Intelligent and Converged Networks
2023, 4(1): 1−14

ISSN 2708-6240
DOI: 10.23919/ICN.2023.0001

Denoising enabled channel estimation for underwater acoustic communications: A sparsity-aware model-driven learning approach

Sicong Liu*, Younan Mou, Xianyao Wang, Danping Su, and Ling Cheng

Abstract: It has always been difficult to achieve accurate information of the channel for underwater acoustic communications because of the severe underwater propagation conditions, including frequency-selective property, high relative mobility, long propagation latency, and intensive ambient noise, etc. To this end, a deep unfolding neural network based approach is proposed, in which multiple layers of the network mimic the iterations of the classical iterative sparse approximation algorithm to extract the inherent sparse features of the channel by exploiting deep learning, and a scheme based on the Sparsity-Aware DNN (SA-DNN) for UAC estimation is proposed to improve the estimation accuracy. Moreover, we propose a Denoising Sparsity-Aware DNN (DeSA-DNN) based enhanced method that integrates a denoising CNN module in the sparsity-aware deep network, so that the degradation brought by intensive ambient noise could be eliminated and the estimation accuracy can be further improved. Simulation results demonstrate that the performance of the proposed schemes is superior to the state-of-the-art compressed sensing based and iterative sparse recovery schems in the aspects of channel recovery precision, pilot overhead, and robustness, particularly under unideal circumstances of intensive ambient noise or inadequate measurement pilots.

Key words: Orthogonal Frequency Division Multiplexing (OFDM); Underwater Acoustic Communications (UAC); sparse recovery; deep learning; sparse learning; denoising; approximate message passing

1 Introduction

Recent years have witnessed rapid prosperity of academic research and industrial deployment of underwater acoustic communication, from the initial application in marine communications to various fields, such as oceanographic science, marine environment monitoring, and marine resources acquisition, etc.[1] Nevertheless, because of the time-varying property of the sea environment, the movement and mobility of the transceivers, and the interference or ambient noise, there are many detrimental properties underlay the channel of underwater acoustic communication, like the fast-fading property, frequency selectivity property, and severe Doppler distortion, which seriously impact the performance and development of high-rate transmission in the underwater environment[2, 3]. Meanwhile, the famous Orthogonal Frequency Division Multiplexing (OFDM) technology has also been widely applied in the underwater acoustic transmission apart from the conventional single carrier mode, thanks to its many advantages including good spectrum efficiency and strong anti-frequency-selectivity capability[4].

The performance of channel estimation is crucially important in order to guarantee and promote the application and effectiveness of Underwater Acoustic

• Sicong Liu is with School of Informatics, Xiamen University, Xiamen 361005, China, and also with National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China. E-mail: liusc@xmu.edu.cn.
• Younan Mou, Xianyao Wang, and Danping Su are with School of Informatics, Xiamen University, Xiamen 361005, China. E-mail: {mouyn, 2332021115497, sudanping}@xmu.edu.cn.
• Ling Cheng is with the Information Engineering, University of the Witwatersrand, Johannesburg 2050, South Africa. E-mail: ling.cheng@wits.ac.za.
∗ To whom correspondence should be addressed.
Manuscript received: 2022-11-22; revised: 2022-12-25; accepted: 2023-01-07

transmission based on OFDM (UA-OFDM) acoustic modulation[5]. However, the complicated characteristics of the UAC and the severe transmission environment tremendously increase the difficulty of channel estimation[6]. Fortunately, usually there exist sparse characteristics within the Channel Impulse Response (CIR) of the UAC, i.e., most dominant power of the channel is clustered in only a small portion of the channel taps, resulting in significant reduction in the amount of the unknown coefficients to be estimated[5, 6]. Thus, it is very crucial to make use of the sparse features of underwater acoustic channels in order to enhance the reliability and accuracy of the channel[7].

Since the past two decades, there have been plenty of research on underwater acoustic channel estimation which lies in different categories of technological routines. The traditional methods, mainly including Least Square (LS) and Minimum Mean Square Error (MMSE)[8, 9], have been widely adopted in UAC estimation. However, the sparsity of the UAC is not considered, and thus the cost of pilots in the frequency domain is relatively high and the performance of channel estimation is constrained[8, 9]. Ever since the advent of Compressed Sensing (CS)[10], it has been widely applied in UAC estimation methods, since CS is able to exploit the channel sparsity and reconstruct the channel coefficients using compressed measurements. Among the CS-based methods, the greedy algorithms based on Orthogonal Matching Pursuit (OMP) have attracted extensive research attention[11−18]. However, even if the compressive sensing schemes are able to exploit the sparsity of the UAC inherently to resolve the difficulty of conventional schemes, the efficacy tends to be degraded or limited especially under some difficult circumstances, like strong background noise or lack of available pilots[2, 19].

In more recent years, the rapidly developing technology of deep learning has been adopted in various research areas, including sparse approximation[20, 21], massive MIMO communications[22, 23], and Internet-of-Things (IoT) systems[24, 25], etc., enlightening a novel possible approach to the estimation of underwater acoustic channels. Inspired by this, making use of the sparse recovery theory, deep learning, and deep unfolding networks, we combine the classical algorithm of iterative sparse approximation, i.e., approximate message passing, with DNN to propose two DL-based schemes for UAC estimation, which are able to achieve better system performance with more precise channel information and lower cost in pilots, and better robustness, compared to the state-of-the-art schemes. The main contributions of this paper are summarized as follows:

- A Sparsity-Aware DNN (SA-DNN) is devised for UAC estimation, which mimics the classical iterative algorithm for sparse approximation, i.e., approximate message passing, using a series of deep unfolding neural layers with learnable weights to extract the inherent sparse features from the underwater acoustic channel by exploiting deep learning in order to enhance the stability of the estimation of the channel.
- An improved Denoising Sparsity-Aware DNN (DeSA-DNN) is devised for UAC estimation, which substitutes the shrinkage function of the SA-DNN with the Denoising Convolutional Neural Network (DnCNN) to enhance the accuracy and reliability of the channel estimation scheme and combat against intensive background noise.
- A State Evolution (SE) framework is formulated to quantitatively predict the average Mean Square Error (MSE) of each layer of the DeSA-DNN. It is theoretically proved that the layer-by-layer training of the DeSA-DNN for UAC estimation is MMSE optimal.

The remainder of this paper is organized as follows. Section 2 lists the related works of UAC estimation, which are our inspiration for identifying the research gaps. Section 3 introduces the system model and problem formulation for UAC estimation. Sections 4 and 5 present the UAC estimation schemes based on SA-DNN and DeSA-DNN, respectively. Section 6 formulates the SE framework and analyzes the parameter tuning problem of the DeSA-DNN. The simulation results are shown in Section 7 with discussions, followed by the conclusion in Section 8.

2 Related work

So far, various approaches towards the inference of underwater acoustic communication channels have been extensively investigated, which mainly includes two-fold classifications: conventional signal processing based schemes and sparse recovery algorithm enabled schemes such as compressive sensing.

Typical conventional signal processing based schemes include LS and MMSE methods[8, 9], etc. In Ref. [8], an improved LS method aimed at estimating the underwater acoustic channel is proposed, utilizing principle of threshold crossing selection to improve the accuracy of the conventional LS method. A transform-domain MMSE based method is proposed in Ref. [9] to improve the performance of UAC estimation. It is usually not so difficult to perform conventional signal processing schemes. However, the inherent sparsity within the underwater acoustic channel has not been made fully use of. This might lead to relatively higher cost of frequency domain pilots in channel estimation and reduce the precision of the channel information inference, which is caused by incorrect paths contained in the estimation result[8, 9].

Motivated by the sparsity of the UAC, some sparse recovery algorithms that emerged in the past decade have been widely applied in sparse UAC estimation with considerable performance gains. There are usually two kinds of mainstream compressive sensing approaches, i.e., the convex relaxation schemes and greedy iterative algorithms. A typical subsidiary scheme under convex relaxation schemes is to relax the nonconvex problem into l_1 norm minimization problem, which is a convex one, including typically Basis Pursuit (BP) and AMP[26, 27], etc. In Ref. [28], the basis pursuit denoising algorithm is adopted to estimate the UAC, which achieves an outstanding performance. A sparsity-aware approach proposed in Ref. [29] uses the enhanced algorithm of GAMP to simultaneously infer channel state information as well as impulsive noise. However, some statistical information with respect to the channel and noise should be obtained in advance. When the convex relaxation approaches are utilized, usually the estimation result will not be exactly sparse, i.e., the estimation result contains some small values that do not contribute significantly to the estimation error.

On the other hand, the methods based on greedy algorithms are the most popular in sparse UAC estimation, including the typical CS-based greedy algorithm of OMP and many related improved greedy algorithms[11−18]. In Ref. [11], an OMP-based UAC estimation scheme with equally spaced pilots provides a closed-form estimate for the path delay and improves the estimation accuracy compared to the traditional methods. In Ref. [12], a low-complexity OMP based UAC estimation method calculates the candidate path delays in advance to avoid the repeated calculations in each iteration. Based on OMP, a two-stage UAC estimation approach is proposed in Ref. [13], which can estimate the path delay and Doppler scale with high accuracy.

