Reservoir computing using networks of memristors: effects of topology and heterogeneity

J. B. Mallinson [a], Z. E. Heywood [a,b], R. K. Daniels [a], M. D. Arnold [c], P. J. Bones [a,b] and S. A. Brown [a]

Reservoir computing (RC) has attracted significant interest as a framework for the implementation of novel neuromorphic computing architectures. Previously attention has been focussed on software-based reservoirs, where it has been demonstrated that reservoir topology plays a role in task performance, and functional advantage has been attributed to small-world and scale-free connectivity. However in hardware systems, such as electronic memristor networks, the mechanisms responsible for the reservoir dynamics are very different and the role of reservoir topology is largely unknown. Here we compare the performance of a range of memristive reservoirs in several RC tasks that are chosen to highlight different system requirements. We focus on percolating networks of nanoparticles (PNNs) which are novel self-assembled nanoscale systems that exhibit scale-free and small-world properties. We find that the performance of regular arrays of uniform memristive elements are limited by their symmetry but that this symmetry can be broken either by a heterogeneous distribution of memristor properties or a scale-free topology. The best perfomance across all tasks is observed for a scale-free network with uniform memistor properties. These results provide insight into the role of topology in neuromorphic reservoirs as well as an overview of the computational performance of scale-free networks of memristors in a range of benchmark tasks.

Introduction

The essential concept in neuromorphic computing[1] is that novel hardware systems capable of brain-like information processing can be designed and implemented using physical devices. Compared with conventional computing hardware, neuromorphic systems promise superior performance in tasks at which the biological brain excels (e.g. classification and time-series prediction), while consuming significantly less energy and avoiding the imminent scalability issues faced by the von Neumann architecture.[2] Much effort has been invested in developing neuromorphic integrated circuits based on complementary metal-oxide semiconductor technologies[3,4] but additionally several nanoscale systems and components show promise, especially spintronic oscillators,[5] and self-organised networks of memristive devices.[6,7] Self organised systems, which include carbon nanotube,[8,9] nanowire[10–14] and nanoparticle[15–18] networks, are appealing because they have the potential to naturally integrate large numbers of memristive devices into brain-like structures that are difficult (or practically impossible) to attain using top-down processes, with low fabrication costs.

Networks of memristive devices are especially attractive for implementation of a brain-inspired computational framework called reservoir computing (RC). Exemplified by the echo-state network (ESN)[19] and the liquid-state machine,[20] RC was first implemented in software as a solution to the high cost of training recurrent artificial neural networks (RNNs).[21] In RC, only the output connections are adapted while the rest of the network (the "reservoir") has fixed connection weights, both simplifying the training process and drastically reducing the associated cost. In *physical* reservoir computing, this simplification translates to the ability to utilize the complex nonlinear dynamics of memristive elements without requiring precise adaptation of the connections between them to facilitate learning. Several hardware reservoirs have been realised using nanoscale memristive devices (see refs. 22,23 for reviews). Among these, percolating networks of nanoparticles (PNNs) are particularly interesting due to their scale-free and small-world topology and associated long range correlations[24,25] and critical avalanche dynamics,[16,26] each thought to be important features of the biological brain.[27–29] "Scale-free networks" are defined by power law degree distributions, where the degree of each node is the number of connections it has to other nodes. "Small-world networks", which may or may not be scale-free, are characterised by a high level of clustering between nodes, coupled with a short mean path length.

It has been demonstrated that information processing is optimised in software-based ESNs with scale-free topologies,[30,31] however relatively little attention has been given to exploring the effects of reservoir topology in RC systems based on physical substrates, such as networks of memristors. Demonstration of RC in real-world physical systems is challenging because a number of physical network parameters must be optimised. We emphasise that it is therefore important to simulate these systems in order to gain insight into the correct choices of parameters, timescales etc. Here we simulate physical RC systems. The reservoir dynamics are a product of the reservoir structure coupled with the underlying mechanisms of information transfer within the reservoir, which may differ vastly between ESNs and various physical reservoir architectures. For example, in memristive electronic networks the dynamics are produced by the memristors and their interactions are driven by the evolution of currents and voltages throughout the network. The constraints imposed by Kirchhoff's laws (e.g. that current must be conserved at junctions) are not present in software-based ESNs. The relationship between the structure and functionality of the reservoir depends crucially on its topology, and so it is

