Predicting transmission loss in underwater acoustics using convolutional recurrent autoencoder network

Wrik Mallik, Rajeev K. Jaiman and Jasmin Jelovica

Citation: The Journal of the Acoustical Society of America **152**, 1627 (2022); doi: 10.1121/10.0013894

View online: https://doi.org/10.1121/10.0013894

View Table of Contents: https://asa.scitation.org/toc/jas/152/3

Published by the Acoustical Society of America

JASA ARTICLE

Predicting transmission loss in underwater acoustics using convolutional recurrent autoencoder network

Wrik Mallik, Rajeev K. Jaiman,[a] and Jasmin Jelovica
Department of Mechanical Engineering, University of British Columbia, Vancouver, British Columbia V5T 1Z4, Canada

ABSTRACT:
Underwater noise transmission in the ocean environment is a complex physical phenomenon involving not only widely varying physical parameters and dynamical scales but also uncertainties in the ocean parameters. It is challenging to construct generalized physical models that can predict transmission loss in a broad range of situations. In this regard, we propose a convolutional recurrent autoencoder network (CRAN) architecture, which is a data-driven deep learning model for learning far-field acoustic propagation. Being data-driven, the CRAN model relies only on the quality of the data and is agnostic to how the data are obtained. The CRAN model can learn a reduced-dimensional representation of physical data and can predict the far-field acoustic signal transmission loss distribution in the ocean environment. We demonstrate the ability of the CRAN model to learn far-field transmission loss distribution in a two-dimensional ocean domain with depth-dependent sources. Results show that the CRAN can learn the essential physical elements of acoustic signal transmission loss generated due to geometric spreading, refraction, and reflection from the ocean surface and bottom. Such ability of the CRAN to learn complex ocean acoustics transmission has the potential for real-time far-field underwater noise prediction for marine vessel decision-making and online control. © *2022 Acoustical Society of America*. https://doi.org/10.1121/10.0013894

(Received 13 April 2022; revised 15 August 2022; accepted 16 August 2022; published online 13 September 2022)
[Editor: Julien Bonnel] Pages: 1627–1638

I. INTRODUCTION

Underwater acoustic propagation in the ocean is a complex wave propagation phenomenon relying on several environmental parameters and complex interaction of acoustic wavefronts with the ocean surface and bottom. Underwater acoustic propagation in the ocean is also a matter of significant concern for both environmentalists and the shipbuilding industry as underwater radiated noise (URN) propagation from marine vessel operations acts as a stressor for underwater marine animals (Duarte *et al.*, 2021; Erbe *et al.*, 2019). Thus, the physical understanding and prediction capability of far-field URN propagation are extremely important for both environmentalists and marine engineers.

Underwater ocean acoustic propagation is analyzed via techniques of varying fidelity according to the nature of the physical problem investigated. As a result, experimental measurements, high-fidelity numerical techniques like the finite-element method, or hybrid numerical techniques like ray tracing all have their necessities according to specific situations. While the physics-based acoustic models can provide useful insight, they are poorly suited to combine different sources and different levels of resolutions. Hence, it would be highly beneficial to obtain a generalized model that could incorporate ocean acoustics information from various sources of data and unify various analysis techniques over a wide range of parameters and physical conditions. Such a generalized model could be employed to analyze underwater ocean acoustic propagation for the development of underwater noise reduction strategies and guidelines. However, it is difficult to develop generalized physical models for underwater ocean acoustics owing to complex multiphysics phenomena, widely varying physical scales, and the uncertainties in several environmental parameters involved (James and Dowling, 2005, 2011).

We propose the application of a deep learning (DL)-based data-driven method for developing a generalized model of underwater ocean acoustic propagation. Being data-driven, such methods rely only on the observed data and remain agnostic to the data generation technique. Therefore, they can approximate the underlying ocean acoustic environment across various scales and environmental conditions and parameters for which data can be obtained from either experiments or computational solvers. Furthermore, these data-driven models can learn a much lower-dimensional representation of the system from high-dimensional physical data. This potentially enhances their scalability and facilitates real-time predictions. Such data-driven models would generally be developed via an offline-online application strategy. The offline phase consists of training the models to learn the problem physics. During the online phase, the data-driven model can provide real-time solutions for various applications, including decision-making, control, and optimization.

A wide range of data-driven techniques has been historically employed for learning complex physical systems exhibiting nonlinear and multi-scale behavior. In recent years, DL architectures based on neural networks have been

[a] Electronic mail: rjaiman@mech.ubc.ca

increasingly used as data-driven models for mechanistic problems (Hsieh, 2009). Recent numerical experiments (Mansour, 2019) indicate that DL models tend to prioritize the learning of inherent simpler features of the data over the complex data patterns. Such learning of simple laws governing the data aligns perfectly with data of mechanistic origin, which are governed by a few fundamental physical principles (e.g., Newton's laws of motion). This underscores the motivation for approximating mechanistic systems with data-driven DL models.

Recent results have shown that the nonlinear mapping developed in DL models via a composition of nonlinear activation functions in multi-layer deep neural networks makes them extremely efficient in learning low-dimensional representations of high-dimensional physical data, especially for wave propagation phenomena (Mallik *et al.*, 2022). Moreover, unlike conventional regression models, certain deep neural network architectures provide geometric priors to their approximation, which can endow specific generalized learning properties (Bronstein *et al.*, 2017). These properties allow them to approximate many complex functions, which has been theoretically demonstrated by various universal approximation theorems (Hornik, 1991; Leshno *et al.*, 1993). This has led to the employment of various DL models like autoencoder networks and recurrent neural networks (RNNs) for accurately learning various nonlinear and complex physical problems in a much lower dimension and with real-time online prediction capability (Bukka *et al.*, 2021; Lee and Carlberg, 2020; Sorteberg *et al.*, 2019).

Recently, various data-driven models, including DL models, have been applied in underwater acoustics for the classification and localization of ship-generated sound sources based on their acoustic signatures in both shallow and deep ocean (Chi *et al.*, 2019; Ferguson, 2021; Huang *et al.*, 2018; Niu *et al.*, 2017). Various DL models have also been recently employed for studying the propagation of shallow water waves (Deo and Jaiman, 2022; Fotiadis *et al.*, 2020). Recent applications of DL models in other areas of acoustic include learning low-dimensional representations of seismic waves (Moseley *et al.*, 2020) and predicting the approach time of seismic waves (Smith *et al.*, 2021). Hybrid physics-based DL approaches are also available (Borrel-Jensen *et al.*, 2021; Mallik *et al.*, 2021). However, such research is still in the developmental stage (Mallik *et al.*, 2021), or the explicit application of partial differential equations (Borrel-Jensen *et al.*, 2021) makes them unsuitable for operating on high-dimensional physical data and for complex systems, where the exact governing equations and the underlying physical process may not be known. The application of DL models for learning various problems in wave propagation is an emerging area of research, especially for predicting far-field URN transmission.

In this research, we develop a convolutional recurrent autoencoder network (CRAN) as our DL architecture for learning far-field underwater acoustic transmission. The CRAN model is a composite encoder-propagator-decoder framework (Bukka *et al.*, 2021), which applies a sequence-to-sequence learning mechanism for autoregressive prediction of far-field propagation of acoustic signals. The learning is performed on a significantly reduced-dimensional subspace, which approximates high-dimensional physical data. The CRAN is employed for learning the spatial distribution of far-field transmission loss from a point source in a two-dimensional underwater ocean environment. We consider a wide range of source depth locations to demonstrate the generalized learning capacity of the CRAN model.

This paper discusses the application of the CRAN model for data-driven learning of far-field URN propagation in an underwater ocean environment. The training strategy of the CRAN model and the generalized learning capacity of underwater acoustic transmission are discussed. Such application of the CRAN model for learning far-field URN propagation is presented here for the first time. Successful application of the CRAN promises future application of such data-driven models to predict far-field underwater noise transmission with a wide range of parameters, based on combined experimental and multi-fidelity computational data sets.

II. DATA-DRIVEN FORMULATION OF ACOUSTIC TRANSMISSION LOSS

Underwater noise transmission is governed by the propagation of acoustic pressure generated from sound sources. In a two-dimensional idealization of the ocean domain, such acoustic pressure can be presented in Cartesian coordinates as $p(z, R; \omega)$. Here, $p(z, R; \omega)$ is a function of the depth z and range R and also relies on the frequency of emitted signal ω. There is a significant loss in the energy, or intensity $I(z, R)$ (where $I \propto p^2$), of the acoustic pressure signal, known as transmission loss (TL). Such TL is a combination of losses due to geometric spreading, losses due to volume attenuation through the medium, and losses due to interaction with the ocean bottom and surface. The losses due to interaction with boundaries include losses due to reflection from the surface and reflection and scattering from the bottom. Such TL can be computed in decibels from the acoustic pressure, $p(z, R)$, at some receiver location (z, R) with respect to the reference signal p_0 at the source as

$$\mathrm{TL}(z, R) = -10 \log \frac{I(z, R)}{I_0} = -20 \log \frac{p(z, R)}{p_0}. \tag{1}$$

Hence, the TL is a measure of the drop in far-field sound pressure level from various sources of URN.