Apart from these, Ref. [15] adopted the greedy iterative compressive sensing algorithm of CoSaMP aimed at obtaining the channel state information of the underwater acoustic communication, but the number of paths, i.e., sparsity level, should be known in advance. Unlike this algorithm, Do et al. in Ref. [16] proposed the algorithm of Sparse Adaptive Matching Pursuit (SAMP), where the sparsity level can be adaptively changed in the iteration process. In this way, the channel state information of the underwater acoustic communication can be obtained without knowing the sparsity level in advance. Based on SAMP, Zhang et al.[17] proposed an adaptive enhanced algorithm based on SAMP, which adopts an adjustable parameter to indicate the sparsity to improve the estimation accuracy, and the number of iterations might increase. Inspired by the AS-SAMP algorithm, Ref. [18] proposed an improved CoSaMP algorithm that supports different sparsity levels by employing adaptive thresholding, which achieves an effective compromise among the precision of channel state information and the cost of computing resource.

It should be pointed out that, despite the fact that the classical compressive sensing approach is able to make use of the sparse features within the underwater acoustic channel to break the bottleneck of conventional signal processing based schemes, its inference precision tends to be degraded or limited in severe conditions, such as strong background interference and lack of available frequency training subcarriers[2, 19].

In the past five years, the rapidly developing technology of DL has been applied in many research areas, e.g., sparse recovery[20, 21], massive MIMO communications[22, 23], and IoT systems[24, 25], etc.[30, 31] In Ref. [20], by unpacking each iteration in the iterative algorithm, a Learned Approximate Message Passing (LAMP) structure is designed for sparse linear inverse problems. Inspired by the LAMP network, Ref. [23] applies it to the massive MIMO systems, which greatly improves the accuracy of channel estimation. Meanwhile, currently there have not been plenty of studies on UAC estimation utilizing DL and DNN[32]. In Ref. [32], a deep learning enabled scheme for underwater acoustic channel estimation method is investigated, which brings adaptability to changing environments through the flexible parameter tuning of the DNN. Considering the scarcity of studies in this direction, the endeavor to utilize deep learning and deep unfolding networks to learn the sparse feature and obtain the information of the underwater acoustic channel is still an unexplored problem.

3 System model and problem formulation

In order to deal with the challenge of frequency selectivity of the underwater channels, a typical broadband communication system uses orthogonal frequency division multiplexing investigated in our model[2, 3]. Assume that the OFDM temporal length and the cyclic prefix temporal length are represented by T and $T_{\rm CP}$, respectively. Thus, an OFDM frame is of temporal length $T_{\rm bl}$, which is given by $T_{\rm CP}+T$. Assuming that the OFDM block size, i.e., the subcarrier amount, is $N_{\rm c}$, the central frequency of the k-th subcarrier of the underwater acoustic communication system is given by

$$f_k = f_{\rm c} + \frac{k}{T},\quad k = -\frac{N_{\rm c}}{2}, -\frac{N_{\rm c}}{2} = 1, 2, \ldots, \frac{N_{\rm c}}{2} - 1 \tag{1}$$

Assume that $s[k]$ represents the modulated symbol at the k-th subcarrier, and then the time-domain OFDM block is given by

$$x(t) = \mathrm{Re}\left\{\sum_{k\in S_{\rm A}} s[k]\mathrm{e}^{\mathrm{j}2\pi f_k t}\right\},\quad t\in[-T_{\rm CP}, T] \tag{2}$$

where $\mathrm{Re}\{\cdot\}$ represents taking real value and $S_{\rm A}$ represents the available subcarrier set.

Usually, the channel impulse response of the fast fading underwater channel with L propagation paths can be represented as[3, 7, 33]

$$h(t;\tau) = \sum_{l=1}^{L} A_l(t)\delta(\tau - \tau_l(t)) \tag{3}$$

where $A_l(t)$ and $\tau_l(t)$ are the amplitude and delay of the l-th path, respectively. Note that the UAC has a sparse structure in the sense that there are only a few, i.e., K channel taps with power much greater than the other taps, while the power of the rest taps is sufficiently small or approximately zero. Thus, the CIR can be regarded as a K-sparse vector with K being much smaller than L[5−7], as shown in Fig. 1.

According to related research in Refs. [3, 7, 33, 34], it is assumed that these conditions related to the underwater channel can be satisfied over the time in one OFDM frame: (1) The power of any path remains approximately invariant, namely, $A_l(t) \approx A_l$. (2) The tap latency is modeled by Doppler scaling coefficient by $\tau_l(t) \approx \tau_l - \alpha t$, with τ_l being original latency, while α is Doppler scaling coefficient that is identical for all paths.

Based on the assumptions related to the channel

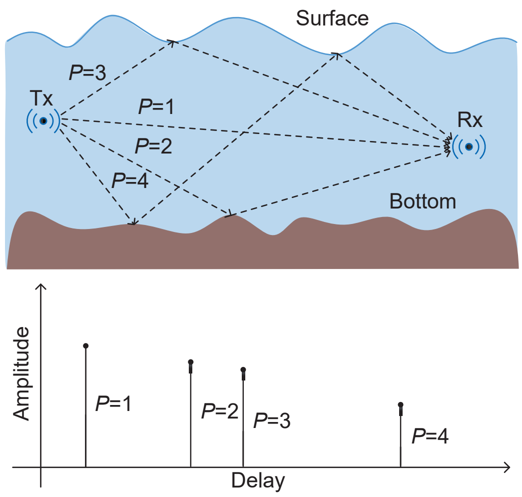

Fig. 1 Schematic diagram of the multipath propagation of the underwater acoustic channel in a shallow water environment.

described above, the UAC model in Eq. (3) can be simplified as[7, 33, 34]

$$h(t;\tau)=\sum_{l=1}^{L}A_l\delta(\tau-(\tau_l-\alpha t)) \tag{4}$$

After passing through the underwater acoustic channel, the OFDM signal at the receiver is given by[7, 33]

$$r(t)=\sum_{l=1}^{L}A_l x((1+\alpha)t-\tau_l)+n(t)= \\ \mathrm{Re}\left\{\sum_{l=1}^{L}A_l\left\{\sum_{k\in S_{\mathrm{A}}}s[k]\mathrm{e}^{\mathrm{j}2\pi f_k((1+\alpha)t-\tau_l)}\right\}\right\}+n(t) \tag{5}$$

where $n(t)$ denotes the Additive White Gaussian Noise (AWGN).

Similar to the procedures in Ref. [33], we adopt the method of Doppler compensation on $r(t)$ to mitigate the Doppler spread, then the received signal after resampling and Carrier Frequency Offset (CFO) compensation is given by

$$y(t)=r\left(\frac{t}{1+\hat{\alpha}}\right)\mathrm{e}^{-\mathrm{j}2\pi\hat{\varepsilon}t}\approx \\ \mathrm{Re}\left\{\sum_{l=1}^{L}A_l\left\{\sum_{k\in S_{\mathrm{A}}}s[k]\mathrm{e}^{\mathrm{j}2\pi f_k(t-\tau_l)}\right\}\right\}+n(t) \tag{6}$$

where $\hat{\alpha}$ denotes the resampling coefficient; $\hat{\varepsilon}$ is the inferred residual Doppler frequency used for CFO compensation.

We adopt the channel inference scheme based on the frequency pilots[7], where a couple of N_{p} subcarriers are utilized to convey known pilot information. Equation (6) can be rewritten in vector format, and thus the issue of underwater acoustic channel estimation is modeled by

$$\boldsymbol{y}=\boldsymbol{X}\tilde{\boldsymbol{h}}+\boldsymbol{n}=\boldsymbol{X}\boldsymbol{F}_{\mathrm{p}}\boldsymbol{h}+\boldsymbol{n}=\boldsymbol{A}\boldsymbol{h}+\boldsymbol{n} \tag{7}$$

where $\boldsymbol{y}=\left[y_1\ y_2\ \cdots\ y_{N_{\mathrm{p}}}\right]^{\mathrm{T}}$ denotes the received pilots. $\boldsymbol{X}=\mathrm{diag}\left[x_1\ x_2\ \cdots\ x_{N_{\mathrm{p}}}\right]$ denotes a diagonal matrix with its diagonal entries being the frequency pilots. $\tilde{\boldsymbol{h}}=\boldsymbol{F}_{\mathrm{p}}\boldsymbol{h}$ is the channel frequency response. $\boldsymbol{h}=[h_1\ h_2\ \cdots\ h_L]^{\mathrm{T}}$ is the K-sparse channel impulse response. $\boldsymbol{F}_{\mathrm{p}}$ is the $N_{\mathrm{p}}\times L$ normalized partial discrete Fourier transform matrix composed of the N_{p} rows corresponding to the pilot position and the first L columns of the original $N_{\mathrm{c}}\times N_{\mathrm{c}}$ discrete Fourier transform matrix. Matrix $\boldsymbol{A}$ is given by $\boldsymbol{X}\boldsymbol{F}_{\mathrm{p}}$. $\boldsymbol{n}=\left[n_1\ n_2\ \cdots\ n_{N_{\mathrm{p}}}\right]^{\mathrm{T}}$ denotes the AWGN vector with zero mean and standard deviation $\sigma_{\boldsymbol{n}}$, i.e., $\boldsymbol{n}\sim\mathcal{N}\left(\boldsymbol{0},\sigma_{\boldsymbol{n}}^2\boldsymbol{I}_{N_{\mathrm{p}}}\right)$.