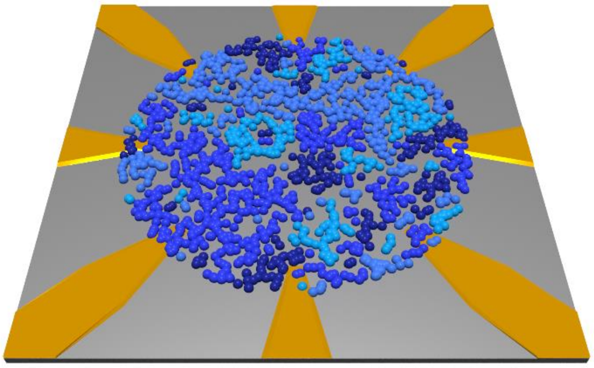

Fig. 1: Schematic representation of a percolating network of nanoparticles (PNN) with multiple gold electrodes. The different particle colours represent individual groups of well-connected particles which are separated by memristive tunnel gaps (MTGs).

important to establish the impact of the topology of specific memristive networks[32,33] on their performance in RC tasks.

Our objective is to compare the performance of scale-free networks of memristive devices with that of other network architectures. We therefore focus on a comparison of PNNs, which have been shown to feature scale-free connectivity,[25] and regular arrays, and consider networks of both uniform and heterogeneous memristive devices (see detailed discussion below). Importantly, the memristive device properties are identical in all cases, allowing the performance of PNNs to be benchmarked against that of other memristive networks, i.e. square arrays and scale-free networks of uniform memristive elements (labelled UPNNs – see discussion below). We use physically realistic simulations to investigate the relationship between reservoir structure and performance in chaotic time-series prediction, nonlinear transformation and temporal classification tasks. We systematically examine the dependence of task performance on (i) the characteristic time scale of the memristors, (ii) the input signal amplitude, (iii) the nonlinearity of the memristor response, (iv) the diversity of the reservoir outputs, and (v) the heterogeneity of the memristors. The best RC performance achieved is obtained using scale-free networks of uniform memristive elements.

Modelling

Networks

A percolating network of nanoparticles (PNN) comprises metallic nanoparticles (Fig. 1) deposited onto an atomically smooth insulating substrate (the experimental system is described in detail in refs. 15,34). Numerical simulations have been used previously to show that the experimental networks of nanoparticles are well described by continuum percolation models.[24,26,35,36] Simulations of both the electrical and neuromorphic properties of PNNs have previously been shown to be in excellent agreement with experimental results,[24,37,38] confirming that the model of percolation with tunnelling provides an accurate description of the real physical system. Conducting particles are represented by uniform discs which are placed randomly on a plane and allowed to overlap. We focus on system sizes of 200×200 particle diameters, which are sufficiently large to capture the complex dynamics but small enough to limit the computational cost of simulations.[24] As particles are deposited onto the substrate, they form groups of well-connected particles which are separated by nanoscale memristive tunnel gaps (MTGs). The charging energy and resistance between overlapping particles is assumed to be negligible so that the only resistance in the network comes from MTGs.[35] Particles are deposited until a surface coverage of 65% is reached, just short of the percolation threshold (~68% for continuum percolation[39]) so that no single group of particles spans the entire plane. The resulting groups of particles have fractal geometry and provide a scale-free network structure.[25] Fig. 2a shows the distribution of voltages across a PNN. Fig. 2b shows a map of the network in which the nodes are the centroids of the groups of particles and the connections between nodes are MTGs. As detailed in the caption of Fig. 2, many groups have similar potentials (and so have the same colours) and so there are many more nodes that have the same colour in Fig. 2b than are immediately visible in Fig. 2a. We emphasise that we model the MTGs as standard memristors (i.e. ideal voltage-controlled variable-resistance memristors[40]) operating in the low-voltage regime. Hence, the MTG networks are completely equivalent to networks of standard memristors.[10]

