2024 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)

Affine Combination of General Adaptive Filters

Danqi Jin*, Yitong Chen†, Jie Chen†, Gongping Huang*
* School of Electronic Information, Wuhan University, Wuhan, China
E-mail: danqijin@mail.nwpu.edu.cn, gongpinghuang@whu.edu.cn
† CIAIC, Northwestern Polytechnical University, Xi'an, China
E-mail: chenyitong@mail.nwpu.edu.cn, dr.jie.chen@ieee.org

***Abstract*—Combination schemes are very effective for adaptive filters; yet, they are still constrained to use a linear model and the mean-square error (MSE) cost. All types of models, both linear and nonlinear, are accommodated in this work by expanding combination schemes to generic cost functions. Specifically, in this kind of combination framework, two candidate filters are firstly designed by optimizing separate general cost functions; then, to optimize overall performance, the combination coefficients that are given to these filters are fine-tuned by minimizing a third general cost function. Separately, we set and optimize each of the three generic cost functions. After that, we design an affine combination scheme by using the stochastic sign-gradient descent to adjust combination coefficients. Also, to enhance its performance, we include a weight-copying trick. Finally, simulation results are provided to validate their effectiveness for both the MSE and the logistic risk cost functions.**

***Index Terms*—Adaptive filter, combination scheme, nonlinear model, general cost function, weight-copying trick.**

I. INTRODUCTION

Adaptive filters are powerful tools for online parameter estimation from streaming data, and typical algorithms include the least-mean-square (LMS), the recursive least-square (RLS), the Kalman filter, and the affine projection algorithm (APA) [1], [2], [3], [4], [5]. Speech signal processing [6], [7], radar signal processing [8], earthquake detection [9], and many other applications have made extensive use of these adaptive filters.

Combination strategies have been suggested and implemented for adaptive filters [10], [11], [12] and distributed adaptive networks [13], [14], [15] with great success in the last few decades. In these studies, it is demonstrated that filters may be created by combining the benefits of all candidate filters via convex or affine combinations. Most often, these combination schemes are employed to make it easier to choose filter settings, to make it more resilient to an unknown environment, or to potentially improve performance beyond what any one filter could do [16], [17]. Acoustic echo cancellation [18], [19], beamforming [20], [21], and speech dereverberation [22], [23] are only a few examples of the actual applications where combination techniques have demonstrated promising results in providing superior solutions.

Use cases like online logistic learning from nonlinear data [24] have prompted the need for combination of general adaptive filters that follow nonlinear models. The majority of the research on adaptive filter combination strategies is devoted to linear models using the MSE cost function [25]. Although some studies have taken into account the combination of nonlinear filters in more general nonlinear models, these models are suboptimal since their combination coefficients are set by minimizing the MSE cost [18], [19]. For this reason, developing a method to integrate nonlinear models with generic adaptive filters is crucial.

In this work, we propose a strategy for combining two general adaptive filters to handle this problem, and we do it by using streaming data that could adhere to nonlinear models. Two candidate filters are built at the adaptive filter layer by minimizing various general cost functions. In order to optimize the overall performance, the combination layer uses the third general cost function to adjust combination coefficients. Every one of the three generic cost functions may be fine-tuned on its own. After that, we use the stochastic sign-gradient descent to design an affine combination scheme. Its performance is further improved by including a weight-copying trick. Finally, their effectiveness has been validated by simulation results. Before concluding this section and proceeding to the next, we provide the notation used throughout this work.

Notation. All scalars are represented by the regular typeface x with the exception of $F^{(i)}$, which indicates the i-th candidate filter. Column vectors are denoted by boldface little letters $\boldsymbol{x}$. The superscript $(\cdot)^\top$ denotes the transpose operator. The mathematical expectation is denoted by $\mathbb{E}\{\cdot\}$. The $\mathrm{mod}(\cdot,\cdot)$ function returns the remainder after division.

II. GENERAL ADAPTIVE FILTERS

Consider the following real-valued, strongly convex cost functions $J^{(i)}(\boldsymbol{w}^{(i)})$ for $i = 1, 2$, representing the expectation of loss functions $Q^{(i)}_{n+1}(\boldsymbol{w}^{(i)}; \boldsymbol{\chi}_{n+1})$:

$$J^{(i)}(\boldsymbol{w}^{(i)}) \triangleq \mathbb{E}\big\{Q^{(i)}_{n+1}(\boldsymbol{w}^{(i)}; \boldsymbol{\chi}_{n+1})\big\}, \tag{1}$$

where the expectation is assessed using the distribution of random data $\boldsymbol{\chi}_{n+1}$, which could adhere to nonlinear models. The subscript $n+1$ indicates the current time instant, and $\boldsymbol{w}^{(i)} \in \mathbb{R}^L$ is a real-valued parameter vector.