According to the CS theory, let $\boldsymbol{A}$ denote the observation matrix and $\boldsymbol{y}$ denote the measurement vector, and then the channel inference model in Eq. (7) can be regarded as a problem of sparse reconstruction that is able to be resolved using the sparse recovery algorithms, such as convex relaxation approaches and CS-based greedy algorithms. In this paper, we will introduce deep CS-based methods, i.e., the sparsity-aware model-driven deep unfolding neural networks mimicking the iterations of sparse recovery algorithms, in order to overcome the limitation of existing CS-based algorithms and achieve better performance of UAC estimation in severe conditions.

4 SA-DNN for underwater acoustic channel estimation

The UAC estimation scheme based on the proposed SA-DNN algorithm is summarized in Algorithm 1, including two phases: training and inference phases. As shown in Fig. 2, the proposed SA-DNN scheme mimics the iterative behaviors of the classical sparse approximation algorithm, i.e., approximate message passing, and converts the iteration to deep unfolding neural layers with distinct learnable parameters, in order to extract the sparsity nature within the underwater channel by exploiting DL[20, 27].

In Algorithm 1, $\boldsymbol{v}_t$ is the residual error of layer-t. σ_t is the estimated standard deviation of $\boldsymbol{v}_t$. $b_t\boldsymbol{v}_{t-1}$ is the Onsager correction term[35], which can increase the rate to converge. $\hat{\boldsymbol{h}}_t$ denotes the inference outcome of layer-t. $\boldsymbol{B}_t$ is a learnable matrix for layer-t. $\boldsymbol{\eta}(\cdot)$ is an operation of soft thresholding identically adopted at all layers as given by

$$\boldsymbol{\eta}(\cdot;\lambda_t,\sigma_t)=\mathrm{sgn}(\cdot)\max(\cdot-\lambda_t\sigma_t,0) \tag{8}$$

where λ_t is the threshold shrinkage factor that can be learnt for layer t. $S\left\{\hat{\boldsymbol{h}}_T,K\right\}$ is a selection operation returning the K biggest elements of $\hat{\boldsymbol{h}}_T$. $\varOmega$ is the corresponding locations of nonzero elements in the sparse channel vector. The $M\times N$ sub-matrix $\boldsymbol{A}_{\varOmega}$ is composed of columns of $\boldsymbol{A}$ corresponding to the support $\varOmega$.

Algorithm 1 SA-DNN based scheme for UAC estimation

Training phase.

Feed-in data: Observation matrix $\boldsymbol{A}$; Dataset $\left\{(\boldsymbol{y}^d,\boldsymbol{h}^d)\right\}_{d=1}^{D}$ for learning (including D data samples, with each consisting of a measurement vector $\boldsymbol{y}$ and related real channel vector $\boldsymbol{h}$).

Before start: $\boldsymbol{v}_0=\boldsymbol{0}$, $\hat{\boldsymbol{h}}_0=\boldsymbol{0}$, $N=L$, $M=N_{\mathrm{p}}$.

/* Learning parameters of the network layer-wise. */

for $t=1,2,3,\ldots$ **do**

1: Set original values for training weights for SA-DNN via $\boldsymbol{B}_t\leftarrow\boldsymbol{A}^{\mathrm{T}}$ and $\lambda_t\leftarrow 1.0$.

2: Calculate $\boldsymbol{v}_t$ through $\sigma_t\leftarrow\dfrac{1}{\sqrt{M}}\|\boldsymbol{v}_{t-1}\|_2$.

3: Calculate the input of the soft threshold shrinkage function $\boldsymbol{\eta}(\cdot)$ via $\boldsymbol{r}_t\leftarrow\hat{\boldsymbol{h}}_{t-1}+\boldsymbol{B}_t\boldsymbol{v}_{t-1}$.

4: Obtain inferred channel vector $\hat{\boldsymbol{h}}_t\leftarrow\boldsymbol{\eta}(\boldsymbol{r}_t;\lambda_t,\sigma_t)$.

5: Derive $b_t\leftarrow\dfrac{1}{M}\|\hat{\boldsymbol{h}}_t\|_0$.

6: Obtain residual measurement vector $\boldsymbol{v}_t\leftarrow\boldsymbol{y}-\boldsymbol{A}\hat{\boldsymbol{h}}_t+b_t\boldsymbol{v}_{t-1}$.

7: Perform SGD optimization to update $\boldsymbol{B}_t$ and λ_t, which optimizes loss function $L_t(\boldsymbol{\Theta})$ in Eq. (9).

8: **If** $L_t(\boldsymbol{\Theta})\geqslant L_{t-1}(\boldsymbol{\Theta})$, **then** set total network layer amount to $T\leftarrow t-1$, Terminate.

end for

Feed-out data: Trained weights $\Theta=\left\{\{\boldsymbol{B}_t\}_{t=1}^{T},\{\lambda_t\}_{t=1}^{T}\right\}$.

Inference phase.

Feed-in data: Actual measurement data $\boldsymbol{y}$; observation matrix $\boldsymbol{A}$; and learnt weights Θ.

Before start: $\boldsymbol{h}_{\mathrm{SA}}=\boldsymbol{0}$.

1: Feed $\boldsymbol{y}$ into the learnt SA-DNN described by weights $\boldsymbol{\Theta}$, and feed out coarsely inferred channel vector $\hat{\boldsymbol{h}}_T=\boldsymbol{\eta}(\boldsymbol{r}_T;\lambda_T,\sigma_T)$, which takes only a single forward calculation through the network.

2: Determine the support, i.e., the positions of the K largest nonzero elements, from the network output $\hat{\boldsymbol{h}}_T$, i.e., $\Omega=S\left(\hat{\boldsymbol{h}}_T,K\right)$.

3: Refine the amplitude of the channel taps on support Ω using least squares, leading to precisely inferred channel vector $\boldsymbol{h}_{\mathrm{SA}}=\boldsymbol{A}_{\Omega}^{\dagger}\boldsymbol{y}=\left(\boldsymbol{A}_{\Omega}^{\mathrm{H}}\boldsymbol{A}_{\Omega}\right)^{-1}\boldsymbol{A}_{\Omega}^{\mathrm{H}}\boldsymbol{y}$.

Feed-out data: Inferred underwater acoustic channel information $\boldsymbol{h}_{\mathrm{SA}}$.

For the training dataset, $\left\{(\boldsymbol{y}^d,\boldsymbol{h}^d)\right\}_{d=1}^{D}$ is the size-D training dataset, where $\boldsymbol{y}^d$ and $\boldsymbol{h}^d$ are the measurement data and its related label, i.e., the real channel impulse response, for sample-d. Each data sample includes (feature, label) couples that are used to train the learnable parameters $\boldsymbol{\Theta}=\left\{\{\boldsymbol{B}_t\}_{t=0}^{T},\{\lambda_t\}_{t=0}^{T}\right\}$ of the SA-DNN by minimizing the following MSE loss function:

$$L_t(\Theta)=\frac{1}{D}\sum_{d=1}^{D}\left\|\boldsymbol{h}^d-\hat{\boldsymbol{h}}_t\left(\boldsymbol{y}^d,\Theta\right)\right\|_2^2 \tag{9}$$

During the inference phase of Algorithm 1, the

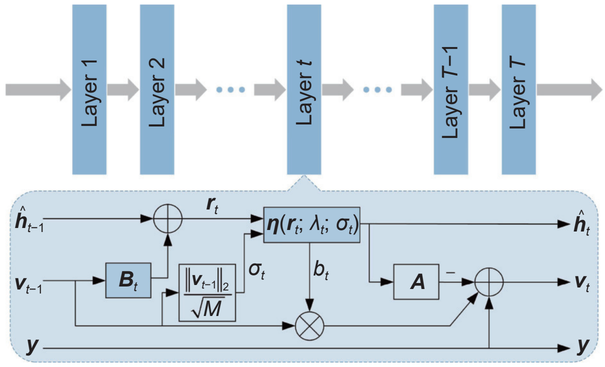

Fig. 2 Proposed SA-DNN scheme consisting of several combined layers, each having same model but different learnable parameters.

learnable parameters $\boldsymbol{\Theta}$ are trained layer by layer: Originally, a single-layer SA-DNN network is formulated whose loss function, i.e., $L_1(\boldsymbol{\Theta})$, is optimized. After that, another new layer is appended to the original network to formulate a new network, whose loss function, $L_2(\boldsymbol{\Theta})$, is also optimized. This process is replicated in the same manner till there are in total T layers in the network, whose loss function, i.e., $L_T(\boldsymbol{\Theta})$, can be optimized. It is worthwhile to point out that during the training of a specific layer of the network, its preceding layers are assumed to be invariant in order to facilitate the layer-wise training mechanism. At the moment, the loss function $L_t(\boldsymbol{\Theta})$ is able to be minimized and the training weights $\boldsymbol{\Theta}$ are learnt via SGD method and back propagation. The algorithm should terminate if the following condition is satisfied: $L_t(\boldsymbol{\Theta})\geqslant L_{t-1}(\boldsymbol{\Theta})$. This is because the increase of the loss function with the layer amount implies possible overfitting. Hence, we can set the final amount for the network layers as $T=t-1$. At this moment of termination of training, the network weights $\boldsymbol{\Theta}$ are also finalized.

