

Analysis of an Ultra-Short True Time Delay Line Optical Reservoir Computer

Nathan McDonald, Alex Hollingsworth, Steven Beninati, Chjan Lim, Wei Ji, Chengyu Shi, Michael L. Aigiorgousis, Shengbai Zhang, Zhaoran Rena Huang

Abstract— This work presents the experimental demonstration of a true-time delay line optical reservoir computer (ORC) using an incoherent light chaos oscillator. In contrast to other benchtop ORC systems, no external fiber spool was employed, enabling a characteristic delay of 28.4 ns, one of the fastest reported optical reservoirs to date. Comparable error metrics were obtained for standard benchmark tasks despite the reduced time scale. Practical experimental techniques, namely preamble functions and fading memory capacity measures, are introduced in this work. A mathematical model of optimized virtual nodes for the best performance of the RC was established. The fast ORC was also applied for two real world applications: respiratory motion prediction used in radiotherapy and perovskite compound property prediction used in photovoltaic material discovery. The respiratory motion prediction was compared with long short-term memory (LSTM) machine learning algorithms, the former attaining compatible results with orders of magnitude faster training speed. The ORC results for the perovskite compound classification task were compared with random forest approach, where the former demonstrated slightly better computation predication but again with much faster computing speed.

Index Terms— optical reservoir computing, echo state network, artificial neural networks, optical neuromorphic computing, analog optical computing, recurrent neural networks.

I. INTRODUCTION

Reservoir computing (RC) is a type of recurrent neural network (RNN) that has attracted much attention lately in the artificial intelligence (AI) and machine learning communities. Like other artificial neural networks (ANN), RC networks have an input, hidden, and output layer of neurons or nodes. Notably, the RC hidden layer is a “reservoir” of neurons recurrently, sparsely, statically, and typically randomly connected [1, 2]. Only the weights connecting the hidden layer to output layer neurons are updated during network training. This simplicity in network topology and training gives rise to huge size, weight, and power (SWaP) efficiencies, particularly important for fieldable hardware [3].

A true time delay line optical RC is efficient for processing temporally correlated signals at very high speeds [4]. A fiber spool often serves as the delay line, storing “virtual nodes.” A delay line of 1-2 km (affording a characteristic delay of ~ 5 μ s) is typical [5 - 8]. Such length (and time scales) are impractical given the goal of RC chip-scale integration. Reducing the delay line length often results in reservoir size reduction unless combined with an increased signal injection rate at the input layer, though this in turn may increase system noise or power

dissipation. The term “true time” here denotes delay is due to the system length itself and not a function of resonance structures such as a ring resonator that produces time delay by “trapping” light in a resonator for a certain amount of time.

Recently a coherent light photonic RC using differential phase shift keying (DPSK) modulation scheme was demonstrated with a characteristic delay of 63.3 ns, affording $N = 1000$ virtual nodes [9]. While DPSK or phase modulation based chaos oscillators reduce noise [10], they are more complex and costly to implement than the architecturally straightforward and inexpensive incoherent light photonic RC used in this work. Here, we focus on an incoherent ORC with time delay of 28.4 ns with a maximum of $N = 564$ nodes.

Parameter selection for ANNs tends to be more art than science. The required number of nodes for an expected level of performance is among them. In RC, node counts range from 32 - 1000 [9, 11, 12]. The general theory is that computation accuracy increases asymptotically with the number of virtual nodes. However, experimentally we observed that an optimal N exists for the least root mean square error (RMSE) of many tasks. In this research, we will report a mathematic model that analyzes the computation capacity of a RC based on information theory.

Besides several standard benchmark tasks [5, 9], viz. sine/square wave classification, nonlinear channel equalization (NCE), and speech recognition, we also introduce two real-world tasks: respiratory motion prediction and perovskite compound property prediction. Respiratory motion prediction is used in radiotherapy where the position of the lung or liver tumors need to be tracked so the imaging system and radiation apparatus can be adjusted in real time. Long short term memory (LSMT) ANNs [13], multi-layer perceptron neural networks (MLP-NN) [14], adaptive boosting and multi-layer perceptron neural networks (ADMLP-NN) [15] have demonstrated good prediction results; however, unlike optical RC, they require significant amounts of training data and time, impractical for real-time training and prediction [16].

Lastly, machine learning has recently expanded into the field of material science to predict relevant properties and accelerate materials discovery in photovoltaics, Li-ion batteries, and ceramics research [17–19]. Band gap prediction in particular has been attempted across a wide range of materials using different methods such as ANNs, support vector machines, and kernel ridge regression [17, 20–25]. In this work, we will apply optical RC to bandgap prediction of perovskite materials.

II. BACKGROUND

A. Echo State Networks (ESN)

Mathematically, the goal of RC is to learn the desired expected value of an l -dimensional output $y(k)$ as a function of n -dimensional input $u(k)$, given a signal(s) of length $k = 1, 2, \dots, K$. This is achieved by first projecting the input into a typically larger N -dimensional space $x(k)$ called a reservoir. For a given discrete time input $u(k)$, the reservoir state is

$$x(k) = f(\beta W_{in} u(k) + \alpha W_{res} x(k-1)), \quad (1)$$

where f is the nonlinear transfer function of the reservoir, β is the input gain, α is the attenuation of the reservoir state, W_{in} is an $n \times N$ matrix of random projection weights between the input layer neurons and the reservoir, and W_{res} is a $N \times N$ matrix describing the internal connections and individual weights of the reservoir neurons. Both α and β are on the interval $[0, 1]$.