For most practical situations in ocean acoustics, the ocean range considered for far-field URN measurement is usually 1–2 orders larger than the ocean depth. Thus, we will assume the data-driven far-field TL prediction as a situation where we autoregressively compute the far-field TL along the ocean range in a sequential manner starting from known near-field observations. This will ensure that very little initial information is required for the data-driven prediction. For a full-order (high-dimensional) acoustic modeling, this can be represented by a partial differential equation governing the evolution of an observable $\mathbf{U}$, which depends on spatial coordinates $\mathbf{x} \in \mathbb{R}^i (i = 1, 2, 3)$, time t, and a set of real-valued parameter $\phi \geq 0$ as

$$\frac{\partial}{\partial t}\mathbf{U}(\mathbf{x}, t; \phi) = \mathcal{F}(\mathbf{U}(\mathbf{x}, t; \phi); \phi). \tag{2}$$

Equation (2) can be subject to any general initial and boundary conditions. $\mathcal{F}$ is any general nonlinear operator governing the dynamics of the system and provides a forward map from the causality (initial and boundary conditions, system properties, and dynamics) to the effects (final solution). On the other hand, the data-driven prediction is performed as

$$\mathbf{U}_{M,k+1}(\mathbf{x}_M, t_{K+1}; \phi) = \mathcal{G}\big(\mathbf{U}_{M,k}(\mathbf{x}_M, t_K; \mathbf{\Theta}); \phi\big), \tag{3}$$

for $k = 0, 1, \ldots, K-1$. Here, $\mathbf{U}_{M,k} \in \mathbb{R}^M$ is the discrete solution with M spatial discretizations, and K represents the discrete time steps. The data-driven operator $\mathcal{G}$ relies on trainable parameters $\mathbf{\Theta}$, which must be learned based on available data $\mathbf{U}_{M,k}$ for a known set of parameters ϕ. On successfully learning $\mathcal{G}$, it provides an inverse map of the system, i.e., learning causality from the effects. $\mathcal{G}$ can then be used to predict the solution at the $k+1$th range-wise location if the solution at the kth range-wise location is provided. Furthermore, the data-driven operator $\mathcal{G}$ can be learned on a low-dimensional representation of the high-dimensional physical observable $\mathbf{U}$, to improve the scalability of the data-driven prediction.

Without any loss of generality, we can rewrite Eq. (2) for TL propagation along the ocean range as

$$\frac{\partial}{\partial R}\mathrm{TL}(z, R; \phi) = \mathcal{F}(\mathrm{TL}(z, R; \phi); \phi), \quad z, R \in \mathbb{R}, \tag{4}$$

where ϕ represents the set of parameters including boundary conditions and domain and source properties. Similarly, following Eq. (3), the discrete TL, $\mathrm{TL}_{M,k}$, can be predicted in a data-driven manner as

$$\mathrm{TL}(z, R_{k+1}; \phi) = \mathcal{G}(\mathrm{TL}(z, R_k; \phi); \mathbf{\Theta}), \tag{5}$$

once the inverse operator $\mathcal{G}$ is learned in a data-driven manner. Autoregressive application of $\mathcal{G}$ can thus predict the far-field TL from a point source over K iterations. We present the data-driven learning methodology of the system evolution in low dimensions in Sec. III.

III. DATA-DRIVEN LEARNING METHODOLOGY

As discussed in Sec. II, the objective of data-driven learning is to obtain the operator $\mathcal{G}$ such that any physically observed field variable (pressure, velocity, etc.) can be propagated in either time or along a spatial direction. Furthermore, we want to learn $\mathcal{G}$ on a reduced dimension for scalability. For any general high-dimensional observed variable of interest $\mathbf{U}_{M,k} \in \mathbb{R}^M$, obtained at propagation step k, the data-driven learning problem can be formulated as the learning of three operators, $\mathcal{E}$, $\mathcal{P}$, and $\mathcal{D}$,

$$\begin{aligned}
\mathbf{A}_{n,k}(\mathbf{x}_M, t_k; \phi) &= \mathcal{E}\big(\mathbf{U}_{M,k}(\mathbf{x}_M, t_k; \phi); \boldsymbol{\theta}_{\mathcal{E}}\big),\\
\mathbf{A}_{n,k+1}(\mathbf{x}_M, t_{k+1}; \phi) &= \mathcal{P}\big(\mathbf{A}_{n,k}(\mathbf{x}_M, t_k; \phi); \boldsymbol{\theta}_{\mathcal{P}}\big),\\
\tilde{\mathbf{U}}_{M,k+1}(\mathbf{x}_M, t_{k+1}; \phi) &= \mathcal{D}\big(\mathbf{A}_{n,k+1}(\mathbf{x}_M, t_{k+1}; \phi); \boldsymbol{\theta}_{\mathcal{D}}\big).
\end{aligned} \tag{6}$$

Here, $\tilde{\mathbf{U}}_{M,k+1} \in \mathbb{R}^M$ and $\mathbf{A}_{n,k} \in \mathbb{R}^n$, for all $k = 0, 1, \ldots, K-1$. On successfully learning the inverse map in low dimensions, we assume $\tilde{\mathbf{U}}_{M,k+1} \approx \mathbf{U}_{M,k+1}$ for $n \ll M$. Here, the operator $\mathcal{E}$ compresses the high-dimensional system to a low-dimensional spatial subspace, $\mathcal{P}$ propagates the low-dimensional system, and $\mathcal{D}$ expands the low-dimensional evolved solution to the original high dimension. Thus, $\mathcal{G}$ can be considered a composition of these three operators,

$$\mathcal{G} = (\mathcal{E} \circ \mathcal{P} \circ \mathcal{D}; \mathbf{\Theta}), \tag{7}$$

where $\mathbf{\Theta} = (\boldsymbol{\theta}_{\mathcal{E}}, \boldsymbol{\theta}_{\mathcal{P}}, \boldsymbol{\theta}_{\mathcal{D}})$. Here, we will consider learning the low-dimensional spatial representation and the evolution of the low-dimensional system as separate learning problems. These are subsequently discussed in this section in detail.

A. Learning low-dimensional spatial representation

DL models, composed of deep neural networks, attempt to find the best low-dimensional nonlinear subspace, which can represent the high-dimensional system. According to the manifold hypothesis, low-dimensional subspaces remain embedded within high-dimensional physical space. Therefore, identifying such low-dimensional subspaces via data-driven models is possible with the correct choice of the data-driven model. Here, we employ a DL autoencoder model for obtaining the low-dimensional spatial representation. The autoencoder consists of an encoder, which compresses high-dimensional spatial data to a much lower-dimensional set of latent states. The latent states can be subsequently expanded to their high-dimensional representation by the decoder. The autoencoder is a combination of the encoder and the decoder and is learned in a semi-supervised manner, as we specify the input and output to the autoencoder but do not supervise how the latent states are learned.

The DL architecture considered here for both the encoder and the decoder is a convolutional neural network (CNN). CNN is specifically selected as its convolution and pooling operations can provide translational equivariance and translational invariance, respectively, to the mapping from high-dimensional physical space to low-dimensional latent states. Thus, unlike regular regression models, the CNN can employ various geometric priors (Bronstein *et al.*, 2017).

The dimension-reduction objective is to find the lowest-dimensional subspace of dimension n such that $\tilde{\mathbf{U}}_{M,K} \approx \mathbf{U}_{M,K}$. It is important to note that the choice of convolutional autoencoder is based on its efficiency for learning low-dimensional subspaces, especially in the presence of discontinuities and convection-dominated problems. Results shown in Mallik *et al.* (2022) comparing the efficiency of convolutional autoencoders to popular projection-based methods like proper orthogonal decomposition corroborate this choice.

B. Learning system evolution

The data-driven learning of the system evolution presented in Eq. (6) is posed as a sequence-to-sequence learning problem. To achieve this, we employ a deeply stacked single-shot long short-term memory (SS-LSTM) network. LSTMs are gated RNNs routinely used for accurately learning sequences with a long-term data dependency. The gating mechanism of LSTMs provides them invariance to time warping. Thus, they are significantly less affected by vanishing gradients compared to non-gated RNNs.

