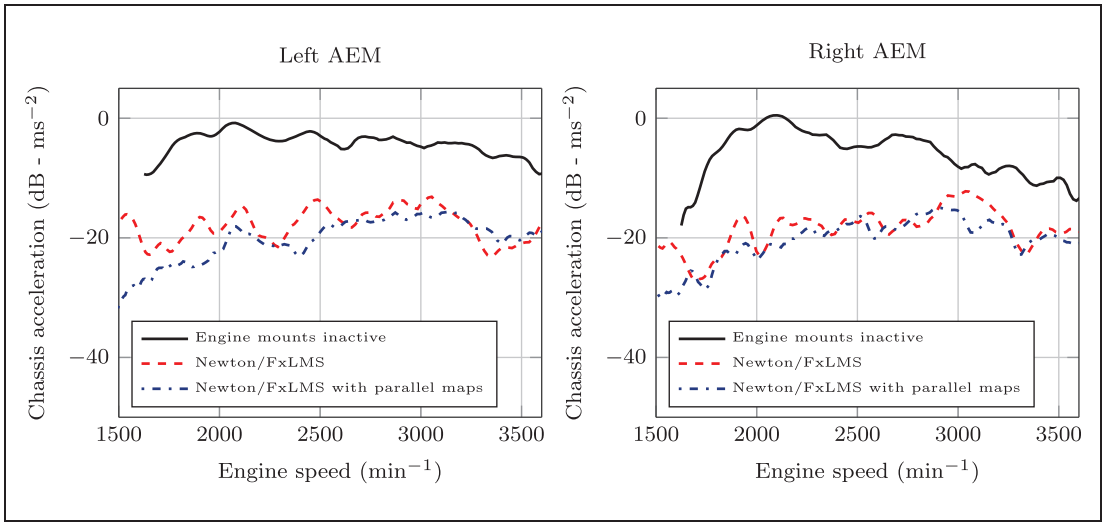

Figure 14. Extracts of the 2nd engine order of the acceleration at the left and right engine mount position.

diagram 13 (b) has been obtained by a tenfold increase of the step-size parameter to $\mu = 0.005$. The convergence time is reduced to approximately 0.25 seconds. Nevertheless, driving tests have shown that this threshold is still perceivable by the driver. The third diagram 13 (c) illustrates that another increase of the step-size to $\mu = 0.01$ results in a further reduction of the convergence time. However, the chosen step-size is very close to the experimentally identified stability limit, whereby the robustness of the algorithm is reduced. In addition, an uneven time series which is observed in the third diagram indicates that there are additional harmonic components present in the control signal. This can be explained by an increased out-of-band sensitivity of the implemented controller, which results from high step-size values (Snyder and Tanaka, 1993).

In the last diagram 13 (d), the convergence behavior of the proposed extension with parallel- or parameter-maps, respectively, is shown. Due to the superposition of the look-up tables' output with the complex filter weight in the case of parallel-maps, and accordingly the previously trained initial conditions in the case of parameter-maps, the convergence time is significantly reduced. The step-size parameter has been chosen to $\mu = 0.001$, which is sufficiently far from the stability limit.

6.3. Tracking behavior

Besides the described reduction of convergence time, the incorporation of parallel-maps in the Newton/FxLMS algorithm shown in Figure 8 promises improved tracking behavior. Generally, tracking

describes the adaptive filter's ability to track variations in the signal statistics. In the present application, these variations occur during changes of the engine speed leading to a modified frequency content of the reference signal $x(n)$ and the error signal vector $\mathbf{e}(n)$. To evaluate the tracking behavior of the Newton/FxLMS algorithm with parallel maps, fast engine run-ups under 3rd gear acceleration have been carried out.