The RC response $\hat{y}(k)$ is the linear sum of the weighted reservoir output neurons,

$$\hat{y}(k) = W_{out} S(k), \quad (2)$$

where $S(k)$ is an $(N+1) \times K$ matrix concatenating the reservoir states $x(k)$ with the addition of a fixed output bias neuron of value 1 and W_{out} is an $l \times (N+1)$ matrix of trained output weights. Knowing the target output $y(k)$ for the training examples, W_{out} may be calculated directly by minimizing the Mean Squared Error (MSE) between the target output $y(k)$ and the reservoir output via ridge regression,

$$W_{out} = y S^T (S S^T + \lambda I)^{-1}, \quad (3)$$

where T denotes the transpose, λ is the regularization factor, and I is the identity matrix. The most computationally expensive calculation then is the initial matrix inverse calculation as opposed to the iterative gradient descent methods typically employed in training ANNs.

B. Time Delay Reservoir (TDR)

A typical ESN distributes the input data in space across multiple nonlinear neurons like any other traditional feed forward ANN, but the ESN can be further simplified to a ring topology with only one nonlinear neuron and an arbitrary number N “virtual nodes” without an appreciable loss of computational power for certain classes of problems. That is, this streamlined configuration, known as a time delay reservoir among other similar names, instead spreads the information in time across multiple “virtual nodes” in the delay line [4]. When using a TDR, the continuous time input signal $u(t)$ is sampled, $u(k)$, and held for time τ , the duration of one round-trip through the delay line reservoir. Per input $u(k)$, the delay line of characteristic duration τ is sampled $N+1$ times with sampling period θ , where $\tau = (N+1)\theta$. Each measurement is a “virtual node,” neither having defined physical location nor performing computation on the signal. Note, while the reservoir is characterized such that $N+1$ virtual nodes are supported, typically only N of those virtual nodes are used for computing $X(k)$ [5].

Since merely sampling $u(k)$ over τ would result in all virtual nodes having the same value, a masking function of fixed arbitrary weights (W_{in}) is applied to each $u(k)$ (Eq. (1)). The result is linearized then sequentially fed to the reservoir. In our experiments a uniform random mask varies in $[-1, +1]$ was used.

C. Characterization of TDR Performance

For very high accuracies, it is convenient to express the symbol error ratio (SER), defined as the fraction of the inputs $u(k)$ that are misclassified, thus $SER = 0$ indicates no misclassifications. Word error ratio (WER) is similar.

For physical reservoir systems, we not only care about the accuracy performance metric for a particular task, but also for the system’s overall noise tolerance and repeatability. In our tests, for each instance j of a task, W_{out}^j is computed; and this W_{out}^j is then tested against all the other trial signals. For j trials, this affords $j(j-1)$ unique training/testing pairs. A well-tuned, physically stable reservoir would demonstrate similar performance across the different trials. Note, the input mask W_{in} must be fixed for all j signals, else W_{out}^j will only work for signal j .

III. METHOD

Since the TDR topology is more algorithmically modular than the general ESN topology, it more readily lends itself to a hardware implementation. In this case, the TDR is realized using off-the-shelf telecommunications components (Fig. 1). Matlab scripts were used to perform initial input masking as well as the output post-processing steps [5]. However, designs are available in the literature to implement both the input and output layers computations in optical hardware [7, 8].

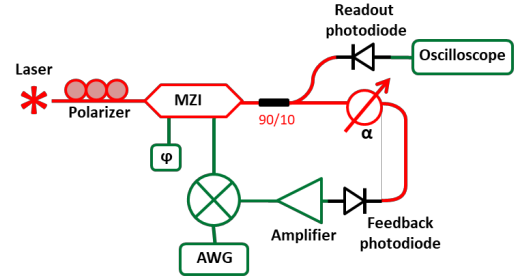


Fig. 1: Hardware schematic. Physical TDR consisted of Mach-Zehnder interferometer (MZI), arbitrary waveform generator (AWG), electrical amplifier, two photodetectors, fiber splitter (90/10), and tunable optical attenuator (α). Red lines denote optical components while green lines indicate electrical components.

A. TDR Experimental Setup

In our work, the optical source was a 1550nm distributed feedback laser source (Thorlabs WDM8-C-25A-20-NM) with output power typically ~ 10 mW. An intensity Mach-Zehnder interferometer (MZI) (EOSPACE AX-0MSS-20-LV) with 20 GHz bandwidth served as the nonlinear node, encoding an electrical signal as light intensity according to a \sin^2 transfer function. An arbitrary waveform generator (AWG) (Keysight M8190A) was used to feed the input to the reservoir. An optical splitter diverted roughly 10% of the optical signal to the output layer which consisted of a high-speed photodetector and an oscilloscope (Keysight DSO-S 804A digital storage oscilloscope). An RF amplifier (Picosecond PSPL5868) and a tunable optical attenuator (Newport 44203-01) were used in combination to quantify the fading memory of the reservoir. The characteristic delay of this reservoir was established only by the intrinsic delays in the electronic components and pigtail fiber lengths of the optical components. The measured delay time of the setup

was $\tau = 28.4 \pm 0.02$ ns. To the best of our knowledge, this is the fastest time-delay RC in network topology similar to Fig. 1 that has been implemented [4, 9, 26].