In order to examine the relationship between reservoir structure and performance, we compare the PNNs with regular (square) arrays of MTGs (Fig. 2c). In both PNNs and arrays, particle groups which overlap the left-hand (right-hand) edge of the network are considered to be input (output) electrodes. In PNNs of size 200×200 particle diameters, we find that: (i) the number of output electrodes which are electrically coupled to the network is most commonly $N_{out} = 12$, (ii) mean path length across the PNNs is typically $\langle L_P \rangle = 12$, and (iii) the number of MTGs is $N_{MTG} \sim 2000$. For the comparison with PNNs we consider two different array sizes: 12×12 nodes ($N_{out} = \langle L_P \rangle = 12$, $N_{MTG} = 242$), which has a similar $\langle L_P \rangle$ to the PNNs and 36×36 nodes ($N_{out} = \langle L_P \rangle = 36$, $N_{MTG} = 2450$), which has a similar N_{MTG} to the PNNs. For the 36×36 arrays, the 36 outputs are grouped such that *all networks have $N = 12$ output electrodes*. See S1 (ESI†) for details on the network statistics and choice of array sizes.

Deposition of the particles which comprise a PNN is an inherently stochastic process, as particles land in random positions on the substrate. Consequently, the MTGs which form as a result of the deposition have a wide range of initial gap sizes D (see S1, ESI† for details on gap size distributions and their limits, especially Fig. S1d). As well as investigating the effect of reservoir topology by comparing the scale-free PNNs with regular array networks, we also consider the effect of heterogeneity in the MTG sizes. PNNs, with their naturally heterogeneous initial MTG sizes (hereafter referred to as HPNNs), are compared with uniform arrays (UAs) where all MTG sizes are the same ($D_{\mathrm{uniform}} = 0.05$ particle diameters) and heterogeneous arrays (HAs) with MTG sizes randomly distributed over the same range as in the HPNNs. For completeness, we also compare these results with PNNs which have uniform initial MTG sizes (UPNN). Hence the present results additionally provide a comprehensive overview of the computational performance of scale-free networks of memristors in a range of benchmark tasks.