In the estimation stage of Algorithm 1, the unknown sparse underwater channel vector can be reconstructed using the SA-DNN with the learnt parameters. It is noted that the inference phase only requires one forward propagation through the network, which does not consume too much complexity. The inferred result out of the network can be regarded as a coarse estimation of the channel, whose support can be recognized as precise while the amplitude might be with greater errors. Thus, the unknown support Ω of the underwater channel $\hat{\boldsymbol{h}}_T$ can be inferred from the network. After that, in order to refine the amplitude of the channel, we can employ the traditional least squares method to obtain a more precise estimation of the amplitude of the channel taps, which yields the final refined channel vector denoted by $\boldsymbol{h}_{\mathrm{SA}}$.

Since the proposed scheme utilizes the SA-DNN, DL, and a big dataset of the underwater channel to extract its sparsity nature, the inference precision and the reliability will be substantially promoted, which leads to the superiority of DL and data-driven approaches compared to classical CS-based algorithms.

5 DeSA-DNN for underwater acoustic channel estimation

In some severe underwater conditions such as offshore sea waters, the existence of intensive ambient noise, such as the noise following Gaussian distribution, might bring about great challenge to the sparsity-aware channel measurements and even the proposed sparse learning based recovery scheme of SA-DNN. To this end, DeSA-DNN scheme is further proposed, which substitutes the soft threshold shrinkage function of the SA-DNN with a denoiser, i.e., DnCNN. In this way, the channel inference precision can be improved especially in complex environments with intensive additive noise[21, 36]. The UAC estimation scheme based on the proposed DeSA-DNN algorithm is summarized in Algorithm 2.

As shown in Fig. 3, the DnCNN is treated as a black box, which is regarded as a denoiser module of the devised DeSA-DNN architecture. Specifically, the DnCNN module is a denoiser composed of 16 convolutional layers, which can deal with the Gaussian denoising problem. The first convolutional layer uses 64 different filters with size of $3\times3\times1$, followed by a Rectified Linear Unit (ReLU). Each of the subsequent 14 convolutional layers uses 64 different filters with size of $3\times3\times64$, and then Batch-Normalization (BN) and ReLU activation are performed, separately. The final convolutional layer uses a separate filter with size of $3\times3\times64$ for signal reconstruction[36].

In Algorithm 2, $\boldsymbol{v}_t$ is the residual measurement data for layer-t. $\hat{\boldsymbol{h}}_t$ is the inference channel information for layer-t. z_t denotes the effective noise for layer-t as given by $z_t = \hat{\boldsymbol{h}}_{t-1} + \boldsymbol{A}^{\mathrm{H}}\boldsymbol{v}_{t-1} - \boldsymbol{h}$. This means that it can be obtained through subtracting the real channel $\boldsymbol{h}$ from $\hat{\boldsymbol{h}}_{t-1} + \boldsymbol{A}^{\mathrm{H}}\boldsymbol{v}_{t-1}$. σ_t is the estimated standard deviation corresponding to equivalent interference z_t that is related with $\boldsymbol{v}_{t-1}$. An Onsager correction item, i.e., $b_t\boldsymbol{v}_{t-1}$, is introduced[35] in order to accelerate

Algorithm 2 DeSA-DNN based scheme for UAC estimation

Training phase.

Feed-in data: Observation matrix $\boldsymbol{A}$; Dataset $\left\{(\boldsymbol{y}^d, \boldsymbol{h}^d)\right\}_{d=1}^{D}$ for learning (including D data samples, with each consisting of a measurement vector $\boldsymbol{y}$ and related real channel vector $\boldsymbol{h}$).

Before start: $\boldsymbol{v}_0 = \boldsymbol{0}$, $\hat{\boldsymbol{h}}_0 = \boldsymbol{0}$, $N = L$, $M = N_{\mathrm{p}}$.

/* Learning parameters of the denoising enabled network layer-wise. */

for $t = 1,2,3,\ldots$ **do**

1: Set original values of the training weights $\boldsymbol{w}_t$ for DeSA-DNN drawn from standard Gaussian distribution $\mathcal{N}\left(\boldsymbol{0}, \boldsymbol{I}_{N_{\mathrm{D}}}\right)$.

2: Derive inferred standard deviation for equivalent interference by $\sigma_t \leftarrow \frac{1}{\sqrt{M}}\|\boldsymbol{v}_{t-1}\|_2$.

3: Calculate feed-in data for denoiser $\boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}(\cdot)$, i.e., $\boldsymbol{r}_t \leftarrow \hat{\boldsymbol{h}}_{t-1} + \boldsymbol{A}^{\mathrm{H}}\boldsymbol{v}_{t-1}$.

4: Obtain inferred channel state information vector $\hat{\boldsymbol{h}}_t \leftarrow \boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}\left(\hat{\boldsymbol{h}}_{t-1} + \boldsymbol{A}^{\mathrm{H}}\boldsymbol{v}_{t-1}\right)$.

5: Derive b_t using Monte-Carlo method, $b_t \leftarrow \frac{1}{M}\mathrm{div}\left(\boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}\right)\left(\hat{\boldsymbol{h}}_{t-1} + \boldsymbol{A}^{\mathrm{H}}\boldsymbol{v}_{t-1}\right)$.

6: Calculate residual measurement vector $\boldsymbol{v}_t \leftarrow \boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{h}}_t + b_t\boldsymbol{v}_{t-1}$.

7: Perform SGD optimization to update $\boldsymbol{w}_t$, which optimizes loss function $L_t(\boldsymbol{\Theta})$ in Eq. (9).

8: **If** $L_t(\boldsymbol{\Theta}) \geqslant L_{t-1}(\boldsymbol{\Theta})$, **then** set total network layer amount to $T \leftarrow t-1$, terminate.

end for

Feed-out data: Trained weights $\boldsymbol{\Theta} = \left\{\{\boldsymbol{w}_t\}_{t=1}^{T}\right\}$.

Inference phase.

Feed-in data: Actual measurement data $\boldsymbol{y}$; observation matrix $\boldsymbol{A}$; and learnt weights $\boldsymbol{\Theta}$.

Before start: $\boldsymbol{h}_{\mathrm{DeSA}} = \boldsymbol{0}$.

1: Feed $\boldsymbol{y}$ into the learnt DeSA-DNN described by weights $\boldsymbol{\Theta}$, and feed-out coarsely inferred channel vector $\hat{\boldsymbol{h}}_T = \boldsymbol{D}^T_{\boldsymbol{w}_T(\sigma_T)}\left(\hat{\boldsymbol{h}}_{T-1} + \boldsymbol{A}^{\mathrm{H}}\boldsymbol{v}_{T-1}\right)$, which takes only a single forward calculation through the network.

2: Determine the support, i.e., the positions of the K largest nonzero elements, from the network output $\hat{\boldsymbol{h}}_T$, i.e., $\Omega = S\left(\hat{\boldsymbol{h}}_T, K\right)$.

3: Refine the amplitude of the channel taps on support Ω using least squares, leading to precisely inferred channel vector $\boldsymbol{h}_{\mathrm{DeSA}} = \boldsymbol{A}_{\Omega}^{\dagger}\boldsymbol{y} = \left(\boldsymbol{A}_{\Omega}^{\mathrm{H}}\boldsymbol{A}_{\Omega}\right)^{-1}\boldsymbol{A}_{\Omega}^{\mathrm{H}}\boldsymbol{y}$.

Feed-out data: Inferred underwater acoustic channel information $\boldsymbol{h}_{\mathrm{DeSA}}$.