B. Memory Capacity

Fading memory, or the echo-state property, is a necessary characteristic of a reservoir. In [27], a linear memory technique to quantify the RC memory capacity is presented, but this technique involves actually performing RC, which may be tedious during setup. In this work, we instead present a very simple technique both to measure the reservoir characteristic delay and to approximate the appropriate fading memory for different tasks. The underlying principle of this approach is that the reservoir is a dynamic system capable of retaining an input signal with a constant time delay [28, 29] until the signal decays to the noise level.

An electrical spike was applied to the reservoir. The readout scope records the number of subsequent spikes as the signal cycles through the reservoir layer with period τ until the spike amplitude is diminished to the noise floor of the system. The fading memory of the reservoir is clearly a function of this spike train length. In our setup, the optical attenuator was tuned to adjust the fading memory capacity.

The experimental measurement of two settings of fading memory (3 pulses or 5 pluses) are shown in Fig. 2, which demonstrate the injected signal can be retained in the reservoir for 2τ and 4τ seconds, respectively. A maximum of 13 spikes have been observed in these experiments, which is useful for tasks where signals have long periods of temporal correlations; while signals with short temporal correlations need only a few spikes. Thus, this simple technique greatly helped in tuning reservoir performance across various tasks.

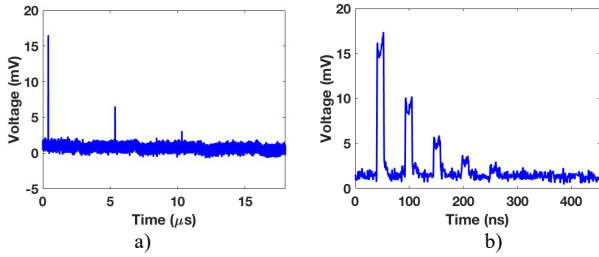


Fig. 2: Spike train characterizing fading memory of reservoir using a) a 1km optical fiber delay line, $\tau = 4.96 \pm 0.02\mu s$, illustrating rapid fading memory (2 repeating spikes), and b) no external delay line, $\tau = 28.4 \pm 0.2ns$, illustrating longer memory retention (5 repeating spikes).

C. Preamble Functions

One challenge in analyzing the reservoir output is identifying the reservoir response starting point at $x[u(k=0)]$ from the background system noise. [5] utilized a fraction of its input as the “warmup” signals for preamble and discarded them during the post-processing. In this work, we added a preamble function prior to the initial data point of the input signals. Two forms of preamble signals were used in this research, namely a) a ramp preamble [0, 1] and b) a high DC signal of 1. The comparison of NCE task using preamble functions against no appended preamble signals is presented in Section V, part B.

D. Averaging Points Per Node

In this work, the input AWG sampling frequency set the number of virtual nodes in the reservoir layer. The readout oscilloscope sampling frequency was often higher, oversampling the reservoir state. By oversampling, sequential measurements could be averaged together to form one virtual node, whose value is thus more resilient to measurement noise [5]. The number of averaged points per node, $\langle ppn \rangle$, was calculated as

$$\langle ppn \rangle = \frac{f_{s,out}}{f_{s,in}}, \quad (5)$$

where $f_{s,in}$ and $f_{s,out}$ are the sampling rate of the input AWG and readout oscilloscope, respectively. For example, operating the input AWG at 2 gigasamples per second (Gsa/s) and the readout oscilloscope at 20 Gsa/s, afforded 10 samples to be averaged to create one output virtual node ($\langle ppn \rangle = 10$).

IV. MATHEMATICAL MODEL FOR VIRTUAL NODE CONNECTIVITY

It is crucial to evaluate the optimal reservoir network size and reservoir states connectivity analytically, which leads to the calculation of the capacity C of a reservoir network. The information-theoretic approach summarized here offers crude estimate. For an improved method and more detailed mathematical treatment, one can refer to [30].

A reasonable starting point is to take the reciprocal of capacity, $\frac{1}{C} = |\{p^*\}|^{-1}$ of a fixed recurrent neural network (RNN) to be the a priori (or least biased) probability of getting the intended Boolean problem p^* from the untrained RNN. Thus, to train an RNN to perform a fixed problem, we will need to use $\ln[C]$ amount of training information. Applied to the discrete approximation for the continuous weights and biases in the RNN, we obtain the capacity, and the learning efficiency ratio in the second line of the following equations

$$C = |\{p^*\}| = \frac{K^N \times 2^{n_0 M^{(k-2) \times N}}}{N!} \quad (6)$$

$$q(p^*)/Q(p^*) = 2^{-(L+1)} \{N[\ln(K \times M^{k-2} - \ln N \times e^{-1} + n_0 \ln 2)]\} \quad (7)$$

where $q(p^*) = \ln[C]$ is the size of the required training set which we take to be equal to the number of training-input binary strings of length $L+1$, $Q(p^*) = 2^{L+1}$ is the total number of all the possible input binary strings for the Boolean problem, K and M is the number of accessible discrete values of the biases and weights, respectively. W_{res} is the reservoir hidden nodes connections, and n_0 is the number of nodes in the output layer. In deriving this useful expression, we have also assumed for concreteness that the connection of the reservoir network has regular degree $k = 4$ associated with the ring topology, and taken the number of accessible values of the hidden weights W_{res} be M . The fading memory is related to the degree k . A large k value indicates that the signal can be retained in the reservoir for a longer time. The quantitative relation between degree k and the fading memory will be studied in the future. We have also taken the input masks m_i to be fixed over the length of the binary input $u(t)$. Clearly, $\frac{q(p^*)}{Q(p^*)}$ is nearly linear with N since $n_0 < N$ that suggests an optimized virtual node number N exist for the maximum information capacity C , experimentally evidenced in the following section. Moreover, the dependence

of the above expression on the degree k of the connectivity of the reservoir nodes gives an effective means to tune the connectivity of the reservoir for an optimal learning efficiency.