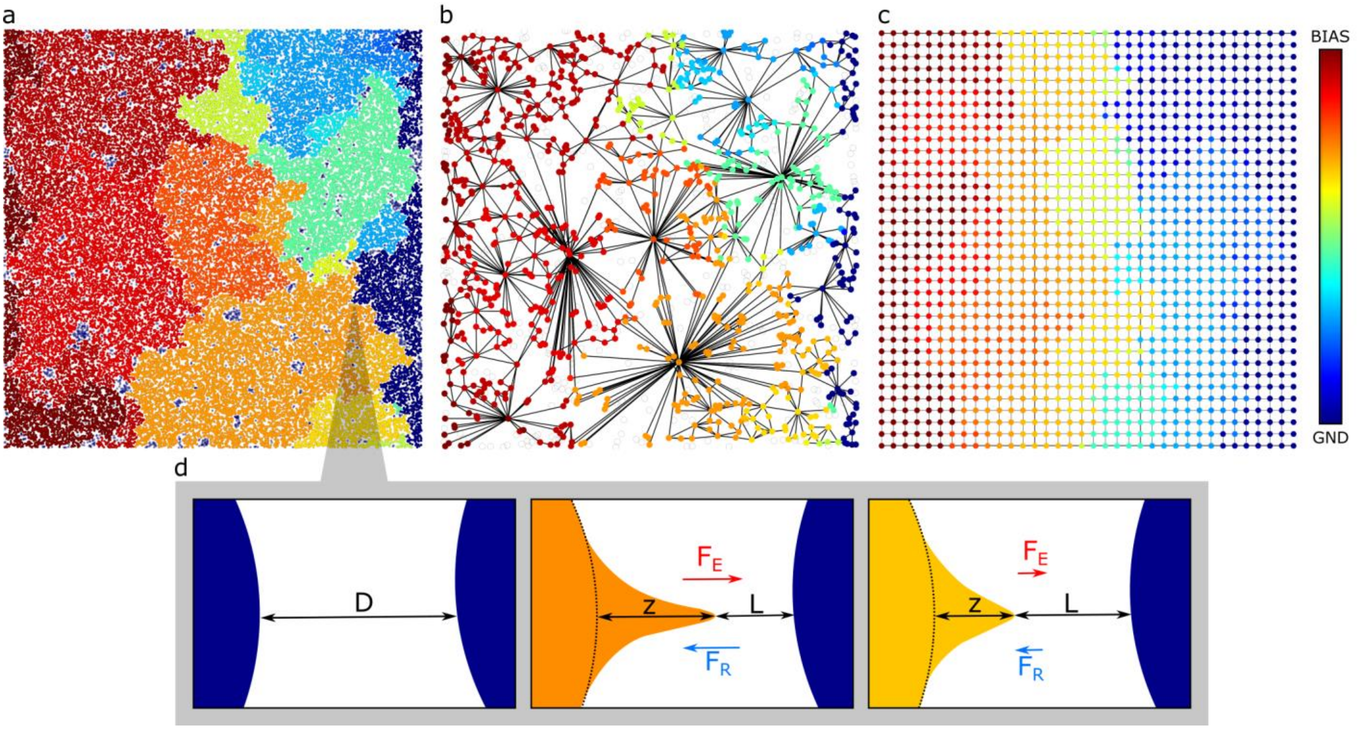

Fig. 2: Network topologies and hillock growth/relaxation. (a) Schematic particle representation of a percolating network of nanoparticles (PNN) of size 200×200 particle diameters, with bias applied on the left side of the network and the right side held at ground potential. Particles are represented by overlapping discs which form groups, the colour of which represents the potential on that group (relative to ground). (b) Vertex graph representation of the percolating network shown in (a). Nodes represent the geometric centres of the individual groups while edges represent tunnel gaps between group boundaries. Note that there are many more groups than apparent in (a) due to groups with similar potentials falling into the same colour bin. (c) Graph representation of a heterogeneous 36x36 array network where each edge represents a memristive tunnel gap of varying size D. Note that in (a-c), the colours correspond to bins of potentials and so there is greater variation in node potential than is apparent. (d) A schematic showing the growth and relaxation of a hillock in a memristive tunnel gap (MTG). Left: MTG with no bias applied to the network. Middle: A hillock of height z forms in response to bias applied to the left side of the network. F_E represents the electric force driving hillock growth while F_R represents the restorative force induced by surface tension. Right: Under reduced bias the hillock relaxes to a smaller height z, and both F_E and F_R decrease proportionately. Note that these schematics show the hillock at dynamical equilibrium, i.e. once all gaps in the network have responded fully to a constant bias.

Memristive Tunnel Gaps

When voltage stimuli are applied to the input electrodes of a network, tunnel currents flow through the MTGs.[24,35,36] Each MTG i is assigned a tunnelling conductance G_i based on its effective gap size L_i:[24,36]

$$G_i = \alpha e^{-\beta L_i}, \tag{1}$$

where α and β are constants (α = 1 and β = 200).[35] A sufficiently large applied voltage leads to electric field-driven surface diffusion processes[37] which cause atoms to migrate within the tunnel gaps, accumulating into "hillocks" as shown in Fig. 2d. Surface energy effects oppose the growth of the hillock[41] such that the changes in hillock of height z_i that result from a local potential V_i are governed by:[10]

$$\frac{dz_i}{dt} = \frac{1}{T}\left[\frac{\mu V_i}{D_i - z_i} - \kappa z_{i,}\right], \tag{2}$$

The effective gap size of the MTG at any time is $L_i = D_i - z_i$ where D_i is the initial size of gap i and μ and κ are the parameters which govern the hillock growth and relaxation respectively. The parameter values (μ =0.346 nm^{2}V^{-1}, κ = 0.038 (dimensionless)) are chosen to give similar timescales for the growth and relaxation, as discussed in more detail in ref. 10. We note that the theoretical model described by Eq. (2) is a based on observations hillock growth and relaxation in PNNs[15] and in similar memristive junctions found in other systems.[10,41] Further, we emphasise again that Eq. (2) describes dynamics that are completely equivalent to those of the standard memristor reported in ref. 40, in which both volatile and non-volatile switching may occur. In the low-voltage tunnelling regime considered here, the memristive behaviour is volatile. The results obtained here are therefore applicable to a broad range of square arrays and scale-free networks composed of standard memristors and the range of different D_i values for the MTGs correspond to heterogeneous characteristics of those memristors.

The characteristic time scale of the MTG (i.e. memristor) response, T in Eq. (2), is the same for all MTGs in the network. This choice was made because of the lack of any clear experimental evidence as to the relationship between response time and gap size. We expect that the distribution of response times in experimental systems will lead to greater richness in the network dynamics, and hence to better RC performance than that demonstrated here. T is a key parameter in the simulations but it is worth noting that in experiments this response time is controlled by the intrinsic physics, and the input rate is scaled to match the intrinsic timescale of the device.