By iteratively minimizing $J^{(i)}(\boldsymbol{w}^{(i)})$ with $i = 1, 2$ over iterations, we may obtain two generic adaptive filters $F^{(1)}$ and $F^{(2)}$, respectively:

$$\boldsymbol{w}^{(i)}_{n+1} = f^{(i)}\Big\{\boldsymbol{w}^{(i)}_n, Q^{(i)}_{n+1}\big(\boldsymbol{w}^{(i)}_n; \boldsymbol{\chi}_{n+1}\big), \mu^{(i)}_{n+1}, \boldsymbol{\beta}^{(i)}_{n+1}\Big\} \tag{2}$$

where the generic update formula for the adaptive filter $F^{(i)}$ is denoted by $f^{(i)}\{\cdot\}$, the non-negative step-size parameter is denoted by $\mu^{(i)}_{n+1}$, and the state vector $\boldsymbol{\beta}^{(i)}_{n+1}$ integrates any extra

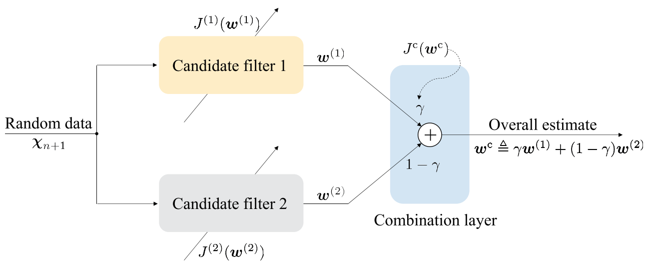

Fig. 1. Illustration of the combination framework for two general adaptive filters.

information necessary for filter adaptation. Also, take notice that in (2) we have used an adaptive technique to approximatively find the unavailable expectation term $\mathbb{E}\{Q^{(i)}_{n+1}(\cdot;\boldsymbol{\chi}_{n+1})\}$ by introducing the stochastic quantity $Q^{(i)}_{n+1}(\cdot;\boldsymbol{\chi}_{n+1})$.

III. AN AFFINE COMBINATION SCHEME

A. A combination framework for general adaptive filters

Fig. 1 shows the architecture of the combination framework for two general adaptive filters, which includes a combination layer and an adaptive filter layer that operate in tandem with one another. Two separate adaptive filters, $F^{(1)}$ and $F^{(2)}$, are considered for the adaptive filter layer; each is defined by an update equation (2). Each of these filters estimates the associated optimum weight vectors under cost functions $J^{(i)}(\boldsymbol{w}^{(i)})$ using the same data set $\boldsymbol{\chi}_{n+1}$. Then, in the combination layer, we link candidate filters $F^{(1)}$ and $F^{(2)}$ with combination coefficients γ and $1-\gamma$, respectively.

The goal of the combination layer is to learn which adaptive filter works better at each time instant, by adjusting γ in order to optimize the criterion $J^{\mathrm{c}}(\boldsymbol{w}^{\mathrm{c}})$ defined as:

$$J^{\mathrm{c}}(\boldsymbol{w}^{\mathrm{c}}) \triangleq \mathbb{E}\big\{Q^{\mathrm{c}}_{n+1}(\boldsymbol{w}^{\mathrm{c}};\boldsymbol{\chi}_{n+1})\big\}, \tag{3}$$

where $Q^{\mathrm{c}}_{n+1}(\cdot;\boldsymbol{\chi}_{n+1})$ is a strongly convex loss function, and $\boldsymbol{w}^{\mathrm{c}} \triangleq \gamma\boldsymbol{w}^{(1)}+(1-\gamma)\boldsymbol{w}^{(2)}$, with the superscript c highlighting that these are quantities at the combination layer. Please take note that in $J^{\mathrm{c}}(\boldsymbol{w}^{\mathrm{c}})$, the optimization variable is γ alone, whereas $\boldsymbol{w}^{(1)}$ and $\boldsymbol{w}^{(2)}$ are estimates of candidate filters. It is also possible to establish and optimize each of the three generic cost functions, namely $J^{(i)}(\boldsymbol{w}^{(i)})$ for $i=1,2$ and $J^{\mathrm{c}}(\boldsymbol{w}^{\mathrm{c}})$, independently. Determining an approach to assess γ by minimizing cost function (3), given $\boldsymbol{w}^{(1)}$ and $\boldsymbol{w}^{(2)}$, becomes the subsequent task.