A single LSTM cell consists of the input gate, the output gate, and the forget gate. The cell input, the cell state, and the cell output are denoted by $\boldsymbol{a}$, $\boldsymbol{c}$, and $\boldsymbol{h}$, respectively. The cell output is then passed to a fully connected layer with a linear activation to keep the input and output ($\boldsymbol{y}$) dimensions consistent. The operation of the LSTM cell can be explained via the following equations:

$$\begin{aligned}
\boldsymbol{f}_t &= \sigma\big(\boldsymbol{W}_f \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{a}_t] + \boldsymbol{b}_f\big),\\
\boldsymbol{i}_t &= \sigma(\boldsymbol{W}_i \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{a}_t] + \boldsymbol{b}_i),\\
\tilde{\boldsymbol{c}}_t &= \tanh(\boldsymbol{W}_c \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{a}_t] + \boldsymbol{b}_c),\\
\boldsymbol{c}_t &= \boldsymbol{f}_t * \boldsymbol{c}_{t-1} + \boldsymbol{i}_t * \tilde{\boldsymbol{c}}_t,\\
\boldsymbol{o}_t &= \sigma(\boldsymbol{W}_o \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{a}_t] + \boldsymbol{b}_o),\\
\boldsymbol{h}_t &= \boldsymbol{o}_t * \tanh(\boldsymbol{c}_t),
\end{aligned} \tag{8}$$

where $\boldsymbol{i}, \boldsymbol{f}$, and $\tilde{\boldsymbol{c}}$ represent the input gate, the forget gate, and the updated cell state, respectively. $\boldsymbol{W}$ and $\boldsymbol{b}$ represent the weights and biases for each of the gates, respectively. σ represents the sigmoid function. An illustration of the LSTM cell structure can be obtained in Bukka *et al.* (2021). On successful learning, we assume $\boldsymbol{y}_t \approx \boldsymbol{a}_{t+1}$. LSTM cells can be stacked together to learn how a present sequence of observables evolves into a future sequence of observables. In the SS-LSTM network, a single-shot learning mechanism is evoked as the network memorizes all the states of the current sequence length and predicts the complete output sequence at one shot. An illustration of the SS-LSTM network is shown in Fig. 1.

C. Convolutional recurrent autoencoder architecture

The two separate data-driven learning tasks discussed previously are performed via the composite CRAN, as shown in Fig. 2. Here, we present a three-layer convolutional encoder $\mathcal{E}$ and decoder $\mathcal{D}$. The encoder and decoder presented here are generated with one-dimensional (1D) convolutional kernels, which operate on 1D inputs. All convolution layers except the last one are followed by a max-pooling kernel. The max-pooling layer calculates the maximum value from patches of the convolution layer feature map. Both the convolution and max-pooling operation in the encoder setting result in dimension reduction of the data set, while providing translation equivariance and invariance properties, respectively, to the encoder mapping (Bronstein *et al.*, 2017). The deconvolution and upsampling operation expand the data set in an exact reverse manner to the convolution and max-pooling layers, respectively. The convolution encoding and decoding can also be performed in an analogous manner for higher-dimensional Euclidean data sets. The LSTM propagator $\mathcal{P}$ presented here can be any general LSTM model, although we will employ the SS-LSTM in the present study.

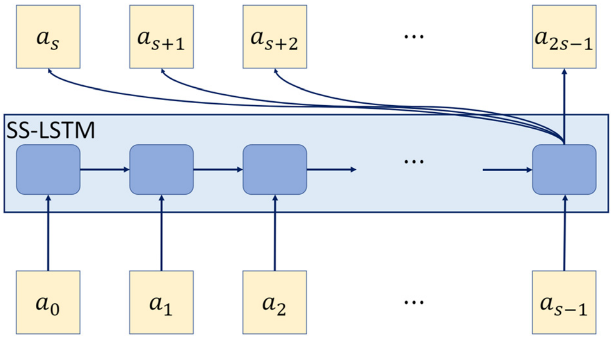


FIG. 1. (Color online) Illustration of sequence-to-sequence learning via SS-LSTM.

We have separated the data-driven learning task into a lower-dimensional spatial representation learning task and a system evolution learning task. The convolutional autoencoder learns the low-dimensional representation of the high-dimensional physical data, and the LSTM network learns the system evolution. These are trained separately. On successful training of these individual components, the CRAN model is employed during the prediction phase on cases not considered for training, in a sequence-to-sequence manner while operating on a low-dimensional subspace. For a sequence containing S units, we employ Eq. (6) with our initial sequence $\mathbf{U}_{M,S_0}$ to obtain $\tilde{\mathbf{U}}_{M,S_1}$, where $\mathbf{U}_{M,S_0} \in \mathbb{R}^{M\times S}$. Feeding back $\tilde{\mathbf{U}}_{M,S_1}$, we can obtain $\tilde{\mathbf{U}}_{M,S_2}$ as the output. Autoregressive application of CRAN over a large number of iterations r can enable us to obtain $\tilde{\mathbf{U}} = [\tilde{\mathbf{U}}_{M,S_1}, \tilde{\mathbf{U}}_{M,S_2}, \ldots, \tilde{\mathbf{U}}_{M,S_r}], \tilde{\mathbf{U}} \in \mathbb{R}^{M\times rS}$. Thus, starting with S initial propagation steps, autoregressive application of the CRAN model can evolve the system to rS propagation steps into the horizon. The accuracy of the prediction $\tilde{\mathbf{U}}$ will indicate CRAN's data-driven learning capability.

For data-driven learning of TL as a propagation problem along the range, we can simply substitute $\mathbf{U}_{M,k}(\mathbf{x}_M, t_k; \phi)$ and $\mathbf{A}_{n,k}(\mathbf{x}_M, t_k; \phi)$ with $\mathbf{TL}_{M,k}(z_M, R_k; \phi)$ and $\mathbf{A}_{n,k}(z_M, R_k; \phi)$, respectively, without any loss of generality. Similarly, for autoregressive prediction of TL propagation along the range for cases not considered for training, we can replace $\mathbf{U}_{M,S_0}(\mathbf{x}_M, t_{S_0}; \phi)$ and $\mathbf{A}_{n,S_0}(\mathbf{x}_M, t_{S_0}; \phi)$ with $\mathbf{TL}_{M,S_0}(z_M, R_{S_0}; \phi)$ and $\mathbf{A}_{n,S_0}(z_M, R_{S_0}; \phi)$, respectively, without any loss of generality.

IV. TEST SCENARIO FOR CRAN-BASED LEARNING

A. Test problem

To demonstrate the learning of ocean acoustic TL with the CRAN, we consider the ocean domain with depth-