The growth and relaxation of hillocks (Eq. (2)) modulate the effective size of tunnel gaps, both altering the local electric fields and tunnelling currents at each MTG (Eq. (1)). If the electric field within a MTG exceeds some threshold, its hillock grows across the gap to form a conducting filament, resulting in a sharp

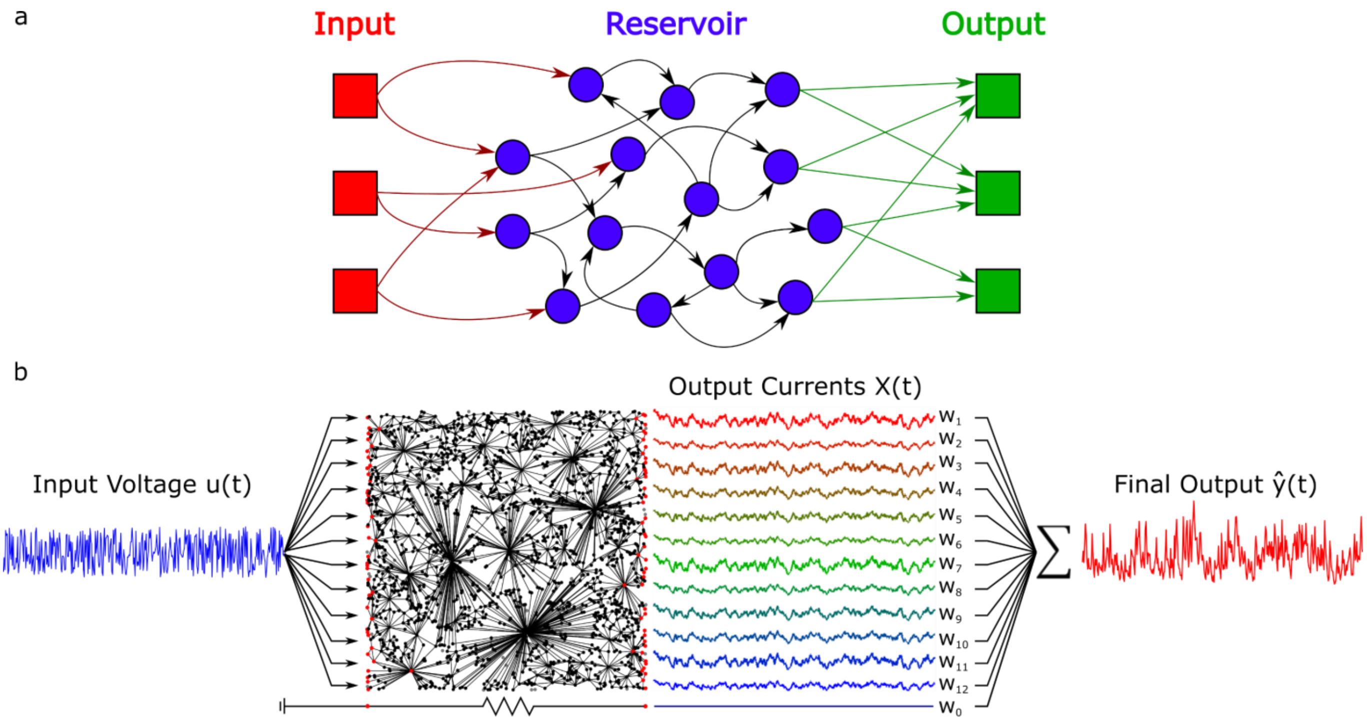

Fig. 3: Reservoir Computing Schematics. (a) A schematic representation of a conventional RC scheme. Input and reservoir connections have fixed random weights and only the output connections are trained. (b) A schematic showing how a PNN (or array network) can be used as a physical reservoir to perform the NARMA-10 task. Input voltages are applied to all left-hand edge groups of the reservoir network. Output signals are taken from all right-hand edge groups (held at ground potential) which transmit non-zero currents. A bias signal can be attained by measuring the current through a parallel resistor under DC bias. Output signals are each weighted and summed to produce the final output which should approximate the chaotic NARMA-10 time series.

conductance change (i.e. switching).[15,37] In the present work, we focus on the low input voltage regime in which hillocks modulate the size of the tunnel gaps, but do not form complete filaments, so as to maintain the equivalence with standard memristors[40] and avoid complications that result from avalanching effects.[11]

The response of an individual MTG varies strongly with the local potential $V_{i,t}$.[10] In a scale-free network where there are many MTGs interconnected in both series and parallel configurations, there is a wide range of local potentials, especially when the MTG sizes are widely distributed as they are in HPNNs. The range of local potentials and gap sizes therefore result in varied MTG responses and we emphasise that these lead to rich network dynamics and diverse reservoir outputs.