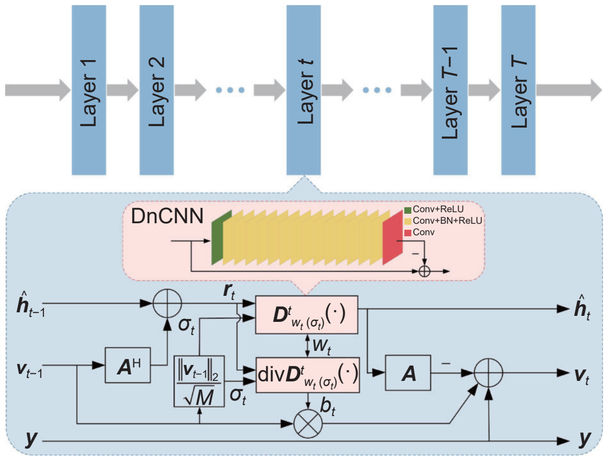

Fig. 3 Proposed DeSA-DNN architecture including several sequentially deployed deep unfolding neural layers of the same formulation; every network layer includes a pair of the same denoiser modules called DnCNN, whose parameters are the same.

convergence of the learning process and to enforce the statistical characteristics of the equivalent interference to approximately follow a normal distribution[21, 37]. Therefore, the equivalent interference is denoted as $\sigma_t \boldsymbol{\epsilon} \sim \mathcal{N}\left(\mathbf{0}, \sigma_t^2 \boldsymbol{I}_N\right)$, with the term $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_N)$[37].

$\boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}$ represents denoiser embedded into layer t, with its learnable parameter $\boldsymbol{w}_t$ dependent on σ_t. As explained above, the denoiser feed-in data for any layer is actually the superposition of the real channel vector and the equivalent interference. Hence, the denoiser takes $\hat{\boldsymbol{h}}_{t-1} + \boldsymbol{A}^{\mathrm{H}} \boldsymbol{v}_{t-1}$ into the module as feed-in data, returning inference channel information $\hat{\boldsymbol{h}}_t$ to implement the denoising of the equivalent interference[37], which is summarized in the fourth line of Algorithm 2.

We can also represent the denoiser divergence as $\mathrm{div} \boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}$. It is usually difficult to obtain a closed-form expression of the divergence of the denoiser, so we use the following Monte-Carlo approximation to calculate the divergence[38]. Given a denoiser $\boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}$, using the Gaussian distribution $\boldsymbol{u} \sim N(\mathbf{0}, \boldsymbol{I}_N)$ and an extremely small positive constant $c > 0$, the divergence can be estimated as

$$\mathrm{div} \boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)} \approx E\left\{ \frac{\boldsymbol{u}^{\mathrm{T}}}{c} \left(\boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}(\cdot + c\boldsymbol{u}) - \boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}(\cdot) \right) \right\} \tag{10}$$

The training and inference phases of the proposed SA-DNN scheme are similar to those of the proposed DeSA-DNN scheme, so the details are omitted for simplicity of presentation. It is noted that, during the learning process, the weights to be trained in the denoising enabled network, i.e., $\boldsymbol{\Theta} = \left\{ \{\boldsymbol{B}_t\}_{t=1}^T, \{\lambda_t\}_{t=1}^T \right\}$, are different from those of the non-denoising based network, i.e., $\boldsymbol{\Theta} = \left\{ \{\boldsymbol{w}_t\}_{t=1}^T \right\}$.

Since the proposed DeSA-DNN scheme has incorporated the denoiser into the neural network architecture to reduce the estimation error caused by the Gaussian noise, the accuracy of UAC estimation can be further improved compared to the SA-DNN based scheme.

6 Theoretical analysis

6.1 State evolution

The SE framework, which is a set of equations, is formulated to quantitatively predict the MSE of each layer of the DeSA-DNN[37]. Starting from $\gamma^0 = \frac{\|\hat{\boldsymbol{h}}_0 - \boldsymbol{h}\|_2^2}{N}$, the SE generates a sequence of values of MSE through the following iterations:

$$\begin{aligned} \gamma^t\left(\boldsymbol{h}, \delta, \sigma_{\boldsymbol{n}}^2\right) &= \frac{1}{N} E \left\| \hat{\boldsymbol{h}}_t - \boldsymbol{h} \right\|_2^2 = \\ &\frac{1}{N} E \left\| \boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}\left(\hat{\boldsymbol{h}}_{t-1} + \boldsymbol{A}^{\mathrm{H}} \boldsymbol{v}_{t-1} \right) - \boldsymbol{h} \right\|_2^2 = \\ &\frac{1}{N} E \left\| \boldsymbol{D}^t_{\boldsymbol{w}_t(\sigma_t)}(\boldsymbol{h} + \sigma_t \boldsymbol{\epsilon}) - \boldsymbol{h} \right\|_2^2 \end{aligned} \tag{11}$$

where $\sigma_t^2 = \dfrac{\gamma^t\left(\boldsymbol{h}, \delta, \sigma_{\boldsymbol{n}}^2\right)}{\delta} + \sigma_{\boldsymbol{n}}^2$; $\boldsymbol{h}$ is the ground-truth CIR; $\delta = M/N$ is the measurement ratio; $\sigma_{\boldsymbol{n}}$ is the standard deviation of the measurement noise. The symbol $\gamma^t\left(\boldsymbol{h}, \delta, \sigma_{\boldsymbol{n}}^2\right)$ denotes the average MSE of the t-th layer, which implies that γ^t may depend on the ground-truth CIR, the measurement ratio, and the measurement noise. By iteratively updating the SE of Eq. (11), the theoretical value of the average MSE of the DeSA-DNN can be predicted[21, 37].

6.2 Parameter tuning and layer-by-layer training

The SE framework is used to analyze the parameter tuning problem of the DeSA-DNN. Note that the symbol of the denoiser is rewritten as $\boldsymbol{D}^t_{\sigma_t, \boldsymbol{w}_t}$. Then according to this notation, the SE is expressed as

$$\gamma^t(\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_t) = \frac{1}{N} E \left\| \boldsymbol{D}^t_{\sigma_t, \boldsymbol{w}_t}(\boldsymbol{h} + \sigma_t \boldsymbol{\epsilon}) - \boldsymbol{h} \right\|_2^2 \tag{12}$$

where $\sigma_t^2 = \frac{\gamma^t(\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_t)}{\delta} + \sigma_{\boldsymbol{n}}^2$. Note that the symbols of the variables in the SE have been changed to imply that γ^t depends on the tuning of the parameters in the previous layers of the DeSA-DNN. For the parameter tuning problem of the DeSA-DNN, we can first refer to the following lemma[21].

Lemma 1 For denoisers $\boldsymbol{D}^1_{\sigma_1,\boldsymbol{w}_1}, \boldsymbol{D}^2_{\sigma_2,\boldsymbol{w}_2}, \ldots, \boldsymbol{D}^T_{\sigma_T,\boldsymbol{w}_T}$, assume that $\inf\left(E\left\|\boldsymbol{D}^t_{\sigma_t,\boldsymbol{w}_t}(\boldsymbol{h}+\sigma_t\boldsymbol{\epsilon})-\boldsymbol{h}\right\|_2^2\right)$ is a non-decreasing function of σ_t. The following procedures are performed: The weight $\boldsymbol{w}_1$ of $\boldsymbol{D}^1_{\sigma_1,\boldsymbol{w}_1}$ is set to make $E\left[\gamma^1\right]$ minimum and fixed; then, the weight $\boldsymbol{w}_2$ of $\boldsymbol{D}^2_{\sigma_2,\boldsymbol{w}_2}$ is set to make $E\left[\gamma^2\right]$ minimum and fixed; and so on; finally, the weight $\boldsymbol{w}_T$ of $\boldsymbol{D}^T_{\sigma_T,\boldsymbol{w}_T}$ is set to make $E\left[\gamma^T\right]$ minimum and fixed. Afterwards, $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_T$ together minimize $E\left[\gamma^T\right]$.

Definition 1 The parameter sequence $\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_t^*$ is called optimal if the following equation holds:

$$\gamma^t\left(\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_t^*\right) = \min \gamma^t(\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_t) \tag{13}$$

which means that the parameter sequence $\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_t^*$ enables the DeSA-DNN to achieve the minimum MSE at the t-th layer.

Inspired by Lemma 1, we adopt layer-by-layer training for the DeSA-DNN to obtain the optimal values of $\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_T$. Firstly, we train a DeSA-DNN with only one layer, tune $\boldsymbol{w}_1$ to minimize $\gamma^1(\boldsymbol{w}_1)$, and fix the optimal $\boldsymbol{w}_1^*$. Then we add a network layer, train the second layer of the resulting two-layer DeSA-DNN, tune $\boldsymbol{w}_2$ to minimize $\gamma^2(\boldsymbol{w}_1, \boldsymbol{w}_2)$, and fix the optimal $\boldsymbol{w}_1^*$ and $\boldsymbol{w}_2^*$. Repeat this procedure until the T-layer of the DeSA-DNN is trained, and finally $\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_T^*$ together are obtained to minimize γ^T, i.e., $\gamma^T = \gamma^T\left(\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_T^*\right)$. The following theorem guarantees that the parameter tuning strategy of layer-by-layer training is MMSE optimal for the DeSA-DNN[21, 38].