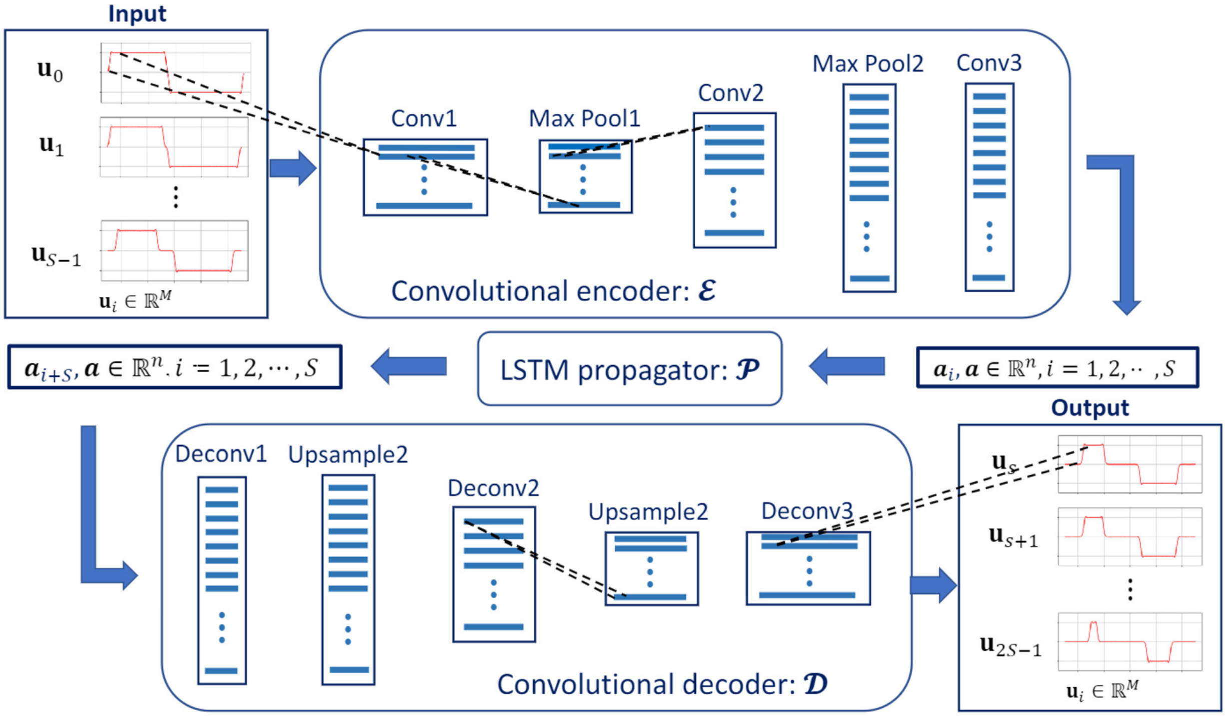

Input
$\mathbf{u}_0$
$\mathbf{u}_1$
$\mathbf{u}_{S-1}$
$\mathbf{u}_i \in \mathbb{R}^M$
Conv1
Max Pool1
Conv2
Max Pool2
Conv3
Convolutional encoder: $\mathcal{E}$
$\boldsymbol{a}_{i+S}, \boldsymbol{a} \in \mathbb{R}^n. i = 1, 2, \cdots, S$
LSTM propagator: $\mathcal{P}$
$\boldsymbol{a}_i, \boldsymbol{a} \in \mathbb{R}^n, i = 1, 2, \cdots, S$
Output
Deconv1
Upsample2
Deconv2
Upsample2
Deconv3
Convolutional decoder: $\mathcal{D}$
$\mathbf{u}_S$
$\mathbf{u}_{S+1}$
$\mathbf{u}_{2S-1}$
$\mathbf{u}_i \in \mathbb{R}^M$

FIG. 2. (Color online) Illustration of 1D CRAN model for data-driven learning.

dependent sound speed following Munk's sound speed profile (Jensen *et al.*, 2011) and representative ocean density profile (McDougall *et al.*, 2003). The computational ocean domain is shown in Fig. 3. For the case considered here, the domain has a range of 100 km and a depth of 5000 m, with a level bathymetry profile. Since all properties and boundary conditions remain uniform with the range, we consider this as a depth-dependent scenario. Here, we are interested in data-driven learning of the TL from depth-varying point sources of 50 Hz., under the prescribed domain and boundary conditions. We consider TLs of the acoustic signal primarily due to geometric spreading and reflection from the top and bottom surfaces. Losses due to volume attenuation are not considered in this study. The ocean bottom is considered as a smooth rigid boundary, and the ocean surface is considered as a fully reflective pressure release boundary (Jensen *et al.*, 2011). Thus, although there are reflections from the ocean surface and ocean bottom, losses due to scattering from the ocean bottom are negligible.

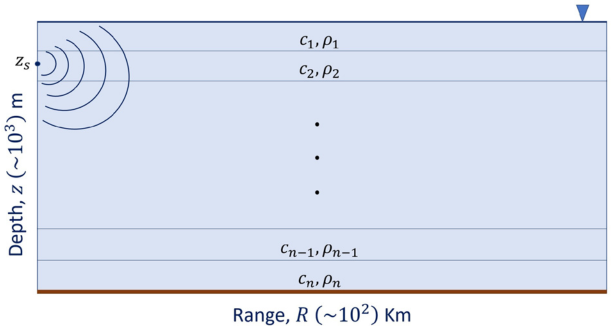


FIG. 3. (Color online) Illustration of ocean domain with depth-dependent properties.

B. Generation of ground truth for far-field TL

For the physical conditions assumed in this study, URN transmission can be considered a linear wave propagation phenomenon. It can be thus modeled in the frequency domain via the *Helmholtz equation* in Cartesian coordinates $\mathbf{x}$,

$$\left[\nabla^2 + k^2(\mathbf{x})\right] p(\mathbf{x}, \omega) = q(\mathbf{x}, \omega; \mathbf{x_0}), \quad \mathbf{x} \in \Omega, \tag{9}$$

subject to the boundary conditions. Here, p represents the far-field acoustic pressure propagated from the source q, situated at a source location $\mathbf{x_0}$. For a two-dimensional ocean domain Ω, we can assume $\mathbf{x} = (R, z)$ and $\nabla^2 = (\partial^2/\partial R^2) + (\partial^2/\partial z^2)$, without any loss of generality. $k(\mathbf{x})$ is the wavenumber of the medium at a radial frequency of ω. Here, R can be considered as the direction along the range of the ocean domain from some reference point, and z is the depth from the ocean surface. Equation (9) can be solved approximately via ray tracing for a wide range of both depth and range-dependent variations in sound source location and ocean environment parameters (e.g., sound speed, bathymetry, etc.). Although ray tracing techniques ignore lower frequency solutions of Helmholtz equation (Jensen *et al.*, 2011), well-known computational techniques for ray tracing (Porter and Bucker, 1987; Porter and Liu, 1994) have shown that reasonably accurate results can be obtained even at frequencies of 50 Hz (Porter, 2010). Ray tracing methods can also show anomalies in the form of sharp caustic and shadow zones in the ray amplitude computation. However, the application of geometric beams in well-known ray/beam tracing solvers can limit the occurrence of caustics to acceptable levels as reported in Porter and Liu (1994). Although ray tracing techniques lack the high physical resolution of the higher-fidelity finite-element method, they are employed here for generating training data over a range of parameters at a reasonable computational cost.

Here, we compute the far-field acoustic pressure levels in the presence of geometric spreading and ocean surface and bottom reflections via BELLHOP, a ray/beam tracing software (Porter, 2010). For a given point source depth (z_s), BELLHOP is employed to compute the acoustic pressure at uniformly distributed receiver locations along the depth (z) throughout the prescribed ocean domain range at uniformly distributed range locations. Such acoustic pressure amplitudes are utilized to obtain their TL. We extract a snapshot of the TL distribution on a uniformly discretized $z-R$ plane for the ocean domain shown in Fig. 3. Such TL distribution is computed for several point source locations z_s distributed throughout the depth. We aim to develop a generalized learning of far-field TL for depth-varying point sources via the CRAN model. This would be achieved by training the model for a few source locations and demonstrating the TL predictions for any general source location not included in the training set.

It is important to note that although BELLHOP actually computes acoustic pressure, we train the CRAN on the TL data obtained by post-processing the pressure signal intensity [according to Eq. (1)]. The reason for such a choice is that acoustic pressure varies over several orders of magnitude in such a large domain, and DL networks cannot be trained properly on such data. On the other hand, the TL of acoustic pressure is both easy to interpret and more suitable for training the network.

V. RESULTS

In this section, we will present the results for learning far-field URN TL in a depth-dependent ocean environment with the CRAN. The training of the CRAN model and its prediction capability for the test cases will be discussed. Various components of the CRAN architecture were trained with TensorFlow 2.5.0 (Abadi, 2015) libraries.

A. Training of the CRAN architecture

As stated earlier, the TL of acoustic pressure from the depth-varying point sources was used for training the DL models. The acoustic pressure computed via BELLHOP is not directly employed for training the DL models as it varies over orders of magnitude in the two-dimensional domain. Thus, the TL of the acoustic pressure is a more suitable choice of training data for the DL network.

The TL distribution computed over the domain via BELLHOP for a source at a depth $z_s = 950$ m is shown in Fig. 4. The TL distribution was obtained by applying 2500 Gaussian beams over an angular fan spanning from $-45°$ to $45°$. The number of Gaussian beams was selected here to obtain converged ray-tracing solutions. BELLHOP and most other routine ray/beam tracing solvers lead to spurious results when the pressure level decreases many orders of magnitude (Jensen *et al.*, 2011; Porter, 2010). Such a rapid drop in pressure levels usually occurs at the boundary of the shadow zones where no ray traces enter. We observed spurious results as the pressure levels dropped beyond 10 orders of magnitude. Thus, pressure levels beyond 10 orders of magnitude were filtered away, leading to a flat region in the TL contour beyond 200 dB in Fig. 4. For obtaining the data sets, we considered Munk's sound speed profile sampled with 26 depth-wise velocities. Therefore, the ocean depth was divided into 25 depth-wise divisions with uniform fluid properties during BELLHOP's numerical ray trajectory computations.

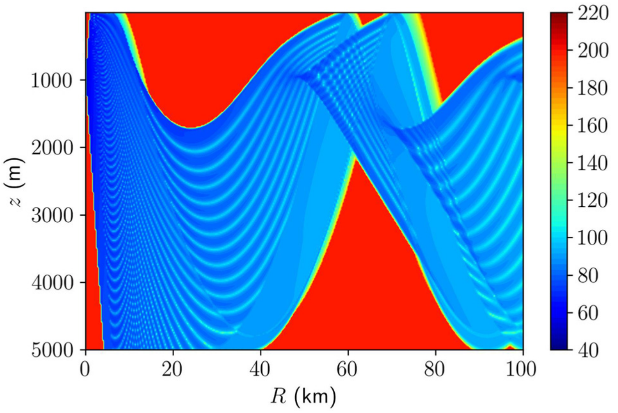


FIG. 4. (Color online) Transmission loss distribution obtained via BELLHOP for a source depth of 950 m.