Figure 14 shows the achieved compensation of the 2nd engine order for the Newton/FxLMS algorithm and its extension with parallel-maps. The step-size parameter of the adaptive feedforward algorithm has been set to $\mu = 0.001$ in both experiments. The vibration level with inactive AEMs is also shown, in order to assess the overall vibration reduction. Due to the improvement in tracking behavior, the extension of the Newton/FxLMS algorithm with parallel maps further reduces the vertical acceleration of the dominant second engine order at the left and right engine mount position across the whole frequency range. The extension with parameter-maps of Figure 7 has not been tested in this section, since the data map values are only passed as initial conditions to the adaptive filter. Therefore, there is no impact on the tracking behavior of the adaptive feedforward algorithm.

7. Conclusion

This contribution proposes methods to combine two commonly applied strategies to control AEMs, which are adaptive feedforward control and map-based feedforward control. Online adapted look-up tables were incorporated as parameter- or parallel-maps into a Newton/FxLMS adaptive feedforward control algorithm. In-vehicle experiments showed that sufficiently filled look-up tables can be obtained within a short driving time. Hence, no prior measurements are necessary in order to obtain appropriate map data. Once the look-up tables are adapted, both methods achieve a substantial reduction of the convergence time. In addition, an improvement of the tracking behavior during fast engine speed changes has been observed when implementing the adaptive feedforward control and the map-based feedforward control in parallel. The application of the proposed control structures is not limited to active engine mounts in passenger cars. They could also be employed in other narrowband active noise or vibration control systems, where combustion engine induced vibrations are canceled and fast convergence and tracking properties are required.

Acknowledgments

The authors gratefully acknowledge the technical and experimental support of Audi AG.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

References

Aoki K, Shikata T, Hyoudou Y, et al. (1999) Application of an active control mount (ACM) for improved diesel engine vehicle quietness. *SAE Paper* 1–6 (1999-01-0832).

Burgess J (1981) Active adaptive sound control in a duct: a computer simulation. *Journal of the Acoustical Society of America* 70(3): 715–726.

Elliott S (2001) *Signal Processing for Active Control*. London, San Diego: Academic Press.

Fakhari V and Ohadi A (2012) Robust control of automotive engine using active engine mount. *Journal of Vibration and Control* 19(7): 1024–1050.

Fakhari V, Ohadi A and Talebi H (2013) A robust adaptive control scheme for an active mount using a dynamic engine model. *Journal of Vibration and Control* Epub ahead of print 29 October 2013, doi: 10.1177/1077546313506927.

Fraanje R (2004) *Robust and fast schemes in broadband active noise and vibration control*. PhD Thesis, University of Twente.

Fursdon P (2006) The design and control of the Avon active engine mount. In: *Proceedings of the 4th styrian noise, vibration and harshness congress*, pp.1–10. Graz, Austria: SAE International.

Hausberg F, Scheiblegger C, Pfeffer P, et al. (2015) Experimental and analytical study of secondary path variations in active engine mounts. *Journal of Sound and Vibration* 340: 22–38.

Hausberg F, Vollmann S, Pfeffer P, et al. (2013) Improving the convergence behavior of active engine mounts in vehicles with cylinder-on-demand engines. In: *Proceedings of the 42nd international congress and exposition on noise control engineering (inter-noise)*, pp.5244–5252. Innsbruck, Austria.

Heiss M (1995) Kennfelder in der Regelungstechnik. *at-Automatisierungstechnik* 43(8): 363–367.

Hillis A (2011) Multi-input multi-output control of an automotive active engine mounting system. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 225(11): 1492–;1504.

Hillis A, Harrison A and Stoten D (2005) A comparison of two adaptive algorithms for the control of active engine mounts. *Journal of Sound and Vibration* 286(1): 37–54.

Hosseini AM, Arzanpour S, Golnaraghi M, et al. (2013) Solenoid actuator design and modeling with application in engine vibration isolators. *Journal of Vibration and Control* 19(7): 1015–1023.

Isermann R (2005) *Mechatronic Systems - Fundamentals*. London: Springer.

Johansson S (2000) *Active control of propeller-induced noise in aircraft*. PhD Thesis, Blekinge Institute of Technology.

Kuo SM and Morgan DR (1999) Active noise control: a tutorial review. *Proceedings of the IEEE* 87(6): 943–973.