Theorem 1 Suppose that the denoiser $\boldsymbol{D}^t_{\sigma_t,\boldsymbol{w}_t}$ is monotonic in the sense that $\inf\left(E\left\|\boldsymbol{D}^t_{\sigma_t,\boldsymbol{w}_t}(\boldsymbol{h}+\sigma_t\boldsymbol{\epsilon})-\boldsymbol{h}\right\|_2^2\right)$ is a non-decreasing function of σ_t. If $\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_t^*$ are obtained according to the layer-by-layer training, then we have $\gamma^t\left(\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_t^*\right) \leqslant \gamma^t(\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_t), \forall \boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_t$.

Proof The proof is based on mathematical induction. Certainly for the first step of the proof procedure, it is natural that we have

$$\gamma^1\left(\boldsymbol{w}_1^*\right) \leqslant \gamma^1(\boldsymbol{w}_1),\ \forall \boldsymbol{w}_1 \tag{14}$$

Now suppose Formula (14) holds for any $T \geqslant 1$, and we would like to prove that the result also holds for $T+1$, i.e.,

$$\gamma^{T+1}\left(\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_{T+1}^*\right) \leqslant \gamma^{T+1}(\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{T+1}), \quad \boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{T+1} \tag{15}$$

Suppose that this does not hold, then there exists $\boldsymbol{w}_1', \boldsymbol{w}_2', \ldots, \boldsymbol{w}_{T+1}'$, such that

$$\gamma^{T+1}\left(\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_{T+1}^*\right) > \gamma^{T+1}\left(\boldsymbol{w}_1', \boldsymbol{w}_2', \ldots, \boldsymbol{w}_{T+1}'\right) \tag{16}$$

Clearly,

$$\gamma^{T+1}\left(\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_{T+1}^*\right) = \frac{1}{N} E\left\|\boldsymbol{D}^{T+1}_{\sigma^*_{T+1},\boldsymbol{w}^*_{T+1}}(\boldsymbol{h}+\sigma^*_{T+1}\boldsymbol{\epsilon})-\boldsymbol{h}\right\|_2^2 \tag{17}$$

where $\left(\sigma^*_{T+1}\right)^2 = \frac{\gamma^{T+1}\left(\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_{T+1}^*\right)}{\delta} + \sigma_{\boldsymbol{n}}^2$. Define $\left(\sigma'_{T+1}\right)^2 = \frac{\gamma^{T+1}\left(\boldsymbol{w}_1', \boldsymbol{w}_2', \ldots, \boldsymbol{w}_{T+1}'\right)}{\delta} + \sigma_{\boldsymbol{n}}^2$, then $\sigma^*_{T+1} \leqslant \sigma'_{T+1}$ is obtained according to the induction assumption. Thus, according to the monotonicity of the denoiser,

$$\begin{aligned}
&\gamma^{T+1}\left(\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_{T+1}^*\right) = \\
&\frac{1}{N} E\left\|\boldsymbol{D}^{T+1}_{\sigma^*_{T+1},\boldsymbol{w}^*_{T+1}}(\boldsymbol{h}+\sigma^*_{T+1}\boldsymbol{\epsilon})-\boldsymbol{h}\right\|_2^2 = \\
&\inf \frac{1}{N} E\left\|\boldsymbol{D}^{T+1}_{\sigma^*_{T+1},\boldsymbol{w}_{T+1}}(\boldsymbol{h}+\sigma^*_{T+1}\boldsymbol{\epsilon})-\boldsymbol{h}\right\|_2^2 \leqslant \\
&\inf \frac{1}{N} E\left\|\boldsymbol{D}^{T+1}_{\sigma'_{T+1},\boldsymbol{w}_{T+1}}\left(\boldsymbol{h}+\sigma'_{T+1}\boldsymbol{\epsilon}\right)-\boldsymbol{h}\right\|_2^2 \leqslant \\
&\frac{1}{N} E\left\|\boldsymbol{D}^{T+1}_{\sigma'_{T+1},\boldsymbol{w}'_{T+1}}\left(\boldsymbol{h}+\sigma'_{T+1}\boldsymbol{\epsilon}\right)-\boldsymbol{h}\right\|_2^2 = \\
&\gamma^{T+1}\left(\boldsymbol{w}_1', \boldsymbol{w}_2', \ldots, \boldsymbol{w}_{T+1}'\right)
\end{aligned} \tag{18}$$

Formula (18) is contradictory to Formula (16), thus

$$\gamma^{T+1}\left(\boldsymbol{w}_1^*, \boldsymbol{w}_2^*, \ldots, \boldsymbol{w}_{T+1}^*\right) \leqslant \gamma^{T+1}(\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{T+1}), \quad \boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{T+1} \tag{19}$$

7 Simulation results and discussion

7.1 Simulation parameters setting

We evaluate the performance of the proposed SA-DNN and DeSA-DNN schemes through extensive simulations. The main simulation parameters set up for a typical UA-OFDM based system are given in Table 1.

The support of the CIR of the UAC with several

Table 1 Simulation parameters.

Parameter	Value
Carrier frequency	12 kHz
Bandwidth	8 kHz
OFDM size	1024
CP length	256
Modulation type	BPSK
Number of pilots	64 and 128
Channel length	256

nonzero paths is randomly generated according to the statistical distribution of the UAC model described in Ref. [39]. The path delays have a maximum of 256 and the path amplitudes are Rayleigh distributed with the average power decaying exponentially with the delay. Note that the channel parameters remain approximately constant within the duration of each OFDM block[34].

The training dataset $\left\{(\boldsymbol{Y}^d, \boldsymbol{h}^d)\right\}_{d=1}^{D}$ with size D = 2000 is randomly generated, and the test dataset is generated in a similar way in Ref. [39]. In the training stage, we trained both the SA-DNN and DeSA-DNN layer-by-layer utilizing the Adam optimizer of the TensorFlow framework with a learning rate of 0.001. After several training epochs, the optimal number of layers of the SA-DNN converged to $T=4$, and the learned parameters $\boldsymbol{\Theta}=\left\{\{\boldsymbol{B}_t\}_{t=1}^4, \{\lambda_t\}_{t=1}^4\right\}$ are obtained. For the DeSA-DNN, we have T = 6 and $\boldsymbol{\Theta}=\left\{\{\boldsymbol{w}_t\}_{t=1}^6\right\}$.

We evaluate the performance of the proposed SA-DNN and DeSA-DNN schemes and compare it with several state-of-the-art methods of UAC estimation as benchmark schemes: traditional LS method, iterative sparse recovery algorithm of AMP, and the CS-based greedy algorithms of OMP, SAMP, and CoSaMP. Among them, default parameters are adopted for LS, OMP, and CoSaMP; the maximum iteration number of AMP is set to 30 to ensure the convergence of the estimation result; the step size of SAMP is set to 1.

7.2 Simulation results and analysis

The performance of the proposed SA-DNN and DeSA-DNN schemes are evaluated in terms of the metrics, including the Normalized MSE (NMSE) and the Probability of Successful Estimation (PSE). The NMSE is defined as $\mathrm{NMSE}=\|\boldsymbol{h}-\hat{\boldsymbol{h}}\|_2^2/\|\boldsymbol{h}\|_2^2$, where $\hat{\boldsymbol{h}}$ denotes the estimation result, and $\|\cdot\|_2$ denotes the l_2 norm. The PSE is defined as the probability that the NMSE of the estimation result is less than a predefined threshold of 0.01, i.e., NMSE < 0.01.

Figure 4 compares the NMSE performance of different UAC estimation methods with respect to different SNR values using 64 pilots for channel estimation. The result shows that the proposed SA-DNN and DeSA-DNN schemes outperform the benchmark schemes in estimation accuracy, especially in the low SNR region. At the target NMSE level of 10^{-2}, the SA-DNN scheme has an SNR gain of 2.7 dB, 3.9 dB and 5.5 dB over the OMP, CoSaMP, and AMP methods, respectively. In addition, the DeSA-DNN scheme exploits the denoiser to reduce the estimation error caused by Gaussian noise, which results in a nearly 0.5 dB to 1 dB improvement in accuracy compared to the SA-DNN scheme. The simulation results demonstate that the DNN in the proposed schemes has successfully extracted the sparse structure of the UAC, and also verify the superior performance of the proposed schemes.

Figure 5 shows the NMSE performance comparison of different UAC estimation methods with respect to the number of available pilots at the SNR of 15 dB. It can be observed that, compared with the benchmark schemes of LS, OMP, AMP, SAMP, and CoSaMP, the proposed SA-DNN and DeSA-DNN schemes can achieve a higher accuracy of UAC estimation with much less pilot overhead, so as to significantly improve the spectrum efficiency.