B. An affine sign-gradient scheme

We propose that γ be assessed by attempting to minimize (3) using the values $\boldsymbol{w}^{(1)}_n$ and $\boldsymbol{w}^{(2)}_n$, that is:

$$\gamma^\star_{n+1} = \mathrm{argmin}_\gamma\ J^{\mathrm{c}}\big(\gamma\boldsymbol{w}^{(1)}_n + (1-\gamma)\boldsymbol{w}^{(2)}_n\big). \tag{4}$$

We may determine that the best solution $\gamma^\star_{n+1}$ meets the above requirement by setting the derivative of $J^{\mathrm{c}}\big(\gamma\boldsymbol{w}^{(1)}_n + (1-\gamma)\boldsymbol{w}^{(2)}_n\big)$ with respect to γ to zero:

$$\mathbb{E}\Big\{\nabla_\gamma Q^{\mathrm{c}}\big(\gamma^\star_n\boldsymbol{w}^{(1)}_n + (1-\gamma^\star_n)\boldsymbol{w}^{(2)}_n;\boldsymbol{\chi}_{n+1}\big)\Big\} = 0, \tag{5}$$

where the derivative of the function $Q^{\mathrm{c}}(\cdot)$ with respect to γ is represented by $\nabla_\gamma Q^{\mathrm{c}}(\cdot)$. Because it necessitates knowledge of statistics, evaluating $\gamma^\star_{n+1}$ using (5) is not feasible.

We provide an adaptive strategy to solve this challenge. One must first determine an estimate at the combination layer and time instant $n+1$ before proceeding with this task:

$$\boldsymbol{w}^{\mathrm{c}}_{n+1} \triangleq \big[\gamma\boldsymbol{w}^{(1)}_{n+1} + (1-\gamma)\boldsymbol{w}^{(2)}_{n+1}\big]\Big|_{\gamma=\gamma_{n+1}}, \tag{6}$$

where γ_{n+1} is an estimate of γ at time instant $n+1$.

By minimizing $J^{\mathrm{c}}\big(\gamma\boldsymbol{w}^{(1)}_n + (1-\gamma)\boldsymbol{w}^{(2)}_n\big)$ in the direction of negative gradient with respect to γ and estimating the expectation terms with instantaneous values, the following affine sign-gradient scheme is obtained:

$$\begin{aligned}\gamma_{n+1} &= \gamma_n - \mu_\gamma \mathrm{sgn}\Big\{\nabla_\gamma J^{\mathrm{c}}\big(\gamma\boldsymbol{w}^{(1)}_n + (1-\gamma)\boldsymbol{w}^{(2)}_n\big)\Big\}\Big|_{\gamma=\gamma_n}\\ &\approx \gamma_n - \mu_\gamma \mathrm{sgn}\big\{\nabla^\top_{\boldsymbol{w}^{\mathrm{c}}} Q^{\mathrm{c}}_{n+1}(\boldsymbol{w}^{\mathrm{c}}_n;\boldsymbol{\chi}_{n+1})\cdot(\boldsymbol{w}^{(1)}_n - \boldsymbol{w}^{(2)}_n)\big\} \end{aligned}\tag{7}$$

where $\mathrm{sgn}\{\cdot\}$ is the sign function, and μ_γ is a positive step-size. Different from the affine combination of two LMS filters for which the power normalization scheme [16] and the sign regressor scheme have been introduced [11], to make it more resistant to changes in data $\boldsymbol{\chi}_{n+1}$ for the combination of two general adaptive filters, the sign gradient descent method is proposed in this paper, and equation (7) is derived by using the chain rule of derivatives.

C. A weight-copying trick

The weight-copying trick, first suggested in [26], is used to the combination of two generic adaptive filters to further improve its performance. To be more precise, at every time instant $n+1$, the parameter γ_{n+1}, which can be found using equation (7), shows which candidate filter performs better locally. As a result, the poorest candidate filter can take the weight vector from the best one. The weight-copying trick writes to:

$$\boldsymbol{w}^{(1)}_{n+1} = \begin{cases}\boldsymbol{w}^{(2)}_{n+1}, & \text{if } \gamma_{n+1}<1-\beta \text{ and } \mathrm{mod}(n+1,N)==0\\ \boldsymbol{w}^{(1)}_{n+1}, & \text{otherwise}\end{cases} \tag{8}$$

and

$$\boldsymbol{w}^{(2)}_{n+1} = \begin{cases}\boldsymbol{w}^{(1)}_{n+1}, & \text{if } \gamma_{n+1}>\beta \text{ and } \mathrm{mod}(n+1,N)==0\\ \boldsymbol{w}^{(2)}_{n+1}, & \text{otherwise,}\end{cases} \tag{9}$$

where $0 \ll \beta < 1$ is a pre-defined threshold value with typical value 0.9, notation "==" denotes the logical equivalence operator, and a free integer $N \geq 2$. In equations (8) – (9), conditions $\gamma_{n+1} > \beta$ and $\gamma_{n+1} < 1-\beta$ imply that the weight-copying can occur when one candidate filter greatly outperforms another, and condition $\mathrm{mod}(n+1,N)=0$ means that this trick is used periodically with period $N \geq 2$. The last step is to execute the weight-copying trick after evaluating (6). The findings will be employed in the next iteration.

For the ease of reference, the proposed algorithm is presented in Algorithm 1.