Lee BH and Lee CW (2009) Model based feed-forward control of electromagnetic type active control engine-mount system. *Journal of Sound and Vibration* 323(3): 574–593.

Mansour H, Arzanpour S and Golnaraghi M (2010) Active decoupler hydraulic engine mount design with application to variable displacement engine. *Journal of Vibration and Control* 17(10): 1498–1508.

Mansour H, Arzanpour S and Golnaraghi M (2011) Design of a solenoid valve based active engine mount. *Journal of Vibration and Control* 18(8): 1221–1232.

Marzbani H, Jazar R and Fard M (2014) Hydraulic engine mounts: a survey. *Journal of Vibration and Control* 20(10): 1439–1463.

Matsuoka H, Mikasa T and Nemoto H (2004) NV countermeasure technology for a cylinder-on-demand engine – development of active control engine mount. *SAE Paper* 149–154 (2004-01-0413).

Morgan D (1980) An analysis of multiple correlation cancellation loops with a filter in the auxiliary path. *IEEE Transactions on Acoustics, Speech and Signal Processing* ASSP-28(4): 454–467.

Nakaji Y, Satoh S, Kimura T, et al. (1999) Development of an active control engine mount system. *Vehicle System Dynamics* 32(2–3): 185–198.

Riley B, Bodie M, Hoying J, et al. (1995) Active vibration and noise cancellation control of four cylinder engines – an application. *SAE Paper* 539–547 (951299).

Rupp M and Hausberg F (2014) LMS algorithmic variants in active noise and vibration control. In: *Proceedings of the 22nd European signal processing conference (EUSIPCO 2014)*, pp.691–695. Lisbon, Portugal: IEEE.

Sayed AH and Rupp M (1996) Error-energy bounds for adaptive gradient algorithms. *IEEE Transactions on Signal Processing* 44(8): 1982–1989.

Schmitt M (1995) *Untersuchungen zur Realisierung mehrdimensionaler lernfähiger Kennfelder in Großserien-Steuergeräten*. PhD Thesis, Technische Universität Darmstadt.

Shin KK (2007) Active vibration control of active fuel management engines using active engine mounts. *ASME international mechanical engineering congress and exposition*. pp. 27–32. Seattle, USA: ASME.

Snyder S and Tanaka N (1993) Modification to overall system response when using narrowband adaptive feedforward control systems. *Journal of Dynamic Systems, Measurement and Control* 115(4): 621–626.

Togashi C and Ichiryu K (2003) Study on hydraulic active engine mount. *SAE Paper* 2003-01-1418: 1–7.

Vogt M, Müller N and Isermann R (2004) On-line adaptation of grid-based look-up tables using a fast linear regression technique. *Journal of Dynamic Systems, Measurement and Control* 126(4): 732–739.

Widrow B, McCool J and Ball M (1975) The complex LMS algorithm. *Proceedings of the IEEE* 63: 719–720.

Widrow B, Shur D and Shaffer S (1981) On adaptive inverse control. In: *Proc. 15th ASILOMAR Conference on Circuits, Systems and Computers*, pp.185–195. Pacific Grove, USA: IEEE.

Yu Y, Naganathan N and Dukkipati R (2001) A literature review of automotive vehicle engine mounting systems. *Mechanism and Machine Theory* 36(1): 123–142.

Appendix

The following derivation relates the convergence properties of the FxLMS algorithm in equation (4) and the Newton/FxLMS algorithm in equation (6), which has been employed in this paper. First, the error equation (1) is written in vectorized form under the assumption of a time-invariant auxiliary path:

$$\mathbf{e}(n) = \mathbf{d}(n) + \mathbf{S}(r\omega)\mathbf{u}(n) \tag{16}$$

Imposing a reference system for the error vector leads to:

$$\mathbf{e}(n) = \mathbf{S}(r\omega)(\mathbf{u}(n) - \mathbf{u}_o(n)) + \mathbf{v}(n) \tag{17}$$