V. RESULTS AND ANALYSIS FOR BENCHMARK TASKS

A. Sine/Square Wave Classification Task

In our work, 5 signals of the sine/square classification task were performed with 1,000 training and 1,000 testing points with reservoirs of $N = 28, 56, 70$ and 141 virtual nodes. The classification results (Fig. 3a) show a minimum SER with $N = 56$. Also, for a given virtual node number N , the SER can be lowered by increasing the $\langle ppn \rangle$ (increasing output sampling frequency) (Fig. 3b).

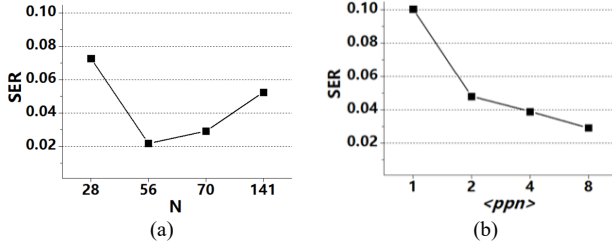


Fig. 3: For the sine vs. square wave classification task, a) SER as a function number of virtual node count and b) SER as a function of number of averaged ppn at $N = 70$.

With a higher signal injection rate, the electrical bandwidth of the photodetectors and the digitizer also need to increase in order to read out the high-speed signals without errors which poses challenges in the hardware realization.

B. Nonlinear Channel Equalization (NCE) Task

For the NCE task, 5 signals of 2,400 symbols were generated (1,000 training and 1,400 testing symbols) over several signal to noise ratios (SNR) = 12-32 dB. A tradeoff was again demonstrated when optimizing the RC for performance (Fig. 4a). Increasing the number of virtual nodes theoretically allowed for greater separation between individual symbols and, in turn, greater computational complexity of the system; however, increasing the AWG input sampling frequency also decrease the averaged samples per virtual node, increasing the effects of noise. Because of the fast time scales involved, further noise/reservoir robustness tradeoffs were not yet fully characterized, resulting in an increase in SER for the cleaner NCE signal (SNR = 32 vs. SNR = 28 dB) (Fig. 4b). We achieve the best results with a moderate fading memory (4-5 spikes) for this task.

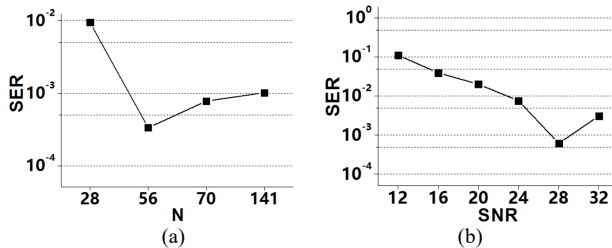


Fig. 4: For the nonlinear channel equalization task, a) SER as a function of the virtual node count and b) SER as a function of the input SNR for $N=56$ and $\langle ppn \rangle = 10$.

The NCE task was also used to study preamble signals for virtual node counts of $N = 56$ and 70. Each preamble signal was of duration τ , in the range $[0, 1]$ scaled by $\beta \times \max[u(k)]$, and sampled over N time steps. Because the shape and length of the preamble are known, the first point of the reservoir response is drastically more conspicuous (Fig. 5b, c vs. a). By appending a preamble signal to the beginning of u , the attained SER improved by more than 3 orders of magnitude (Fig. 5d).

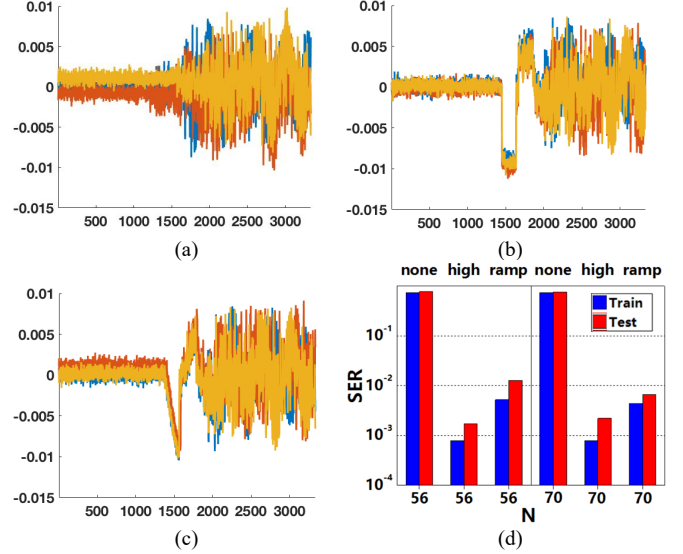


Fig. 5: Effect of preamble for determining $x(u(k=0))$ for a) none, b) high, or c) ramp preamble signal. Colors depict reservoir responses for different trials. All units A.U. d) NCE SER as a function of different preambles for $N=56$ and 70.