A schematic of a generalized reservoir computing (RC) scheme is shown in Fig. 3a and the physical implementation of RC using a PNN is shown in Fig. 3b. As described above the reservoir comprises a network of non-linear memristive elements (i.e. MTGs). We feed input signals into the system as sequences of voltages $u(k)$ applied to electrodes on one side of the network while electrodes at the opposite side are held at ground potential. The ensemble of currents $X(k)$ flowing through $N = 12$ output electrodes constitutes the reservoir output, which is an N-dimensional nonlinear representation of the (1-D) input sequence of applied voltages. $X(k)$ is then augmented with a constant bias signal resulting in a set of $N + 1$ predictor variables which are used to train the output layer weights w. The final output $\hat{y}(k)$ is a linear sum of weighted predictor variables which should approximate some target function $y(k)$ as closely as possible (see Methods for details on training). The index k is an integer which measures progression of time in 'time steps' which are defined to be the time between discrete input signals. Thus, the MTG response time T is also expressed in time steps. In practical implementations the input timing is chosen in accordance with the timescale of the memristor response.

Results and Discussion

We performed five RC tasks (NARMA-2, NARMA-10, Memory Capacity (MC), Nonlinear Transformation (NLT) and Waveform Discrimination (WD), see Methods for task details) using three different input amplitudes (maximum voltages $\mathrm{V_{max}} = 0.2$ V, 0.5 V and 1.5 V, minimum voltage $\mathrm{V_{min}} = 0.1$ V). Larger input amplitudes cause the hillocks which form inside the MTGs to grow to larger heights z thereby modulating the effective gap size L more strongly, thus producing more nonlinear responses from the reservoirs. This range of $\mathrm{V_{max}}$ allows for the exploration of the effects of nonlinearity while avoiding the complete filament formation within MTGs which occurs at higher voltages. The relation between $\mathrm{V_{max}}$ and nonlinearity is illustrated in Fig. 4, which shows examples of the normalized total output current $\mathrm{I_{tot}^*}$ ($\mathrm{I_{tot}^*} = \mathrm{I_{tot}}/\mathrm{max(I_{tot})}$ where $\mathrm{I_{tot}}$ is the sum of all output electrode currents) from each type of network in response to slow voltage sweeps V_{app} (I-V characteristics are plotted against the normalized applied voltage $\mathrm{V^*} = \mathrm{V_{app}}/\mathrm{V_{max}}$, see Fig. 4 caption). These I-V characteristics represent the effective transfer function of the networks for three different sweep amplitudes: $\mathrm{V_{max}} = 0.2$ V (blue), 0.5 V (green) and 1.5 V (red). It is clear that for larger $\mathrm{V_{max}}$, the nonlinearity increases. This effect is less obvious in the 36×36 UA because the mean path length is ~3 times longer than that of the other reservoirs and the applied voltage is distributed over a larger number of MTGs (see S1, ESI†).

The I-V characteristics in Fig. 4 are for the total sum of output currents and yield a single curve for each case. However, there are in fact N output currents for each network and in principle each output has its own unique effective transfer function. This is demonstrated in Fig. S2 (ESI†), which shows the individual normalized output currents I_i^* from each output electrode i when subjected to much faster voltage sweeps. As $\mathrm{V_{max}}$ increases from 0.2 V to 1.5 V the level of both nonlinearity and hysteresis increases for all of the reservoir types. Furthermore, as $\mathrm{V_{max}}$ increases, the output currents from each reservoir become more diverse, reflecting an increase in the richness of the reservoir dynamics (except for the UAs, as discussed further below).