Since we consider a depth-dependent ocean environment with depth-varying sound speed and source locations, a sufficiently high resolution of the data along the depth is preferred. Thus, the ray-tracing solutions were observed at 2049 uniform depth-wise locations. This led to an extremely accurate resolution of beam tracing solution along the depth. We expect that our data-driven autoencoders will still be able to learn a sufficiently low-dimensional representation along the depth even for a very high depth-wise resolution. Similarly, a range-wise discretization was also performed, but a lower resolution was considered for the range. To select the optimal resolution that will lead to sufficiently accurate sampling, a convergence study was performed for various range-wise discretizations. 176, 352, 704, and 1408 uniform range-wise stations were considered, and the L_1 and L_2 error for sampling on all the coarser discretizations compared to 1408 range-wise stations were computed. The results showed that 352 range-wise sampling points are sufficiently accurate with an L_1 error of 1% and an L_2 error of 2.5%.

To train the CRAN with TL distribution for various depth-varying source locations, BELLHOP was employed to obtain data for 21 source depths. Data were also obtained for three validation depths and 12 test depths. These 36 source depth locations were randomly sampled from the domain and then corrected so that their location coincided with the nearest depth-wise sampling point. The CRAN training was separated into a convolutional autoencoder training on high-dimensional physical data and an LSTM training on the reduced-dimensional latent states obtained from the trained convolutional autoencoder's encoder.

First, we discuss the convolutional autoencoder training. The training was performed via the ADAM optimizer (Kingma and Ba, 2014) with the standard objective of minimizing the mean square error between the target TL results and those obtained via the convolutional encoding and decoding operations. A hyperparameter tuning was performed for optimal convolutional autoencoder training and generalization. Several network parameters like CNN filter size, number of filters, etc., were considered for optimal tuning of the network. However, the most important hyperparameters considered here are the dimension of the latent states, n, and the number of training epochs. Since the true and predicted solutions considered here can be represented in a pixelated format on a Cartesian grid, the accuracy of the predicted solutions was measured via the structural similarity index measure (SSIM). SSIM is a statistical measure used for comparing two images or two same-dimensional data sets with pixelated representation. Readers can explore Wang *et al.* (2004) for further details on SSIM. A SSIM of 1.0 between two images indicates that the two images are identical, whereas 0 implies no similarity.

The convolutional autoencoder was trained for 320 epochs with $n = 16$, 32, 64, and 128, resulting in SSIM $= 0.95$, 0.973, 0.99, and 0.995, respectively, between the training ground truth and the decoder outputs. The SSIM values presented here are mean SSIM values over all the 21 training sets combined. We select $n = 64$ as the optimal dimension of the latent states and SSIM $= 0.99$ as sufficient accuracy for training sets. The dimension of the latent states will determine the capacity of the CNN autoencoder to obtain a low-dimensional representation. It is sufficient to tune this hyperparameter based on its effect on the training predictions only. However, the training epoch can affect both the training and validation, as training beyond optimal duration can cause an overfitting of the network, leading to a decay in the validation accuracy. This can be observed in Fig. 5, showing the mean training SSIM for all the 21 training sets combined and also the mean SSIM for the three validation sets, with varying training epochs. We can observe that mild effects of overfitting are observed beyond 320 epochs as the validation accuracy decreases after this point. Thus, we consider the convolutional autoencoder to be optimally trained with 320 epochs as further training would adversely affect the generalized prediction capacity of the network.

Next, we discuss the SS-LSTM training. Since we plan to train the SS-LSTM on the reduced-dimensional latent states obtained from the trained convolutional encoder, there are no specific target latent data sets. We instead inspect the accuracy of the SS-LSTM training and validation sets on the reconstructed CRAN predictions and the target, for both training and validation data sets. While the convolutional autoencoder network remains fixed, any change in the errors will solely be due to the SS-LSTM network.

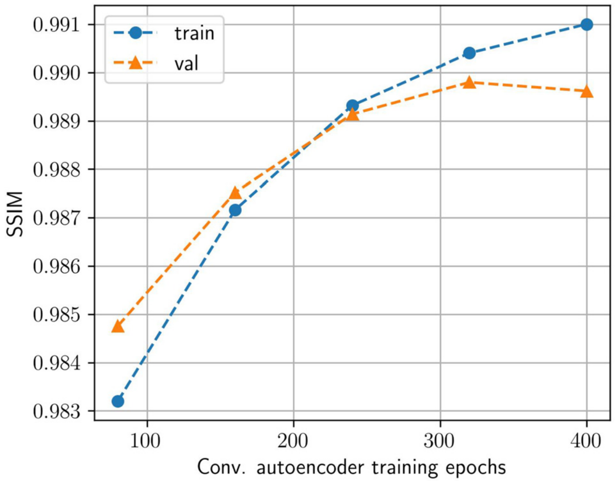
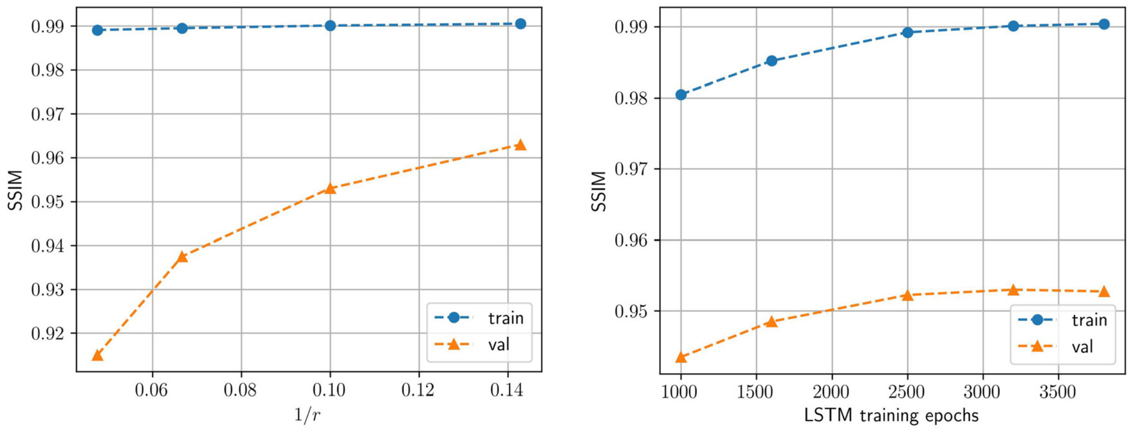


FIG. 5. (Color online) Variation of SSIM on training epoch the network training and validation. SSIM is computed between convolutional autoencoder predictions and ground truth.

During the LSTM training, the total propagated distance of the ray traces along the range, $\bar{R}$, is first divided into $r+1$ sequences, where $R = \bar{R}/(r+1)$ is the span of each sequence. The LSTM training is performed over r sequences for all the training source depths, with sequences 1 to r being the input data sets and sequences 2 to $r+1$ being their corresponding output sets. During the LSTM validation stage, we use the CRAN in an autoregressive prediction phase initiated with the first sequence for each validation source depth. Thus, r would represent the maximum number of sequences into which we can possibly divide the complete domain data. Here, r is another hyperparameter specifically required for LSTM validation to ensure the CRAN could be employed for autoregressive prediction of source depths not considered in the training data set.

Figure 6(a) shows how the validation and training prediction SSIM varies with r for the SS-LSTM trained for 3200 epochs. The results show that r does not have much influence on the training accuracy but significantly influences the validation accuracy. We do not expect the sequence number r to affect training the accuracy as we will be training the SS-LSTM on all the sequences. However, during the prediction phase (validation and testing), the number of sequences r would determine the initiation sequence size, which would eventually affect the autoregressive prediction accuracy. Thus, r indicates the near-field dependency of the CRAN for an accurate far-field prediction, with a higher r indicating lower near-field dependency. From Fig. 6(a), we consider $r = 10$ to be an optimally tuned hyperparameter, with a cut-off tolerance of 5% in SSIM accuracy. While further improvement beyond this value is possible, it comes at the cost of a huge initiation data requirement. The larger initiation sequence size required for accurate autoregressive far-field propagation can be attributed to the significantly varying near-field TL distribution pattern with source depth.



(a) Hyperparamter: $1/r$

(b) Hyperparamter: LSTM training epochs

FIG. 6. (Color online) Dependence of SSIM on LSTM hyperparameter tuning: (a) $1/r$ and (b) training epochs. Here, SSIM measures the difference between the CRAN predictions and ground truth during training and validation.

The CRAN is therefore required to observe the near-field TL evolution longer before it could properly predict the far-field TL distribution for source depths different from the training source depths. Thus, r also indicates the data efficiency of the CRAN during prediction phase. A high r indicates that for cases outside the training range one would need to provide very little near-field TL data to the CRAN network to obtain accurate far-field TL prediction.