Due to fading memory effects, the shape of the preamble is repeated (though inverted) in the reservoir response; afterwards, the reservoir responses are visually fairly similar. In effect, the preamble helps stabilize the initial internal dynamics of the reservoir response over multiple trials. For this example case, the high preamble afforded the best reservoir performance. In general practice though, the high function was liable to push the reservoir into an unstable chaotic state due to the impulse response; thus the ramp preamble was used for the remaining of the experiments described.

C. Speech Recognition Task

For the speech recognition task, we used the TI-20 dataset (subset of TI-46 speech dataset) which is comprised of recordings of 5 female speakers pronouncing 10 digits $\{0, \dots, 9\}$ 10 times each (500 recordings total). These speech data were first preprocessed via the Lyon ear model, converting a 1D sound wave (pressure variations in time) into a 2D frequency in time matrix, called a cochleagram, of $Q = 78$ frequency channels over 24-53 steps (mean of 34.4) using a decimation of 400. Because of the small size of the audio dataset, there is high variability in the reservoir performance dependent upon an arbitrarily chosen test set. To reduce this variability and obtain realistic typical performance measures for the reservoir, the testing procedure for this task utilized k-fold cross validation [5, 9, 31]. In this work, the 500 signals were divided into 20 subsets, such that every signal was tested exactly once.

In comparison with [9], to process 1 million words a second given $\tau = 28.4$ ns, the average spoken digit duration needed to be ~ 34 steps, that is u is a 78×34 matrix. However, the difficulty

of the task is inversely proportional to the signal length of each spoken digit. The best WER was 5.5% using $N = 284$ virtual nodes where $\langle ppn \rangle = 2$ (Fig. 6). In [9], only a fraction of the reservoir was used by $u(k)$, allowing for 3 such values concurrent in their reservoir. With such a configuration they demonstrated a classification speed of $\sim 750,000$ words/s. We observed that a smaller fading memory (1 - 2 spikes) gives rise to the best classification results.

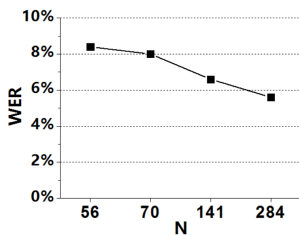


Fig. 6: Speech classification WER for TI-46 task as function of virtual node count.

VI. NOVEL TASKS RESULTS AND ANALYSIS

A. Respiratory Motion Prediction Task

During radiation therapy for cancer treatment, targets such as lung tumors will change position due to the patient's respiratory cycles. Machine learning based motion prediction is sought to reduce the radiation dose and minimize the exposure to surrounding healthy tissue. A tumor motion curve follows a semi-periodic pattern, and for stable, regular breathing patterns, simple model-based predictions can achieve good results [32]. In clinical practice, the breathing pattern can vary considerably during treatment due to changes in the patient's physical conditions such as coughing or nervousness as well as varying according to different patients. Early work by Lee *et al.* [33] and based on Puskorius *et al.* [34], using an extend Kalman filter (EKF) in combination with RNNs, generally have demonstrated good prediction results. RC therefore appears ideally suited for this task. We will compare our results based on clinical data with a long short-term memory (LSTM) approach [13].

Clinic data from 4 patients of lung cancer were provided (subset of [13]). The data were acquired by placing sensors on the patient's chest to record position movement at a 34 Hz sampling frequency ($\Delta t = 29.4$ ms). Typically image guided radiotherapy system reports system response latency of 50 ms - 1400 ms [14]. In this work, the current data point along with the previous 15 data points were used to predict the future tumor position 10 time steps ahead, enabling a 294 ms prediction window. Typical target metrics for the signal in a 300–800 ms prediction window are mean absolute error (MAE) < 0.5 mm and root mean square error (RMSE) = 0.67–1.2 mm [14-16,35].

The signals ranged in duration from 45-160 seconds, so for consistency, the last 20 seconds (600 points) of each signal were used for testing. For our optical TDR, $N = 70$ and $\langle ppn \rangle = 20$. Once the training data have been processed through the reservoir in $k\tau$ seconds, a single W_{out} per patient was calculated using Matlab within 30 - 40 ms. Each prediction (Eq. 2) is subsequently calculated within on an average of 11 μ s.

ORC predictions follow the actual breath patterns well (Table 1 and Fig. 7). In Table 1, NMSE, RMSE, and MAE for

motion forecasts of 30 ms (1 point), 147 ms (5 points), and 294 ms (10 points) were reported; and, as expected, the accuracy metrics dramatically improve inversely with the forecast distance. The larger errors occurring near the inflection points of the breathing pattern. Clinically, good alignment between the predicted and actual curves at the peaks and valleys of the breath patterns are more important than prediction errors at the inflection points as a threshold can be set to disable the radiation beam at the proximity of inflection points.