Algorithm 1 The proposed affine sign-gradient scheme with a weight-copying trick

Initialization: Set step-sizes $\mu_{n+1}^{(i)}$ and μ_γ, set integer $N \geq 2$ and $0 \ll \beta < 1$, initialize $\boldsymbol{w}_0^{(1)}$, $\boldsymbol{w}_0^{(2)}$ and γ_0; evaluate $\boldsymbol{w}_0^{\mathrm{c}} = \gamma_0 \boldsymbol{w}_0^{(1)} + (1-\gamma_0)\boldsymbol{w}_0^{(2)}$.

Iteration: Repeat the following steps for $n = 0, 1, 2 \cdots$:

1) Evaluate two generic adaptive filters $F^{(1)}$ and $F^{(2)}$ by using (2) for $i = 1, 2$.
2) Evaluate the combination coefficient γ_{n+1} by using (7).
3) Evaluate the estimate $\boldsymbol{w}_{n+1}^{\mathrm{c}}$ by using (6).
4) If conditions $\mod (n+1, N) == 0$ and $\gamma_{n+1} < 1 - \beta$ are satisfied simultaneously, then evaluate $\boldsymbol{w}_{n+1}^{(1)} = \boldsymbol{w}_{n+1}^{(2)}$.
5) If conditions $\mod (n+1, N) == 0$ and $\gamma_{n+1} > \beta$ are satisfied simultaneously, then evaluate $\boldsymbol{w}_{n+1}^{(2)} = \boldsymbol{w}_{n+1}^{(1)}$.

Output: Vectors $\boldsymbol{w}_{n+1}^{\mathrm{c}}$ for all n.

IV. SIMULATION RESULTS

In this section, we present simulation results to illustrate the proposed affine sign-gradient scheme and the weight-copying trick. All simulated curves were averaged over 100 Monte Carlo runs.

A. Combination of two sparsity-promoting adaptive filters

In the first experiment, we considered the following linear model as a special case:

$$y_{n+1} = \boldsymbol{x}_{n+1}^\top \boldsymbol{w}^\star + s_{n+1}, \tag{10}$$

in order to ensure that our proposed scheme works for linear model. A regression vector $\boldsymbol{x}_{n+1} \in \mathbb{R}^{35}$ was generated from a zero-mean Gaussian distribution with a unit covariance matrix, and an unknown sparse parameter vector $\boldsymbol{w}^\star \in \mathbb{R}^{35}$ was specified in equation (10). The additive noise s_{n+1} was also an i.i.d. zero-mean white Gaussian noise with variance $\sigma_s^2 = 0.01$. For candidate filters $F^{(1)}$ and $F^{(2)}$, two distinct cost functions were taken into account, specifically:

$$J^{(1)}(\boldsymbol{w}^{(1)}) = J_{\mathrm{MSE}}\big(\boldsymbol{w}^{(1)}\big) + \lambda \sum_{i=1}^{35} \log\Big(1 + |w_i^{(1)}|/\varepsilon\Big) \tag{11}$$

$$J^{(2)}(\boldsymbol{w}^{(2)}) = J_{\mathrm{MSE}}\big(\boldsymbol{w}^{(2)}\big) + \lambda \sum_{p=1}^{7} \log\Big[1 + \frac{\|\boldsymbol{w}_{\mathcal{G}_p}^{(2)}\|_2}{\varepsilon}\Big] \tag{12}$$

where $\lambda \geq 0$ was the regularization parameter, $\varepsilon > 0$ was a parameter for reweighted penalty, $w_i^{(1)}$ was the i-th entry of vector $\boldsymbol{w}^{(1)}$, $\boldsymbol{w}_{\mathcal{G}_p}^{(2)} \in \mathbb{R}^5$ denoted a subvector of $\boldsymbol{w}^{(2)}$ with entries indexed by $\mathcal{G}_p$, $\{\mathcal{G}_p\}_{p=1}^7$ was a partition of the index set $\mathcal{G} \triangleq \{1, \ldots, 35\}$, and $J_{\mathrm{MSE}}(\cdot)$ was the MSE cost defined as:

$$\begin{aligned} J_{\mathrm{MSE}}(\boldsymbol{w}) &\triangleq \mathbb{E}\big\{Q_{n+1}(\boldsymbol{w}; \boldsymbol{\chi}_{n+1})\big\} \\ &= \frac{1}{2}\,\mathbb{E}\big\{\|y_{n+1} - \boldsymbol{w}^\top \boldsymbol{x}_{n+1}\|^2\big\}, \end{aligned} \tag{13}$$

with quantities $\{y_{n+1}, \boldsymbol{x}_{n+1}\}$ standing for the random data $\boldsymbol{\chi}_{n+1}$. By using the stochastic subgradient descent to minimize (11) and (12), RZA-LMS algorithm and GRZA-LMS algorithm for sparse system identification had been derived in [27], [28], [29], [30] as:

$$\begin{aligned} \boldsymbol{w}_{n+1}^{(1)} &= \boldsymbol{w}_n^{(1)} + \mu^{(1)}\,(y_{n+1} - \boldsymbol{w}_n^{(1)\top}\boldsymbol{x}_{n+1})\boldsymbol{x}_{n+1} \\ &\quad - \lambda \mu^{(1)}\,\frac{\mathrm{sgn}\{\boldsymbol{w}_n^{(1)}\}}{\boldsymbol{\varepsilon} + |\boldsymbol{w}_n^{(1)}|} \end{aligned} \tag{14}$$

$$\begin{aligned} \boldsymbol{w}_{n+1,\mathcal{G}_p}^{(2)} &= \boldsymbol{w}_{n,\mathcal{G}_p}^{(2)} + \mu^{(2)}\,(y_{n+1} - \boldsymbol{w}_n^{(2)\top}\boldsymbol{x}_{n+1})\boldsymbol{x}_{n+1,\mathcal{G}_p} \\ &\quad - \lambda\mu^{(2)}\,\beta_{n,p}\boldsymbol{v}_{n,\mathcal{G}_p}, \quad \forall p = 1, 2, \cdots, 7 \end{aligned} \tag{15}$$

respectively, where the weighting coefficient $\beta_{n,p}$ was defined as:

$$\beta_{n,p} \triangleq \frac{1}{\|\boldsymbol{w}_{n,\mathcal{G}_p}^{(2)}\|_2 + \varepsilon} \tag{16}$$

subvector $\boldsymbol{v}_{n,\mathcal{G}_p}$ was defined as:

$$\boldsymbol{v}_{n,\mathcal{G}_p} = \begin{cases} \frac{\boldsymbol{w}_{n,\mathcal{G}_p}}{\|\boldsymbol{w}_{n,\mathcal{G}_p}\|_2} & \text{when } \|\boldsymbol{w}_{n,\mathcal{G}_p}\|_2 \neq 0 \\ 0 & \text{when } \|\boldsymbol{w}_{n,\mathcal{G}_p}\|_2 = 0, \end{cases} \tag{17}$$

and $\boldsymbol{x}_{n+1,\mathcal{G}_p} \in \mathbb{R}^5$ denoted a subvector of $\boldsymbol{x}_{n+1}$ with entries indexed by $\mathcal{G}_p$. Note that the division operator and the absolute value operator $|\cdot|$ in the third term on right-hand-side of (14) were applied in an element-wise manner, and $\boldsymbol{\varepsilon} \in \mathbb{R}^{35}$ was a vector with each entry being the scalar ε. Filters (14) and (15) were used as candidate filters $F^{(1)}$ and $F^{(2)}$, respectively, with step-sizes being 0.003 and 0.01. Regarding the combination layer, we determined that $J^{\mathrm{c}}(\boldsymbol{w}^{\mathrm{c}}) = J_{\mathrm{MSE}}(\boldsymbol{w}^{\mathrm{c}})$ and $\mu_\gamma = 0.015$. These values were then used to estimate γ_{n+1} by substituting them into equation (7), leading to:

$$\begin{aligned} &\gamma_{n+1} \approx \gamma_n + \\ &\mu_\gamma \mathrm{sgn}\big\{(y_{n+1} - \boldsymbol{x}_{n+1}^\top \boldsymbol{w}_n^{\mathrm{c}}) \cdot \boldsymbol{x}_{n+1}^\top(\boldsymbol{w}_n^{(1)} - \boldsymbol{w}_n^{(2)})\big\}. \end{aligned} \tag{18}$$

Fig. 2(a) displays the learning curves for two candidate filters, the affine sign-gradient scheme, and the weight-copying trick with respect to mean-square deviation (MSD). A better steady-state performance is associated with the RZA-LMS filter, but the GRZA-LMS filter exhibits a quicker initial convergence rate. The affine sign-gradient scheme proves its efficacy in the special case of linear model by integrating the RZA-LMS and GRZA-LMS algorithms to produce a learning curve with a lower steady-state MSD and a quicker initial convergence rate. In addition, the affine sign-gradient scheme can benefit from the weight-copying trick, which can enhance its convergence rate. Once again, the efficiency of the suggested affine sign-gradient scheme is validated by the evolution of parameter γ_n in Fig. 2(b).