Herein, $\mathbf{v}(n)$ describes uncorrelated noise and $\mathbf{u}_o(n)$ is the control vector of a reference system that generates $\mathbf{d}(n)$. Note that $\mathbf{u}_o(n)$ is not necessarily identical to the steady-state solution $\mathbf{u}_{\text{opt}}$. With the further relation of the excitation $x(n)$ to $\mathbf{u}(n)$, equation (17) becomes

$$\mathbf{e}(n) = \mathbf{S}(r\omega)(\mathbf{w}(n) - \mathbf{w}_o(n))e^{jrn\omega T} + \mathbf{v}(n), \tag{18}$$

with the corresponding filter weight vector $\mathbf{w}_o(n)$ of the reference system. Here, the conjugate complex term in $\mathbf{w}(n) - \mathbf{w}_o(n)$ has been dropped, as explained in Rupp and Hausberg (2014).

First, the update equation (4) is considered, where the gradient is provided by an estimate of the auxiliary path filter $\hat{\mathbf{S}}(r\omega)$. Subsequently, effects of path estimation errors are neglected and it is assumed that the estimate $\hat{\mathbf{S}}(r\omega)$ is sufficiently close to the true value $\mathbf{S}(r\omega)$.

Introducing a parameter error vector

$$\tilde{\mathbf{w}}(n) = \mathbf{w}(n) - \mathbf{w}_o(n) \tag{19}$$

and substituting the error expression of equation (18) into the update equation (4) leads to

$$\begin{aligned}\tilde{\mathbf{w}}(n+1) = \tilde{\mathbf{w}}(n) &- \mu\hat{\mathbf{S}}^H(r\omega)\mathbf{S}(r\omega)\tilde{\mathbf{w}}(n) \\ &- \mu\hat{\mathbf{S}}^H(r\omega)e^{-jrn\omega T}\mathbf{v}(n)\end{aligned} \tag{20}$$

$$= \left(\mathbf{I} - \mu\hat{\mathbf{S}}^H(r\omega)\mathbf{S}(r\omega)\right)\tilde{\mathbf{w}}(n) - \mu\hat{\mathbf{S}}^H(r\omega)e^{-jrn\omega T}\mathbf{v}(n) \tag{21}$$

It is recognized here that if $\hat{\mathbf{S}} = \mathbf{S}$, the term $\mu\hat{\mathbf{S}}^H(r\omega)\mathbf{S}(r\omega)$ is positive definite and there exist step-sizes $\mu > 0$ for which the algorithm behaves stable and converges to the desired $\mathbf{w}_o$ (Sayed and Rupp, 1996). Its convergence speed is defined by the eigenvalue spread of $\hat{\mathbf{S}}^H(r\omega)\mathbf{S}(r\omega)$ (Elliott, 2001).

If we now replace $\hat{\mathbf{S}}^H(r\omega)$ by $\hat{\mathbf{S}}^{-1}(r\omega)$ in equation (6), we also recognize how the above update equation changes in terms of its parameter error vector:

$$\begin{aligned}\tilde{\mathbf{w}}(n+1) &= \tilde{\mathbf{w}}(n) - \mu\hat{\mathbf{S}}^{-1}(r\omega)\mathbf{S}(r\omega)\tilde{\mathbf{w}}(n) \\ &\quad - \mu\hat{\mathbf{S}}^{-1}(r\omega)e^{-jrn\omega T}\mathbf{v}(n) \qquad (22)\end{aligned}$$

$$= \left(\mathbf{I} - \mu\hat{\mathbf{S}}^{-1}(r\omega)\mathbf{S}(r\omega)\right)\tilde{\mathbf{w}}(n) - \mu\hat{\mathbf{S}}^{-1}(r\omega)e^{-jrn\omega T}\mathbf{v}(n) \qquad (23)$$

We have essentially the same type of algorithm, but now the eigenvalue spread of $\hat{\mathbf{S}}^{-1}(r\omega)\mathbf{S}(r\omega)$ can become one and thus takes on its smallest value, resulting in fastest convergence speed. However, such speed up is obtained at the expense of potentially larger noise influence as small eigenvalues of $\mathbf{S}(r\omega)$ now enhance the noise.