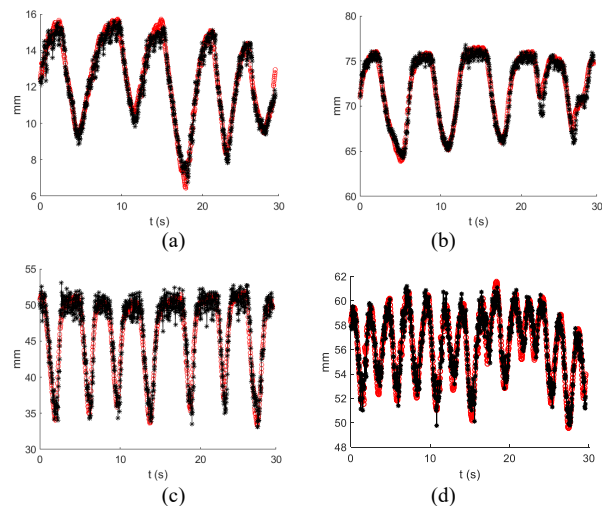


Fig. 7: Actual (red) and predicted ($t+294$ ms) (black) tumor displacement curves for sample a) #440, b) #805, c) #642, and d) #817.

An LSTM network with 3 hidden layers and 15 epochs [13] attained comparable results as the ORC prediction. LSTM networks have many more tuning parameters, potentially attaining greater prediction accuracy at the expense of additional computing resources. Several hours were required to train the LSTM network on just one signal (~ 1000 data points) as opposed to < 1 second training time using RC. Again, while a general model may be trained beforehand, the goal is for any machine learning technique to adapt to a patient's unique breathing cadence at the time of the operation. The ORC can be trained rapidly with minimal training data, making it a strong candidate towards real-time adaptive radiotherapy.

B. Perovskite Bandgap Prediction

Organometallic halide perovskites are a group of materials that have revolutionized the field of photovoltaics due to their optimal band gaps, high absorption, long carrier diffusion length, and many other suitable properties. The major obstacle in industrial deployment is the challenge of long-term stability due to moisture, heat and light. To address this, there have been considerable research efforts searching for alternative halide double perovskites with the atomic structure of a pair of 1^+-3^+ cations that constructs a basic formula unit of $A_2B^+B^{3+}X_6$. This strategy has opened the search space of halide perovskites to include Na, K, Cs, Cu, Ag, Au, In, and Tl as the monovalent cation, B^+ , and Sc, Y, Al, Ga, In, Tl, Sb, and Bi as the trivalent cation, B^{3+} , and combinations for many novel photovoltaic compound materials. The material energy bandgap is the most important property of a solar absorber and is commonly modeled by density functional theory (DFT) method and a higher order exchange correlation method. The former tends to overestimate the bandgap, and the latter, while more accurate,

Table 1. Performance of ORC on respiratory motion prediction task

Patient #	NMSE			RMSE			MAE		
	$t+30\text{ms}$	$t+147\text{ms}$	$t+294\text{ms}$	$t+30\text{ms}$	$t+147\text{ms}$	$t+294\text{ms}$	$t+30\text{ms}$	$t+147\text{ms}$	$t+294\text{ms}$
440	0.0047	0.0176	0.0465	0.1565	0.3033	0.4930	0.1156	0.2311	0.3798
642	0.0054	0.0306	0.0962	0.3927	0.9378	1.6610	0.2998	0.7241	1.2943
805	0.0026	0.0117	0.0386	0.1830	0.3915	0.7113	0.1402	0.3050	0.5375
817	0.0048	0.0768	0.2030	0.1884	0.7580	1.2350	0.1430	0.5587	0.9722

is computationally expensive, making a high throughput search inefficient. Machine learning models using random forest lately has been established to classify perovskites as potential solar cell absorbers based on its properties such as electronegativity, ionization potential, and atomic radius of the atoms composing the perovskite [25]. We explore whether photonic RC can also be used here.

The data set of study consisted of 220 data points constructed by randomly selecting elements from the A, B/B' and X regions, where the A site refers to $\{\text{Ca}^{2+}, \text{Sr}^{2+}, \text{Ba}^{2+}\}$; the B/B' site refers to $\{(4^+, 4^+), (3^+, 5^+), (2^+, 6^+)\}$ on the periodic table; and the X site refers to $\{\text{O}^{2-}, \text{S}^{2-}, \text{Se}^{2-}\}$ [25]. A total of 9 inputs are mapped to a single binary output in the optical RC network, viz, 1) energy bandgap in eV, 2) binary gap, 3) B-site electronegativity (EN), 4) B'-site EN, 5) average B/B' EN, 6) X-site EN, 7) difference between X-site EN and B-site EN, 8) difference between X-site EN and B'-site, and 9) difference between X-site EN and averaged B/B' EN.

Among the 220 data, the first 180 were used for RC network training while the last 40 data points were held out for testing. The predication performance of the ORC was compared with a random forest algorithm run on the *scikit-learn* Python module. The forest contained 80 trees with a maximum depth of 5 and they were optimized according to the Gini entropy criterion. The random forest machine learning model achieved 95% testing classification accuracy while the ORC achieved 94.5% accuracy. Splitting the examples 50% and 50%, the random forest achieved 68% accuracy yet the ORC attained 82% accuracy.

Machine learning models for material science promises to enable knowledge extraction from large datasets, revealing hidden correlations among data points beyond the established physical and chemical theories [36]. While not a temporal dataset, the underlying hyperdimensional network mapping of the optical TDR may be still used for the perovskite prediction task [37], with a minimal memory window. While minimal training data are necessary for highly temporally correlated data, 220 samples may be too few for optimal performance, though comparable with the random forest approach. Yet again, the computational speed of the physical reservoir layer and training algorithm favors the ORC. It takes ~ 30 s to run random forest algorithms on Python while it takes ~ 0.2 ms to execute the same task on the bench top reservoir kernel. A fully analog optical RC was demonstrated [7], suggesting a path towards eliminating digital pre- and post-processing and enabling optical RC to fully exploit its unprecedented computation speed.