B. Combination of two logistic regression learners

We took a nonlinear model into account in the second experiment. Assume that $\boldsymbol{x}_{n+1} \in \mathbb{R}^{15}$ was a real-valued random vector and that z_{n+1} was a streaming series of binary random variables with values of $+1$ or -1. In a machine learning

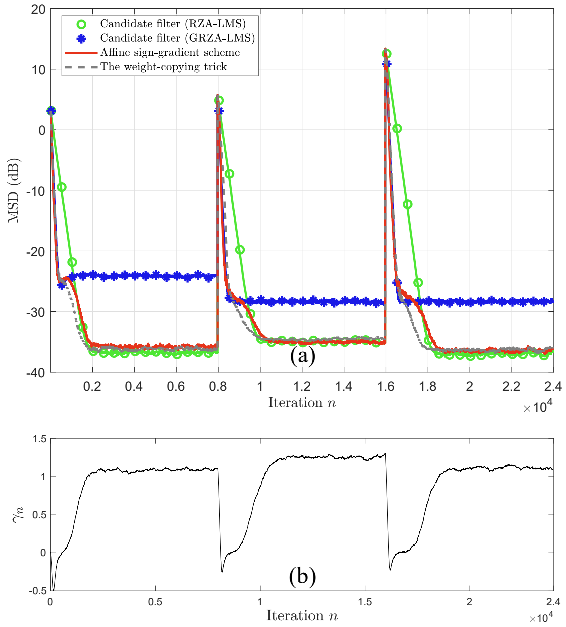

Fig. 2. Simulation results of the affine sign-gradient scheme and the weight-copying trick. (a) MSD learning curves and (b) evolution of combination coefficient γ_n for the affine sign-gradient scheme.

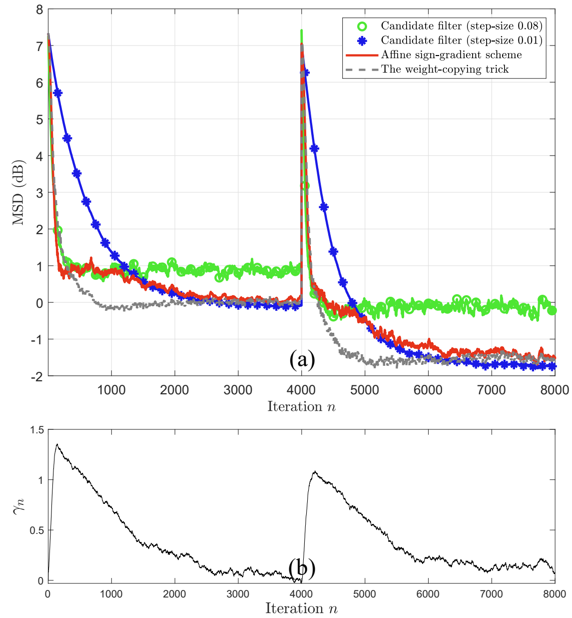

Fig. 3. Simulation results of the affine sign-gradient scheme and the weight-copying trick. (a) MSD learning curves and (b) evolution of combination coefficient γ_n for the affine sign-gradient scheme.

setting, the class to which feature vector $\boldsymbol{x}_{n+1}$ belonged was denoted by variable z_{n+1}. In these problems, we sought a vector $\boldsymbol{w}^\star$ that minimized the following regularized, strongly convex logistic risk function:

$$J_{\text{logis}}(\boldsymbol{w}) \triangleq \mathbb{E}\{Q_{\text{logis}}(\boldsymbol{w}; \boldsymbol{\chi}_{n+1})\}, \tag{19}$$

with:

$$Q_{\text{logis}}(\boldsymbol{w}; \boldsymbol{\chi}_{n+1}) \triangleq \frac{\rho}{2} \cdot \|\boldsymbol{w}\|_2^2 + \ln\big(1 + e^{-z_{n+1}\boldsymbol{x}_{n+1}^\top \boldsymbol{w}}\big), \tag{20}$$

where $\rho > 0$ was the regularization parameter, notations $\ln(\cdot)$ and $e^{(\cdot)}$ were the logarithmic and exponential functions, respectively, and the combined quantities $\{z_{n+1}, \boldsymbol{x}_{n+1}\}$ represented the random variable $\boldsymbol{\chi}_{n+1}$ appeared in (19). Usually, data $\{z_{n+1}, \boldsymbol{x}_{n+1}\}$ were generated under an underlying optimal vector $\boldsymbol{w}^\star \in \mathbb{R}^{15}$ in a nonlinear fashion [24]. Additionally, we assumed a non-stationary environment in this experiment, which meant that $\boldsymbol{w}^\star$ may take on two different values at time instants $n \in [0, 3999]$ and $n \in [4000, 7999]$. We examined the merging of two logistic regression learners, $F^{(1)}$ and $F^{(2)}$, to verify the affine sign-gradient scheme. In particular, the stochastic gradient descent approach was used to minimize (19) in both $F^{(1)}$ and $F^{(2)}$. We could easily get to:

$$\boldsymbol{w}_{n+1}^{(i)} = \big(1 - \mu_{n+1}^{(i)} \cdot \rho\big)\boldsymbol{w}_n^{(i)} + \frac{\mu_{n+1}^{(i)} z_{n+1}\boldsymbol{x}_{n+1}}{1 + e^{z_{n+1}\boldsymbol{x}_{n+1}^\top \boldsymbol{w}_n^{(i)}}} \tag{21}$$