Figure 6(b) shows the validation and training SSIM of the CRAN predictions for both the training and validation sets with $r = 10$. Similar to the convolutional autoencoder training, we observe mild effects of overfitting during the SS-LSTM training. The validation set indicates optimal training epochs to be 3200 as the validation error begins to increase slowly beyond this stage. It is often difficult to predict *a priori* how many training cases are required to ensure sufficient generalization capacity for DL models. Here, by inspecting the validation SSIM in conjunction with the training SSIM and using early stopping, we have shown that CRAN generalization can be improved even when training data are limited.

The trained CRAN parameters for $r = 10$ and $n = 64$ are presented in Fig. 7. Starting with an input dimension of (32, 2049, 1), the dimension of the feature space from each layer of the various networks is shown in round brackets. The layer parameters are shown in square brackets. The encoder

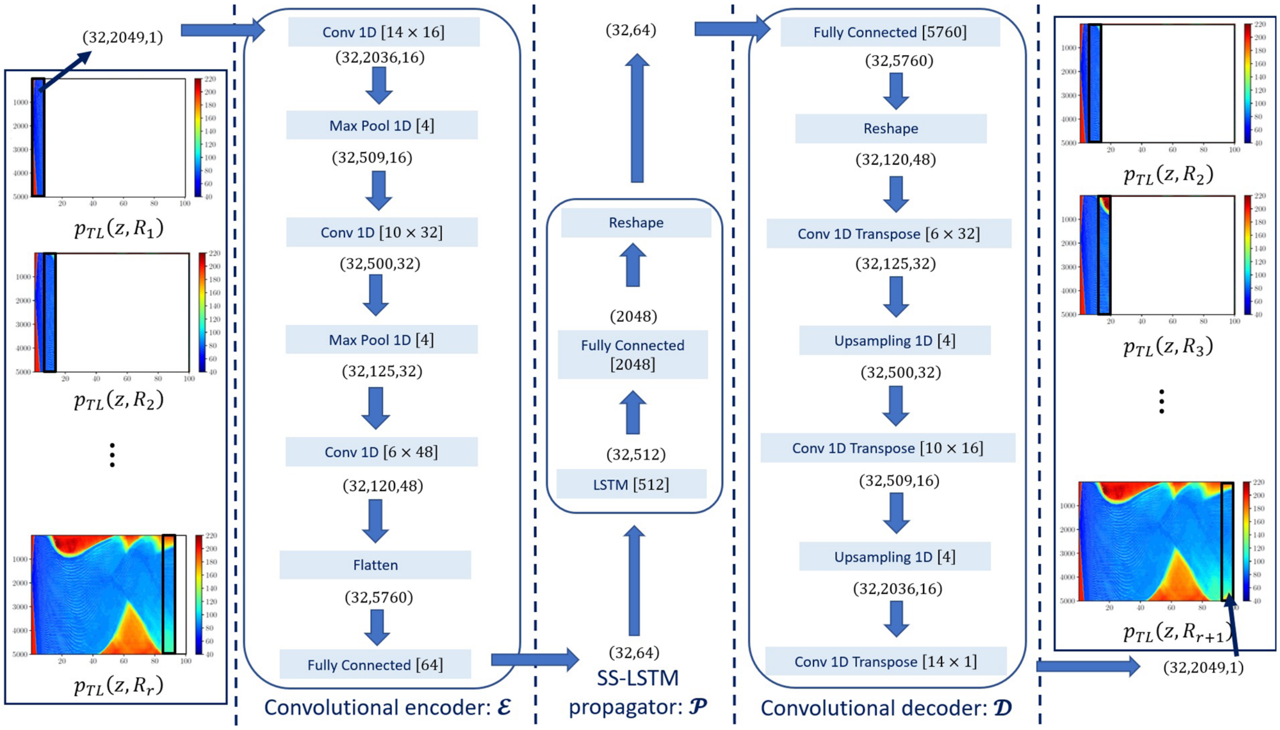


FIG. 7. (Color online) Trained CRAN parameters for SS-LSTM sequence length of 32 snapshots and convolutional autoencoder latent state dimension of 64.

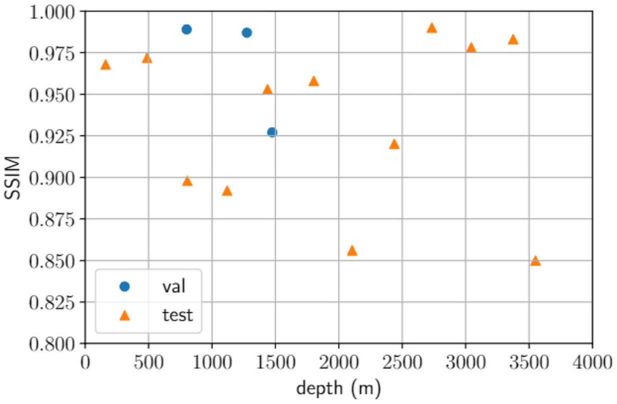


FIG. 8. (Color online) Structural similarity index measure of predicted transmission loss for all validation and test source depths.

has three 1D convolutional layers of filter sizes 14, 10, and 6 and two 1D max-pooling layers, each having a filter size of 4. The number of filters increases by 16 for each convolutional 1D layer of the encoder. The decoder has three 1D transpose convolution and two 1D upsampling layers, which perform just the reverse operations of the 1D convolutional and 1D max-pooling layers. Here, we use 1D convolution and pooling operations since we are only reducing the spatial dimension along the ocean depth. Thus, instead of using the whole depth discretization dimension of 2049, we are propagating along the range with a reduced dimension of 64 via the SS-LSTM. Figure 7 also shows how the autoregressive CRAN predictions are performed to obtain the TL distribution along the range, starting from the near-field TL data for $z_s = 1802$ m.

It is important to note here that we intentionally do not consider an end-to-end learning mechanism by directly training the whole CRAN architecture with the high-dimensional input and output TL data. Such a decision was taken to improve the flexibility of the network and the interpretability of the learning mechanism. Improved flexibility enabled us to assess the performance of individual components and only tune the necessary hyperparameters for improved performance. This was evident as we could identify the CRAN validation error was arising due to the SS-LSTM and not the CNN autoencoder. Furthermore, we could observe how the near-field TL data for validation cases affected the far-field TL propagation.

B. Far-field TL prediction for test cases

The CRAN network with the trained convolutional autoencoder and the SS-LSTM was next employed on the test source depths to predict the far-field propagation of the TL along the range. Figure 8 shows the SSIM for the various validation and test cases along the domain. It can be observed that the test cases cover most of the domain and thus can properly indicate the CRAN's generalization capacity. The CRAN predictions show a SSIM of 85% considering all the cases, and a prediction SSIM rises close to 90% or above for most of the validation and test cases.

To further investigate the CRAN predictions, we compare the fully propagated predictions and the target solutions in Figs. 9–11, for source depths $z_s = 161$ m, $z_s = 1802$ m, and $z_s = 3550$ m, respectively. Thus, they represent three completely different TL distribution patterns. The predicted solutions are obtained by initiating the CRAN with the first set of snapshots from the target solution and completing the rest of the prediction autoregressively (see Fig. 7). We can see an excellent similarity between the predicted and target solutions for $z_s = 161$ m (Fig. 9), with only minor differences in magnitude near the upper limit of the TL values. This is also corroborated by a SSIM of 0.97. For $z_s = 1802$ m (Fig. 10), the TL distribution pattern of the predictions closely matches the target solutions. However, some differences in the magnitude are observed far into the propagated range at both high and low depths. However, overall, the prediction shows high accuracy with a SSIM of 0.953.