VII. CONCLUSION

In this paper we have experimentally demonstrated a fast delay line optical reservoir computer with a short characteristic reservoir time of 28.4 ns. We have introduced the use of a preamble to remove ambiguity in post-processing and have characterized the performance of various preamble functions. We have also introduced a simple yet practical method to quickly characterize the fading memory of the ORC.

Beyond the standard RC benchmark tasks, we evaluated the reservoir computer on a respiratory motion prediction task and found performance accuracy to be comparable to more widely used machine learning algorithms but significantly more computationally efficient, making this approach highly attractive for real-time machine learning in medical embedded computing applications. We also introduced a perovskite bandgap prediction task and again demonstrated comparable results but with more computational efficiency. The competitive performance of ORC on the respiration motion prediction task and the material bandgap classification task demonstrate the algorithmic power of reservoir computing. Optical reservoir computing then, with additional orders of magnitude improvements in energy and speed efficiency via optics, is well positioned to make significant contributions in embedded machine learning applications, where strictly digital approaches are often inadequate in data processing speed.

ACKNOWLEDGMENT

This work was supported in part by the Air Force Research Laboratory under contract: SA2018-UP-020-NOA. Any opinions, findings and conclusions, or recommendations expressed in this material are those of the authors, and do not necessarily reflect the views of the US Government, the Department of Defense, or the Air Force Research Lab. The materials and results presented in this paper have been cleared for public release, unlimited distribution by AFRL, case number 88ABW-2019-2179. We would like to thank Dr. Mahmood Hameed for his valuable discussions and Juanyan Li for his help with the graph editing.

REFERENCES

- [1] Y LeCun, Y Bengio, and G Hinton, "Deep learning" *Nature*, **521**, 436–444, 2015.
- [2] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Proc. Adv. Neural Inf. Process. Syst.*, pp. 3104–3112, 2012.
- [3] R. Shafin, L. Liu, J. Ashdown, J. Matyjias, M. Medley, B. Wysocki, and Y. Yi, "Realizing green symbol detection via reservoir computing: an energy-efficiency perspective," *IEEE International Conference on Communications*, pp. 1–6, 2018.
- [4] Apostolos Argyris, Julian Bueno and Ingo Fischer, "Photonic machine