with $i = 1, 2$ for $F^{(1)}$ and $F^{(2)}$, respectively, since the derivative of $Q_{\text{logis}}(\boldsymbol{w}; \boldsymbol{\chi}_{n+1})$ with regard to $\boldsymbol{w}$ was provided by:

$$\nabla_{\boldsymbol{w}} Q_{\text{logis}}(\boldsymbol{w}; \boldsymbol{\chi}_{n+1}) = \rho\boldsymbol{w} - \frac{z_{n+1}\boldsymbol{x}_{n+1}}{1 + e^{z_{n+1}\boldsymbol{x}_{n+1}^\top \boldsymbol{w}}}. \tag{22}$$

At all time instants, filters $F^{(1)}$ and $F^{(2)}$ had step-sizes of $\mu_{n+1}^{(1)} = 0.08$ and $\mu_{n+1}^{(2)} = 0.01$, respectively, which was the sole difference between them. Additionally, the cost function of combination layer was also set to (19), namely:

$$J_{\text{logis}}^{\text{c}}\big(\gamma\boldsymbol{w}_n^{(1)} + (1-\gamma)\boldsymbol{w}_n^{(2)}\big) \triangleq \mathbb{E}\big\{Q_{\text{logis}}\big(\gamma\boldsymbol{w}_n^{(1)} + (1-\gamma)\boldsymbol{w}_n^{(2)}; \boldsymbol{\chi}_{n+1}\big)\big\}, \tag{23}$$

which was substituted into (7) to evaluate γ, resulting in:

$$\gamma_{n+1} \approx \gamma_n - \mu_\gamma \text{sgn}\Big\{\Big(\rho\boldsymbol{w}_n^{\text{c}} - \frac{z_{n+1}\boldsymbol{x}_{n+1}}{1 + e^{z_{n+1}\boldsymbol{x}_{n+1}^\top \boldsymbol{w}_n^{\text{c}}}}\Big)^\top \big(\boldsymbol{w}_n^{(1)} - \boldsymbol{w}_n^{(2)}\big)\Big\} \tag{24}$$

Fig. 3 displays the outcomes of the simulation. Fig. 3(a) shows that two candidate filters with different steady-state MSDs and convergence rates are the result of using different step-sizes. The affine sign-gradient scheme reduces the steady-state MSD and speeds up the convergence rate by integrating the best features of two candidate filters. The affine sign-gradient scheme achieves a better convergence rate with the aid of the weight-copying trick. Furthermore, for the sake of demonstration, the change in parameter γ_n is shown in Fig. 3(b). All of the experimental findings show that the suggested affine sign-gradient scheme and the weight-copying trick work well with both nonlinear models and generic cost functions.

V. CONCLUSIONS

The affine combination scheme was successfully extended to generic cost functions in this paper, enabling a nonlinear model. All three generic cost functions might be specified and optimized separately inside this combination framework. We proposed an affine combination scheme by using the stochastic sign-gradient descent, and a weight-copying trick had been introduced to enhance its performance. Simulation results were also provided to validate their effectiveness.

ACKNOWLEDGEMENT

The work of Danqi Jin was supported in part by the National Natural Science Foundation of China (NSFC) under Grant No. 62401405, and in part by the 'Postdoctoral Excellence Programme' of Wuhan University. The work of Jie Chen was supported in part by the NSFC under Grants Nos. 62171380 and 62192713. The work of Gongping Huang was supported in part by the NSFC under Grant No. 62471340.

REFERENCES

[1] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1985.

[2] S. Haykin, *Adaptive Filter Theory*, Pearson Education India, 4th edition, 2005.

[3] A. H. Sayed, *Adaptive Filters*, John Wiley & Sons, Inc., 2008.

[4] D. Jin, J. Chen, C. Richard, and J. Chen, "Online proximal learning over jointly sparse multitask networks with $\ell_{\infty,1}$ regularization," *IEEE Trans. Signal Process.*, vol. 68, pp. 6319–6335, 2020.

[5] Y. Chen, D. Jin, J. Chen, C. Richard, W. Zhang, G. Huang, and J. Chen, "Online parameter estimation over distributed multitask networks with a rank-one model," in *32nd European Signal Processing Conference*, 2024, pp. 1042–1046.

[6] J. Benesty, T. Gnsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*, Springer Publishing Company, Incorporated, 1st edition, 2010.

[7] C. Paleologu, J. Benesty, and S. Ciochina, *Sparse adaptive filters for echo cancellation*, Morgan & Claypool, San Rafael, Calif, 2010.

[8] S. Haykin, *Adaptive Radar Signal Processing*, John Wiley & Sons, 2007.

[9] S. Bose, A. De, and I. Chakrabarti, "Framework for automated earthquake event detection based on denoising by adaptive filter," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 67, no. 9, pp. 3070–3083, 2020.