We further compare the TL predictions along the range and the ground truth for $z_s = 1802$ m at fixed receiver depths. These receivers are placed at depths of 1200 and 3200 m, as shown by dotted lines in Fig. 10. The TL comparisons are shown in Figs. 12(a) and 12(b) for receiver depths of 1200 and 3200 m, respectively. As observed earlier, we see a close match between the CRAN predictions and ground truth throughout the domain for both the cases, except for the shadow zone observed from 60 to 70 km for a

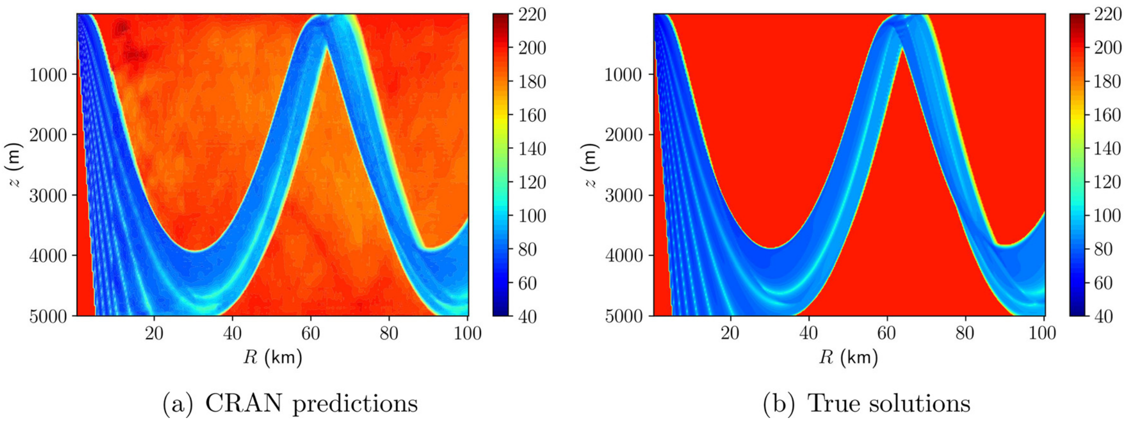
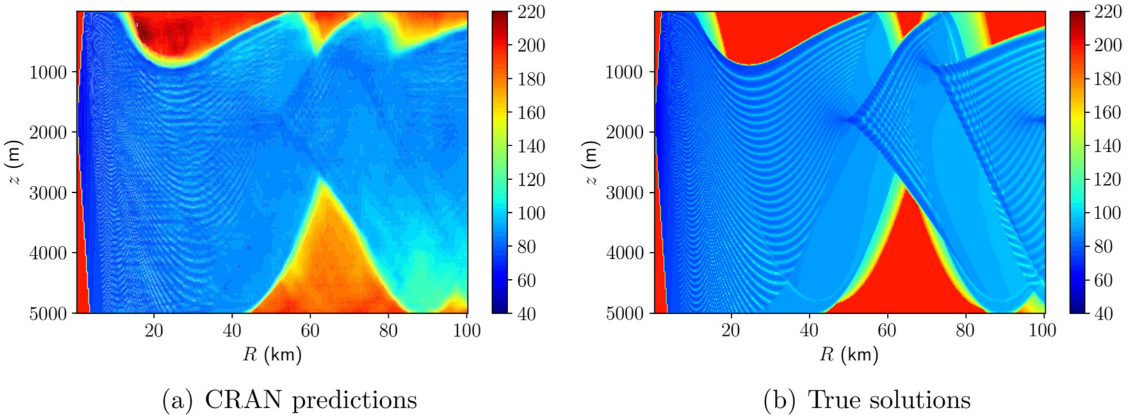


FIG. 9. (Color online) Comparison of (a) CRAN predictions and (b) true solutions of transmission loss distribution for source depth of 161 m.



(a) CRAN predictions

(b) True solutions

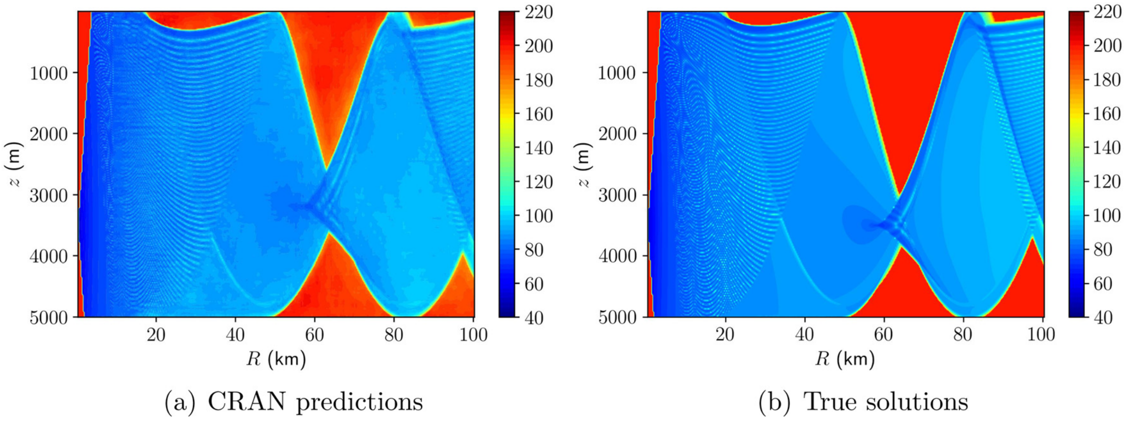
FIG. 10. (Color online) Comparison of (a) CRAN predictions and (b) true solutions of transmission loss distribution for source depth of 1802 m.

receiver depth of 3200 m. The relative difference between the predicted and true TL usually lies within 10%, except at the shadow zones far from the source locations. For the shadow zones, the CRAN predictions do not exactly match the TL ceiling of 200 dB set in the ground truth for shadow zones or regions with very weak relative signal intensity. However, the locations of the shadow zones are detected accurately by the CRAN predictions.

The CRAN predictions at $z_s = 3550$ m (Fig. 11) show the lowest SSIM of 0.85. The nearest training source to this test source was at $z_s = 3250$ m. The 300 m difference in source depth between test and training for this case is the farthest among all the test cases and is the primary reason for a somewhat lower prediction SSIM. Notably, the outer structure of the predicted TL distribution for $z_s = 3550$ m matches the target solution closely. However, some differences in the magnitude and also the TL patterns can be observed throughout the domain, especially for the lower TL values. These differences increase as the predicted TL is propagated farther away from the source. These differences in both magnitude and structure of the TL patterns lead to a somewhat lower value of SSIM for this case.

Overall, generalized prediction capabilities of far-field TL propagation via CRAN were demonstrated for any general source depth over the domain. It must be noted that the data obtained from BELLHOP show a sharp change in TL patterns traveling from the ensonified zone to the shadow zone. As shown in Fig. 10, for certain test source depths that were not considered for training, the CRAN may not be able to fully capture the sharp change in TL patterns at certain regions of the domain. However, overall, the CRAN predictions closely matched the TL distribution patterns of the target solutions in most cases. This is demonstrated by the mean SSIM accuracy of 94% over all the 15 source depths that were not considered for training (Fig. 8). Even for the source depth almost 300 m away from a training source depth over a domain depth of 5000 m, we could predict the solution with 85% overall accuracy.

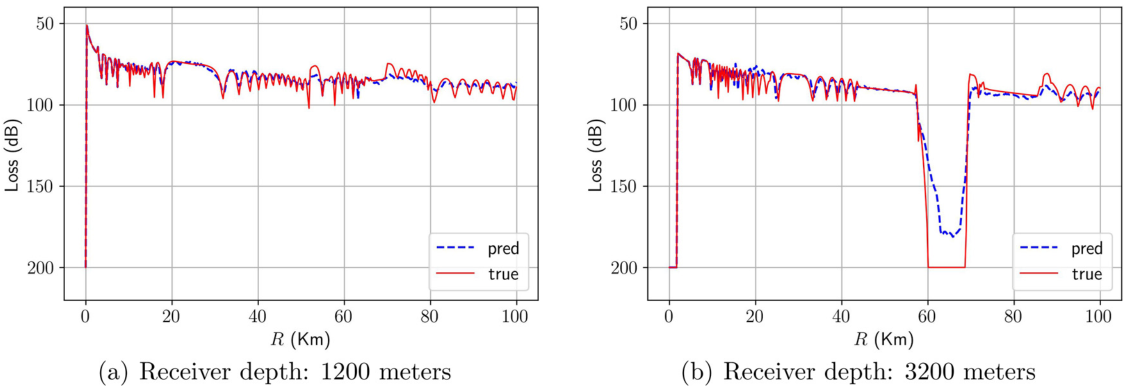
The TL distribution over the whole domain was obtained within 4–6 central processing unit (CPU) seconds with the CRAN model for each of the source depths. On the other hand, similar predictions for each source depth via BELLHOP required 400–450 CPU seconds. This demonstrates the potential application of the CRAN model for real-time decision-making and control. Here, we have considered the data-driven CRAN model for TL prediction with depth-varying sound sources. Future studies will include a wider range of parameters (e.g., varying ocean medium or ocean bottom properties) to demonstrate the scalability of the CRAN model in a more complicated and realistic underwater ocean environment.



(a) CRAN predictions

(b) True solutions

FIG. 11. (Color online) Comparison of (a) CRAN predictions and (b) true solutions of transmission loss distribution for source depth of 3550 m.



(a) Receiver depth: 1200 meters

(b) Receiver depth: 3200 meters

FIG. 12. (Color online) Comparison of CRAN predictions and true transmission loss along range for source depth of 1802 m: (a) receiver depth of 1200 m; (b) receiver depth of 3200 m.