- learning implementation for signal recovery in optical communications", *Scientific Reports*, 8:8487, 2018.
- [5] Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman and S. Massar, "Optoelectronic Reservoir Computing", *Scientific Reports*, vol. 2, pp. 287, 2012.
 - [6] Francois Duport, Anteo Smerieri, Akram Akrou, Marc Haelterman, and Serge Massar, "Virtualization of a photonic reservoir computer", *J. of Lightwave Tech.*, Vol. 34, No. 9, pp. 2085-2091, 2016
 - [7] François Duport, Anteo Smerieri, Akram Akrou, Marc Haelterman and Serge Massar, "Fully analogue photonic reservoir computer," *Scientific Reports*, Vol. 6, Article number: 22381, 2016
 - [8] François Duport, Bendix Schneider, Anteo Smerieri, Marc Haelterman, and Serge Massar, "All-optical reservoir computing," *Opt. Express* 20, 22783-22795, 2012
 - [9] Laurent Larger, Antonio Baylón-Fuentes, Romain Martinenghi, Vladimir S. Udaltsov, Yanne K. Chembo, and Maxime Jacquot, "High-speed photonic reservoir computing using a time-delay-based architecture: million words per second classification", *Physical Review X*, 011015, 2017
 - [10] Binti Haya Afecqah Amalina, Chihiro Sugano, Kazutaka Kanno, Atsushi Uchida, "Implementation of optical feedback modulation in photonic reservoir computing", *Proceeding of OSA, Fio conference*, Washington DC, 2019.
 - [11] M. Bauduin, Q. Vinckier, S. Massar, F. Horlin, "High performance bio-inspired analog equalizer for DVB-S2 non-linear communication channel", *IEEE 17th International workshop on signal processing advances in wireless communications*, 2016
 - [12] Jeremy Vatin, Damien Rontani, and Marc Sciamanna, "Enhanced performance of a reservoir computer using polarization dynamics in VCSELs" *Optics Lett.*, Vol. 43, No. 18, pp. 4497, 2018
 - [13] Hui Lin, Chengyu Shi, Brian Wang, Maria F. Chan, Xiaoli Tang, Wei Ji, "Towards real-time respiratory motion prediction based on long short-term memory neural networks", *Physics in Medicine and Biology*, April, 2019
 - [14] Troy P. Teo, Syed bilal Ahmed, Philip Kawalec, Nadia Alayoubi, Neil Bruce, Ethan Lyn, and Stephen Pistorius, "Feasibility of predicting tumor motion using online data acquired during treatment and a generalized neural network optimized with offline patient tumor trajectories," *Med. Phys.*, 45: 830-845, 2018
 - [15] W Z Sun, M Y Jiang, L Ren, J Dang, T You, F-F Yin, "Respiratory signal prediction based on adaptive boosting and multi-layer perceptron neural network," *Phys Med Biol*. Vol. 62, pp. 6822-6835, 2017
 - [16] E. W. Pepin, H. Wu, Y. Zhang, and B. Lord, "Correlation and prediction uncertainties in the CyberKnife Synchrony respiratory tracking system," *Med. Phys.*, Vol. 38, pp. 4036-4044, 2011
 - [17] S. Shi, J. Gao, Y. Liu, Y. Zhao, Q. Wu, W. Ju, C. Ouyang, R. Xiao, "Multi-scale computation method: their application in lithium-ion battery research and development" *Chinese Phys. B*, 25, 018212. (2016)
 - [18] Y. Liu, T. Zhao, W. Ju, S. Shi, "Materials discovery and design using machine learning", *J. Materiomics*, Vol. 3, pp. 159-177, 2017
 - [19] J. Lee, A. Seko, K. Shitara, K. Nakayama, I. Tanaka, "Prediction model of band gap for inorganic compounds by combination of density functional theory calculations and machine learning techniques" *Phys. Rev. B*, 93, 115104, 2016
 - [20] L. Ward, A. Agrawal, A. Choudhary, C. Wolverton, "A general-purpose machine learning framework for predicting properties of inorganic materials" *Nat. Computational Materials*, 2, 16028, 2016
 - [21] T. Gu, W. Lu, X. Bao, N. Chen, "Using support vector regression for the prediction of the band gap and melting point of binary and ternary compound semiconductors" *Solid State Sci.*, 8, pp. 129, 2006
 - [22] G. Pilania, J. E. Gubernatis, T. Lookman, "Multi-fidelity machine learning models for accurate bandgap predictions of solids", *Comput. Mater. Sci.*, Vol. 129, pp. 156, 2017
 - [23] M. Fernandez, J. I. Abreu, H. Shi, A. S. Barnard, "Machine learning and genetic algorithm prediction of energy differences between electronic calculations of graphene nanoflakes" *ACS Comb. Sci.*, 18, pp. 661, 2016
 - [24] L. Ward, C. Wolverton, "Atomistic calculations and materials informatics: a review" *Current Opinion in Solid State and Materials Science*, Vol. 21, pp. 167-176, 2017.
 - [25] M.L. Agiorgousis, Y.-Y. Sun, D.-H. Choe, D. West, S. Zhang, "Machine learning augmented discovery of chalcogenide double perovskites for photovoltaics," *Adv. Theory Simul.* 2 1800173, 2019
 - [26] Julián Bueno, Daniel Brunner, Miguel C. Soriano, and Ingo Fischer, "Conditions for reservoir computing performance using semiconductor lasers with delayed optical feedback" *Optics Express* Vol. 25, Issue 3, pp. 2401-2412, 2017
 - [27] H. Jaeger, "Short term memory in echo state networks," *GMD Rep.* 152, pp. 60, 2002
 - [28] X. Steve Yao and Lute Maleki, "Optoelectronic oscillator for photonic systems", *IEEE J. of Quan. Electro.*, Vol. 32, No. 7, pp. 1141-1149, 1996
 - [29] Laurent Larger, Pierre-Ambroise Lacourt, Stephane Poinsot, and Marc Hanna, "From flow to map in an experimental high-dimensional electro-optic nonlinear delay oscillator", *Physical Review Letts*, 95, 043903, 2005
 - [30] C. Lim, "Learning efficiency coefficient in ORC", under review by *Phys Rev E*, 2019.
 - [31] Q. Vinckier, F. Duport, M. Haelterman, and S. Massar, "Information processing using an autonomous all-photonic reservoir computer based on coherently driven passive cavities", *Proc. of Frontiers in Optics, OSA Technical Digest*, paper FTu3B.6. pp. 2015
 - [32] H. Wu, G.C. Sharp, B. Salzberg, D. Kaeli, H. Shirato and S. B. Jiang, "A finite state model for respiratory motion analysis in image guided radiation therapy" *Physics in Medicine and Biology*, 49, 5357, 2004
 - [33] S. J. Lee, Y. Motai and M. Murphy, "Respiratory motion estimation with hybrid implementation of extended Kalman filter", *IEEE Transactions on Industrial Electronics*, Vol. 59, pp. 4421-4432, 2012
 - [34] G. V. Puskorius and L. A. Feldkamp, "Neurocontrol of nonlinear dynamical systems with Kalman filter trained recurrent networks" *IEEE Transactions on neural networks*, Vol. 5, pp. 279-97, 1994
 - [35] P. Verma, H. Wu, M. Langer, I. Das and G. Sandison, "Survey: real-time tumor motion prediction for image-guided radiation treatment," *Computing in Science & Engineering*, vol. 13, no. 5, pp. 24-35, 2011
 - [36] Mitsutaro Umehara, Helge S. Stein, Dan Guevarra, Paul F. Newhouse, David A. Boyd and John M. Gregoire, "Analyzing machine learning models to accelerate generation of fundamental materials insights", *NPJ Computational Materials*, Vol. 5, 34, 2019
 - [37] S. Ortín, M. C. Soriano, L. Pesquera, D. Brunner, D. San-Martin, I. Fischer, C. R. Mirasso, and J. M. Gutierrez, "A unified framework for reservoir computing and extreme learning machines based on a single time-delayed neuron," *Sci. Rep.* Vol. 5, 2015.