Figure 6 shows the PSE performance of different

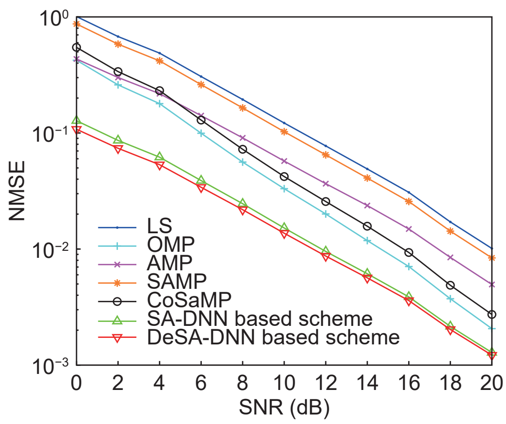

Fig. 4 NMSE performance of different UAC estimation methods with respect to different SNR values.

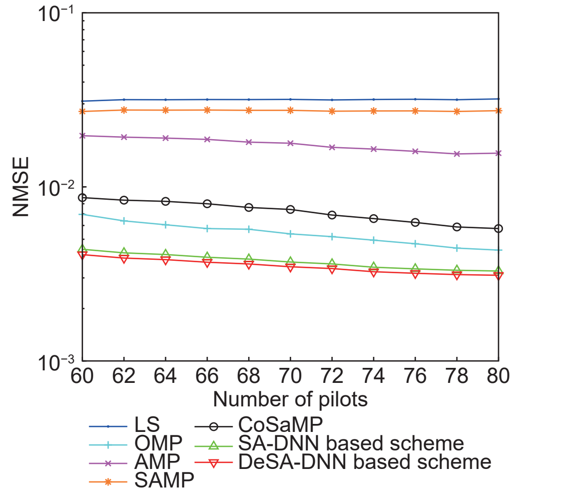

Fig. 5 NMSE performance of different UAC estimation methods with respect to the number of available pilots.

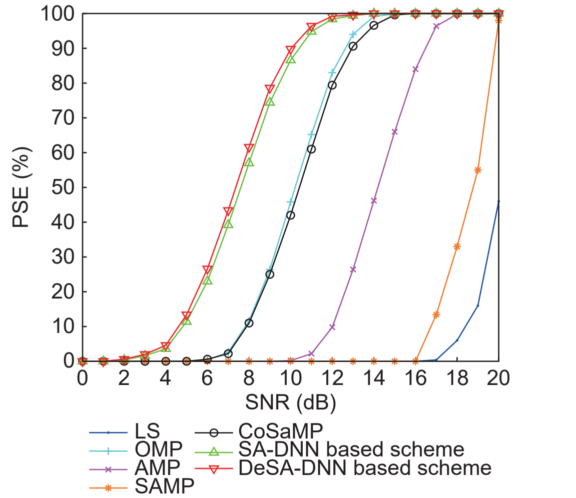

Fig. 6 Result of probability of successful estimation with respect to SNR.

UAC estimation methods under different SNR values using 128 pilots for channel estimation. The result illustrates that the proposed SA-DNN and DeSA-DNN schemes can achieve a higher probability of successful estimation at the same SNR level compared to the benchmark schemes, especially in the case of intensive ambient noise. Figure 7 shows the PSE performance of different UAC estimation methods with different number of available pilots at the SNR of 15 dB. The result shows that, compared with the benchmark schemes, the proposed schemes can achieve a higher probability of successful estimation at the same number of available pilots. The simulation results of Figs. 6 and 7 demonstate that the proposed schemes have better robustness than the benchmark schemes.

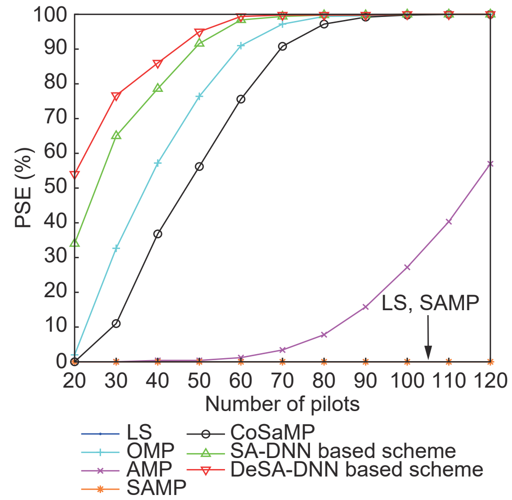

Fig. 7 PSE performance of different UAC estimation methods with respect to the number of available pilots.

8 Conclusion

In this paper, we propose two novel DL-based schemes for UAC estimation based on the devised model-driven deep unfolding architectures of SA-DNN and DeSA-DNN. In order to fully make use of the inherent sparsity property of the underwater acoustic channels for an improved inference precision of the channel state information, this paper utilized a model-driven architecture of deep unfolding networks that mimics the iterations of the classical iterative sparse reconstruction algorithm using network layers described by distinct learnable parameters. Moreover, an effective denoising module incorporating the convolutional neural networks is introduced to the network, which can effectively eliminate the influence of intensive interference on the channel estimation performance. Simulation results have validated that the proposed schemes significantly outperform the benchmark schemes if we look at various metrics including the inference precision, the pilot consumption, and robustness, particularly in severe conditions such as intensive noise or lack of pilots.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (No. 61901403), the

Science and Technology Key Project of Fujian Province, China (Nos. 2021HZ021004 and 2019HZ020009), the Open Research Fund of National Mobile Communications Research Laboratory, Southeast University (No. 2023D10), the Youth Innovation Fund of Natural Science Foundation of Xiamen (No. 3502Z20206039), the Science and Technology Key Project of Xiamen (No. 3502Z20221027), and the Xiamen Special Fund for Marine and Fishery Development (No. 21CZB011HJ02).

References

[1] A. C. Singer, J. K. Nelson, and S. S. Kozat, Signal processing for underwater acoustic communications, *IEEE Commun. Mag.*, vol. 47, no. 1, pp. 90–96, 2009.

[2] S. Banerjee and M. Agrawal, Underwater acoustic noise with generalized Gaussian statistics: Effects on error performance, in *Proc. 2013 MTS/IEEE OCEANS - Bergen*, Bergen, Norway, 2013, pp. 1–8.

[3] P. Chen, Y. Rong, S. Nordholm, Z. He, and A. J. Duncan, Joint channel estimation and impulsive noise mitigation in underwater acoustic OFDM communication systems, *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 6165–6178, 2017.

[4] A. Radosevic, R. Ahmed, T. M. Duman, J. G. Proakis, and M. Stojanovic, Adaptive OFDM modulation for underwater acoustic communications: Design considerations and experimental results, *IEEE J. Oceanic Eng.*, vol. 39, no. 2, pp. 357–370, 2014.

[5] E. Panayirci, H. Senol, M. Uysal, and H. V. Poor, Sparse channel estimation and equalization for OFDM-based underwater cooperative systems with amplify-and-forward relaying, *IEEE Trans. Signal Process.*, vol. 64, no. 1, pp. 214–228, 2016.

[6] Y. Zhang, R. Venkatesan, O. A. Dobre, and C. Li, Efficient estimation and prediction for sparse time-varying underwater acoustic channels, *IEEE J. Oceanic Eng.*, vol. 45, no. 3, pp. 1112–1125, 2020.

[7] C. R. Berger, S. Zhou, J. C. Preisig, and P. Willett, Sparse channel estimation for multicarrier underwater acoustic communication: From subspace methods to compressed sensing, *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1708–1721, 2010.

[8] K. S. Priyanjali and A. V. Babu, An improved least square channel estimation technique for OFDM systems in sparse underwater acoustic channel, in *Proc. 2014 Int. Conf. on Advances in Computing, Communications and Informatics* (*ICACCI*), Delhi, India, 2014, pp. 2521–2525.

[9] Y. Zhang, H. Sun, F. Xu, and D. Wang, OFDM transform-domain channel estimation based on MMSE for underwater acoustic channels, in *Prce. 2008 2nd Int. Conf. on Anti-Counterfeiting, Security and Identification*, Guiyang, China, 2008, pp. 177–181.

[10] W. Lu, T. Dai, and S. T. Xia, Binary matrices for compressed sensing, *IEEE Trans. Signal Process.*, vol. 66, no. 1, pp. 77–85, 2018.

[11] L. Wan, X. Qiang, L. Ma, Q. Song, and G. Qiao, Accurate and efficient path delay estimation in OMP based sparse channel estimation for OFDM with equispaced pilots, *IEEE Wireless Commun. Lett.*, vol. 8, no. 1, pp. 117–120, 2019.

[12] G. Qiao, Q. Song, L. Ma, and L. Wan, A low-complexity orthogonal matching pursuit based channel estimation method for time-varying underwater acoustic OFDM systems, *Appl. Acoust.*, vol. 148, pp. 246–250, 2019.

[13] F. Qu, X. Nie, and W. Xu, A two-stage approach for the estimation of doubly spread acoustic channels, *IEEE J. Oceanic Eng.*, vol. 40, no. 1, pp. 131–143, 2015.