VI. CONCLUSIONS

In this article, the authors have presented the CRAN for data-driven learning of far-field acoustic transmission from depth-dependent point sources in the underwater ocean environment. The CRAN model consisted of a convolutional autoencoder for learning a low-dimensional representation of high-dimensional physical data and a LSTM propagator for learning the TL propagation in low dimensions.

When trained with full-domain snapshots of TL distribution for a few distributed source depths, the CRAN model showed generalized prediction capacity of far-field TL for other source depths outside the training set. Furthermore, such predictions were obtained in a physically consistent manner. Specifically, the CRAN model trained with TL distribution obtained via ray/beam tracing solver BELLHOP for 21 sources distributed along the ocean depth could predict the transmission loss distributions for another 15 source depths outside the training set with a mean SSIM accuracy of 94%. Even for the source depth almost 300 m away from the closest training source depth over a domain depth spanning 5000 m, the CRAN could predict the solution with 85% overall accuracy. It was observed that the CRAN predictions were within a relative error bound of 10% to the true solutions except for the sharp changes in transmission loss at the shadow zones. However, even though the relative error of CRAN predictions was larger at the shadow zones, it could still detect the existence of shadow zones accurately. This indicated the physical consistency of the CRAN predictions. Overall, the CRAN was able to achieve generalized learning of the underlying physics of underwater ocean acoustic transmission phenomena. It was also observed that for a limited number of training cases (21 training source depths), the generalization capacity of the CRAN could be improved via careful inspection of validation errors, adequate hyperparameter tuning, and early stopping of CRAN training.

Here, we have considered medium-fidelity ray-tracing solvers for obtaining the ocean acoustic transmission loss ground truth for CRAN training and testing. The CRAN is a data-driven model agnostic to how the data are acquired. Thus, the CRAN model is equally applicable if the present data are augmented with data obtained from higher-fidelity solvers or experimental measurements. This shows the potential of data-driven DL models like CRAN to obtain a digital twin of ocean acoustic transmission over a wide range of parameters and physical phenomena of varying complexity. The real-time online prediction of such a digital twin can be applied for fast and physics-informed decision-making in marine vessel operations.

ACKNOWLEDGMENTS

This research was supported by Natural Sciences and Engineering Research Council of Canada (NSERC) Grant No. IRCPJ 550069-19 and Seaspan Shipyard. We would also like to acknowledge use of the graphics processing unit (GPU) facilities at the Digital Research Alliance of Canada clusters for the training of our DL models.

Abadi, M. (**2015**). “TensorFlow: Large-scale machine learning on heterogeneous systems” https://www.tensorflow.org/ (Last viewed April 30, 2022).

Borrel-Jensen, N., Engsig-Karup, A. P., and Jeong, C.-H. (**2021**). “Physics-informed neural networks for one-dimensional sound field predictions with parameterized sources and impedance boundaries,” JASA Express Lett. **1**(12), 122402.

Bronstein, M. M., Bruna, J., LeCun, Y., Szlam, A., and Vandergheynst, P. (**2017**). “Geometric deep learning: Going beyond Euclidean data,” IEEE Signal Process. Mag. **34**(4), 18–42.

Bukka, S. R., Gupta, R., Magee, A. R., and Jaiman, R. K. (**2021**). “Assessment of unsteady flow predictions using hybrid deep learning based reduced-order models,” Phys. Fluids **33**(1), 013601.

Chi, J., Li, X., Wang, H., Gao, D., and Gerstoft, P. (**2019**). “Sound source ranging using a feed-forward neural network trained with fitting-based early stopping,” J. Acoust. Soc. Am. **146**(3), EL258–EL264.

Deo, I. K., and Jaiman, R. K. (**2022**). “Predicting waves in fluids with deep neural network,” Phys. Fluids **34**, 067108.

Duarte, C. M., Chapuis, L., Collin, S. P., Costa, D. P., Devassy, R. P., Eguiluz, V. M., Erbe, C., Gordon, T. A., Halpern, B. S., Harding, H. R., Havlik, M. N., Meekan, M., Merchant, N. D., Miksis-Olds, J. L., Parsons, M., Predragovic, M., Radford, A. N., Radford, C. A., Simpson, S. D., Slabbekoorn, H., Staaterman, E., Van Opzeeland, I. C., Winderen, J., Zhang, X., and Juanes, F. (**2021**). “The soundscape of the Anthropocene ocean,” Science **371**(6529), eaba4658.

Erbe, C., Marley, S. A., Schoeman, R. P., Smith, J. N., Trigg, L. E., and Embling, C. B. (**2019**). "The effects of ship noise on marine mammals—A review," Front. Mar. Sci. **6**, 606.

Ferguson, E. L. (**2021**). "Multitask convolutional neural network for acoustic localization of a transiting broadband source using a hydrophone array," J. Acoust. Soc. Am. **150**(1), 248–256.

Fotiadis, S., Pignatelli, E., Valencia, M. L., Cantwell, C., Storkey, A., and Bharath, A. A. (**2020**). "Comparing recurrent and convolutional neural networks for predicting wave propagation," arXiv:2002.08981.

Hornik, K. (**1991**). "Approximation capabilities of multilayer feedforward networks," Neural Netw. **4**(2), 251–257.

Hsieh, W. W. (**2009**). *Machine Learning Methods in the Environmental Sciences: Neural Networks and Kernels* (Cambridge University, Cambridge, UK).

Huang, Z., Xu, J., Gong, Z., Wang, H., and Yan, Y. (**2018**). "Source localization using deep neural networks in a shallow water environment," J. Acoust. Soc. Am. **143**(5), 2922–2932.

James, K. R., and Dowling, D. R. (**2005**). "A probability density function method for acoustic field uncertainty analysis," J. Acoust. Soc. Am. **118**(5), 2802–2810.

James, K. R., and Dowling, D. R. (**2011**). "Pekeris waveguide comparisons of methods for predicting acoustic field amplitude uncertainty caused by a spatially uniform environmental uncertainty (L)," J. Acoust. Soc. Am. **129**(2), 589–592.

Jensen, F. B., Kuperman, W. A., Porter, M. B., Schmidt, H., and Tolstoy, A. (**2011**). *Computational Ocean Acoustics* (Springer, New York).

Kingma, D. P., and Ba, J. (**2014**). "Adam: A method for stochastic optimization," arXiv:1412.6980.

Lee, K., and Carlberg, K. T. (**2020**). "Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders," J. Comput. Phys. **404**, 108973.

Leshno, M., Lin, V. Y., Pinkus, A., and Schocken, S. (**1993**). "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," Neural Netw. **6**(6), 861–867.

Mallik, W., Jaiman, R. K., and Jelovica, J. (**2021**). "Kinematically consistent recurrent neural networks for learning inverse problems in wave propagation," arXiv:2110.03903.

Mallik, W., Jaiman, R. K., and Jelovica, J. (**2022**). "Convolutional recurrent autoencoder network for learning underwater ocean acoustics," arXiv:2204.05573.

Mansour, T. (**2019**). "Deep neural networks are lazy: On the inductive bias of deep learning," Master's thesis, MIT, Cambridge, MA.

McDougall, T. J., Jackett, D. R., Wright, D. G., and Feistel, R. (**2003**). "Accurate and computationally efficient algorithms for potential temperature and density of seawater," J. Atmos. Ocean. Technol. **20**(5), 730–741.

Moseley, B., Nissen-Meyer, T., and Markham, A. (**2020**). "Deep learning for fast simulation of seismic waves in complex media," Solid Earth **11**(4), 1527–1549.

Niu, H., Ozanich, E., and Gerstoft, P. (**2017**). "Ship localization in Santa Barbara Channel using machine learning classifiers," J. Acoust. Soc. Am. **142**(5), EL455–EL460.

Porter, M. B. (**2010**). "Bellhop: A beam/ray trace code," https://oalib-acoustics.org/website_resources/AcousticsToolbox/Bellhop-2010-1.pdf (Last viewed June 20, 2022).

Porter, M. B., and Bucker, H. P. (**1987**). "Gaussian beam tracing for computing ocean acoustic fields," J. Acoust. Soc. Am. **82**(4), 1349–1359.

Porter, M. B., and Liu, Y.-C. (**1994**). "Finite-element ray tracing," Theor. Comput. Acoust. **2**, 947–956.

Smith, J. D., Azizzadenesheli, K., and Ross, Z. E. (**2021**). "Eikonet: Solving the eikonal equation with deep neural networks," IEEE Trans. Geosci. Remote Sens. **59**(12), 10685–10696.

Sorteberg, W. E., Garasto, S., Cantwell, C. C., and Bharath, A. A. (**2019**). "Approximating the solution of surface wave propagation using deep neural networks," in *INNS Big Data and Deep Learning Conference* (Springer, New York), pp. 246–256.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (**2004**). "Image quality assessment: From error visibility to structural similarity," IEEE Trans. Image Process. **13**(4), 600–612.