[10] M. Martinez-Ramon, J. Arenas-Garcia, A. Navia-Vazquez, and A. R. Figueiras-Vidal, "An adaptive combination of adaptive filters for plant identification," in *Proc. Int. Conf. Digital Signal Process.*, 2002, vol. 2, pp. 1195–1198.

[11] L. A. Azpicueta-Ruiz, A. R. Figueiras-Vidal, and J. Arenas-Garcia, "A normalized adaptation scheme for the convex combination of two adaptive filters," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Las Vegas, NV, USA, Mar. 2008, pp. 3301–3304.

[12] S. Zhang, H. Zhao, and H. So, "Delayed combination of adaptive filters in colored noise," *IEEE Trans. Signal Process.*, vol. 70, pp. 1918–1931, 2022.

[13] D. Jin, J. Chen, C. Richard, J. Chen, and A. H. Sayed, "Affine combination of diffusion strategies over networks," *IEEE Trans. Signal Process.*, vol. 68, no. 1, pp. 2087–2104, Dec. 2020.

[14] D. Jin, J. Chen, C. Richard, J. Chen, and A. H. Sayed, "Convex combination of diffusion strategies over networks," *IEEE Trans. Signal Inf. Process. Networks.*, vol. 6, pp. 714–731, 2020.

[15] D. Jin, J. Chen, and J. Chen, "Convex combination of diffusion strategies over distributed networks," in *Proc. Asia-Pacific Signal Inf. Process. Association*, Hawaii, USA, Nov. 2018, pp. 224–228.

[16] J. Arenas-Garcia, L. A. Azpicueta-Ruiz, M. T. M. Silva, V. H. Nascimento, and A. H. Sayed, "Combinations of adaptive filters: Performance and convergence properties," *IEEE Signal Process. Mag.*, vol. 33, no. 1, pp. 120–140, Jan. 2016.

[17] S. S. Kozat, A. T. Erdogan, A. C. Singer, and A. H. Sayed, "Steady-state MSE performance analysis of mixture approaches to adaptive filtering," *IEEE Trans. Signal Process.*, vol. 58, no. 8, pp. 4050–4063, Aug. 2010.

[18] L. Azpicueta-Ruiz, M. Zeller, A. Figueiras-Vidal, J. Arenas-Garcia, and W. Kellermann, "Adaptive combination of volterra kernels and its application to nonlinear acoustic echo cancellation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 19, no. 1, pp. 97–110, 2011.

[19] W. Lu, L. Zhang, J. Chen, and J. Chen, "Generalized combined nonlinear adaptive filters: From the perspective of diffusion adaptation over networks," *Signal Process.*, vol. 172, pp. 107507, 2020.

[20] C. Pan, J. Chen, J. Benesty, and G. Shi, "On the design of target beampatterns for differential microphone arrays," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 8, pp. 1295–1307, 2019.

[21] J. Jin, X. Luo, G. Huang, J. Chen, and J. Benesty, "Beamforming through online convex combination of differential beamformers," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2024, pp. 8561–8565.

[22] G. Huang, J. Benesty, I. Cohen, E. Winebrand, J. Chen, and W. Kellermann, "Switching kronecker product linear filtering for multispeaker adaptive speech dereverberation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2023, pp. 1–5.

[23] G. Huang, J. Benesty, and J. Chen, "Kronecker product multichannel linear filtering for adaptive weighted prediction error-based speech dereverberation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 30, pp. 1277–1289, 2022.

[24] A. H. Sayed, *Adaptation, Learning, and Optimization over Networks*, vol. 7, Now Publishers Inc., Hanover, MA, USA, Jul. 2014.

[25] N. J. Bershad, J. C. M. Bermudez, and J. Y. Tourneret, "An affine combination of two LMS adaptive filters—transient mean-square analysis," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1853–1864, May 2008.

[26] V. H. Nascimento and R. C. de Lamare, "A low-complexity strategy for speeding up the convergence of convex combinations of adaptive filters," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2012, pp. 3553–3556.

[27] Y. Chen, Y. Gu, and A. O. Hero, "Sparse LMS for system identification," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2009, pp. 3125–3128.

[28] J. Chen, C. Richard, Y. Song, and D. Brie, "Transient performance analysis of zero-attracting LMS," *IEEE Signal Process. Letters*, vol. 23, no. 12, pp. 1786–1790, Dec. 2016.

[29] D. Jin, J. Chen, C. Richard, and J. Chen, "Model-driven online parameter adjustment for zero-attracting LMS," *Signal Process.*, vol. 152, pp. 373–383, Nov. 2018.

[30] D. Jin, J. Chen, C. Richard, and J. Chen, "Adaptive parameters adjustment for group reweighted zero-attracting LMS," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, April 2018, pp. 4294–4298.