[14] S. Liu, F. Yang, W. Ding, X. Wang, and J. Song, Two-dimensional structured-compressed-sensing-based NBI cancelation exploiting spatial and temporal correlations in MIMO systems, *IEEE Trans. Veh. Technol.*, vol. 65, no. 11, pp. 9020–9028, 2016.

[15] Y. Chen, C. Clemente, J. Soraghan, and S. Weiss, Fractional Fourier based sparse channel estimation for multicarrier underwater acoustic communication system, in *Proc. 2016 Sensor Signal Processing for Defence* (*SSPD*), Edinburgh, UK, 2016, pp. 1–5.

[16] T. T. Do, L. Gan, N. Nguyen, and T. D. Tran, Sparsity adaptive matching pursuit algorithm for practical compressed sensing, in *Proc. 2008 42nd Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, CA, USA, 2008, pp. 581–587.

[17] Y. Zhang, R. Venkatesan, O. A. Dobre, and C. Li, An adaptive matching pursuit algorithm for sparse channel estimation, in *Proc. 2015 IEEE Wireless Communications and Networking Conf.* (*WCNC*), New Orleans, LA, USA, 2015, pp. 626–630.

[18] J.Wang, Z. Yan, W. Shi, and X. Yang, Underwater acoustic sparse channel estimation based on DW-SACoSaMP reconstruction algorithm, *IEEE Commun. Lett.*, vol. 23, no. 11, pp. 1985–1988, 2019.

[19] S. Liu, F. Yang, W. Ding, and J. Song, Double kill: Compressive-sensing-based narrow-band interference and impulsive noise mitigation for vehicular communications, *IEEE Trans. Veh. Technol.*, vol. 65, no. 7, pp. 5099–5109, 2016.

[20] M. Borgerding, P. Schniter, and S. Rangan, AMP-inspired deep networks for sparse linear inverse problems, *IEEE*

Trans. Signal Process., vol. 65, no. 16, pp. 4293–4308, 2017.

[21] C. A. Metzler, A. Mousavi, and R. G. Baraniuk, Learned D-AMP: Principled neural network based compressive image recovery, in *Proc. 31st Int. Conf. Neural Information Processing Systems* (*NIPS'17*), Long Beach, CA, USA, 2017, pp. 1772–1783.

[22] H. He, C. K. Wen, S. Jin, and G. Y. Li, Deep learning-based channel estimation for beamspace mmwave massive MIMO systems, *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 852–855, 2018.

[23] X. Wei, C. Hu, and L. Dai, Deep learning for beamspace channel estimation in millimeter-wave massive MIMO systems, *IEEE Trans. Commun.*, vol. 69, no. 1, pp. 182–193, 2021.

[24] Y. Xiao, G. Niu, L. Xiao, Y. Ding, S. Liu, and Y. Fan, Reinforcement learning based energy-efficient Internet-of-things video transmission, *Intelligent and Converged Networks*, vol. 1, no. 3, pp. 258–270, 2020.

[25] S. Nath and J. Wu, Deep reinforcement learning for dynamic computation offloading and resource allocation in cache-assisted mobile edge computing systems, *Intelligent and Converged Networks*, vol. 1, no. 2, pp. 181–198, 2020.

[26] J. Huang, C. R. Berger, S. Zhou, and J. Huang, Comparison of basis pursuit algorithms for sparse channel estimation in underwater acoustic OFDM, in *Proc. Oceans'10 IEEE Sydney*, Sydney, Australia, 2010, pp. 1–6.

[27] D. L. Donoho, A. Maleki, and A. Montanari, Message passing algorithms for compressed sensing: II. analysis and validation, in *Proc. 2010 IEEE Information Theory Workshop on Information Theory* (*ITW 2010, Cairo*), Cairo, Egypt, 2010, pp. 1–5.

[28] Y. Yin, S. Liu, G. Qiao, Y. Yang, and Y. Yang, OFDM demodulation using virtual time reversal processing in underwater acoustic communications, *J. Comput. Acoust.*, vol. 23, no. 4, p. 1540011, 2015.

[29] M. Nassar, P. Schniter, and B. L. Evans, A factor graph approach to joint OFDM channel estimation and decoding in impulsive noise environments, *IEEE Trans. Signal Process.*, vol. 62, no. 6, pp. 1576–1589, 2014.

[30] J. Dong, W. Wu, Y. Gao, X. Wang, and P. Si, Deep reinforcement learning based worker selection for distributed machine learning enhanced edge intelligence in internet of vehicles, *Intelligent and Converged Networks*, vol. 1, no. 3, pp. 234–242, 2020.

[31] B. Hartpence and A. Kwasinski, CNN and MLP neural network ensembles for packet classification and adversary defense, *Intelligent and Converged Networks*, vol. 2, no. 1, pp. 66–82, 2021.

[32] M. Zhou, J. Wang, H. Sun, J. Qi, X. Feng, and H. Esmaiel, A novel DNN based channel estimator for underwater acoustic communications with IM-OFDM, in *Proc. 2020 IEEE Int. Conf. Signal Processing, Communications and Computing* (*ICSPCC*), Macau, China, 2020, pp. 1–6.

[33] B. Li, S. Zhou, M. Stojanovic, L. Freitag, and P. Willett, Multicarrier communication over underwater acoustic channels with nonuniform Doppler shifts, *IEEE J. Oceanic Eng.*, vol. 33, no. 2, pp. 198–209, 2008.

[34] P. Qarabaqi and M. Stojanovic, Statistical characterization and computationally efficient modeling of a class of underwater acoustic communication channels, *IEEE J. Oceanic Eng.*, vol. 38, no. 4, pp. 701–717, 2013.

[35] M. Borgerding and P. Schniter, Onsager-corrected deep learning for sparse linear inverse problems, in *Proc. 2016 IEEE Global Conf. Signal and Information Processing* (*GlobalSIP*), Washington, DC, USA, 2016, pp. 227–231.

[36] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising, *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, 2017.

[37] C. A. Metzler, A. Maleki, and R. G. Baraniuk, From denoising to compressed sensing, *IEEE Trans. Inform. Theory*, vol. 62, no. 9, pp. 5117–5144, 2016.

[38] S. Ramani, T. Blu, and M. Unser, Monte-Carlo sure: A black-box optimization of regularization parameters for general denoising algorithms, *IEEE Trans. Image Process.*, vol. 17, no. 9, pp. 1540–1554, 2008.

[39] S. Zhou and Z. Wang, *OFDM for Underwater Acoustic Communications*. Hoboken, NJ, USA: John Wiley & Sons, Ltd, 2014.

Xianyao Wang received the BEng degree in communication engineering from Ningxia University, China in 2021. He is currently a master student at the Department of Information and Communication Engineering, Xiamen University, Xiamen, China. His research interests include visible light positioning, deep learning, and compressed sensing.

Danping Su received the BEng degree from Zhejiang Normal University, China in 2020. She is currently a master student in communication engineering at Xiamen University, Fujian, China. Her current research interests include deep learning, visible light communication, and its applications to the field of indoor positioning.

Sicong Liu received the BEng and PhD degrees in electronic engineering both from Tsinghua University, Beijing, China in 2012 and 2017 with the highest honor, respectively. He is currently an associate professor at the Department of Information and Communication Engineering, School of Informatics, Xiamen University, China. He was a senior engineer at Huawei Technologies Co. Ltd. between 2017 and 2018. He was a visiting scholar at City University of Hong Kong in 2010. His current research interests are compressed sensing, AI-assisted communications, integrated sensing and communications, and visible light communications. He has authored over 60 journal and conference papers, and four monographs in the related areas. He won the best paper award at the ACM UbiComp 2021 CPD WS. He has served on the editorial board or as a chair in the technical program committee of several IEEE and other international academic journals and conferences. He is a senior member of IEEE, and a senior member of China Institute of Communications (CIC).

Younan Mou received the BEng degree in communication engineering from Xi’an University of Science and Technology, Xi’an, China in 2021. He is currently a master student at the Department of Information and Communication Engineering, Xiamen University, Xiamen, China. His research interests include visible light communications, deep learning, and compressed sensing.

Ling Cheng received the BEng degree from Huazhong University of Science and Technology in 1995, and the MSc degree in 2005, and the PhD degree in 2011, both from University of Johannesburg (UJ), South Africa. His research interests are in telecommunications and artificial intelligence. In 2010, he joined University of the Witwatersrand where he was promoted to a full professor in 2019. He serves as an associate editor of three journals. He has published more than one hundred research papers in journals and conference proceedings. He has been a visiting professor at five universities and the principal advisor for over forty full research post-graduate students. He was awarded the Chancellor’s medals in 2005 and 2019, and the South Africa National Research Foundation Ratings in 2014 and 2020. The IEEE ISPLC 2015 best student paper award was made to his PhD student in Austin. He is a senior member of IEEE and the vice-chair of IEEE South African Information Theory Chapter.