PNNs and Uniform MTG Arrays

As described above, our objective is to elucidate the roles of topology and MTG heterogeneity in reservoir computing performance for scale-free PNNs and regular square arrays. To set a baseline for comparison, we begin by comparing performance of the PNNs, which naturally have heterogeneous MTG sizes (HPNN), with that of simple uniform arrays (UAs) in which each MTG in the array has the same initial gap size $D_{\mathrm{uniform}} = 0.05$ particle diameters. Note that the UAs are perfectly-ordered ideal systems which serve as a useful starting point for investigating the roles of reservoir structure and MTG heterogeneity. We expect these networks to perform poorly in RC tasks (see discussion on symmetry below). Performance is shown in Fig. 5 for all five tasks using a range of characteristic time scales T: NARMA-2 (Fig. 5a), NARMA-10 (Fig. 5b), MC (Fig. 5c) NLT (Fig. 5e), WD (Fig. 5f and g). Each point in Fig. 5 is an average obtained from a number of different network instances and input sequences (see captions for details), and the shaded bands show the standard error. Fig. 5d and h show the rank of the predictor matrix X for the NARMA/MC and WD tasks respectively. Note that the input data and the network responses are identical for NARMA-2, NARMA-10 and MC – the only thing that changes is the target function. Therefore, rank(X) is the same for all three tasks. The three columns from left to right correspond to $\mathrm{V_{max}} = 0.2$ V, 0.5 V and 1.5 V, and the three colours used in each panel represent the 12×12 UA (purple), 36×36 UA (light green) and HPNN (red). S3 (ESI†) contains examples of target function $y(k)$ overlaid with the final output $\hat{y}(k)$ for each type of network and for each task. "Optimal performance" is defined as the smallest NMSE (Normalized Mean-Square Error, see Methods) value as a function of T for the NARMA, NLT and WD tasks and the largest MC and SCORE values for the MC and WD tasks. Better performance is indicated by the arrows to the right of each panel.

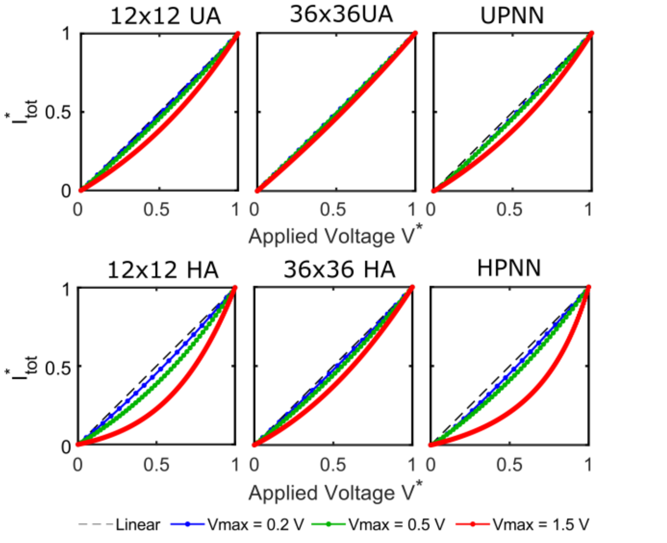

Fig. 4: Network nonlinearity at different applied voltages. Normalized I-V characteristics of representative networks showing how the effective nonlinearity of the different reservoir transfer functions increase with input amplitude. $\mathrm{I_{tot}^*} = \mathrm{I_{tot}}/\mathrm{max(I_{tot})}$ where $\mathrm{I_{tot}}$ is the sum of all output electrode currents. $\mathrm{V^*} = \mathrm{V_{app}}/\mathrm{V_{max}}$ where $\mathrm{V_{app}}$ is the applied voltage and $\mathrm{V_{max}} = \mathrm{max(V_{app})}$. The voltage sweep was applied slowly with respect to the network response times (period = 300 time steps, T = 1 time step) such that the network response is saturated at each discrete voltage and no hysteresis is observed. Dashed lines with slope 1 are shown for reference.

We begin by considering the optimal performance of HPNNs and influence of the MTG response time (T) for each task. HPNNs perform well in time-series prediction and recall tasks (i.e. NARMA and MC). They achieve average NMSE values as low as 0.17 for NARMA-2 (Fig. 5a) and 0.45 for NARMA-10 (Fig. 5b), with the larger NMSE for the NARMA-10 task reflecting the fact that it is more challenging because it requires more memory. This requirement is also reflected in the dependence of NMSE on T, with the best performance arising from very short MTG response times for NARMA-2 ($T = 1$), and longer response times for NARMA-10 (T~5; the longer response time allows the reservoir to retain information about inputs from the more distant past). For the HPNNs, the MC is as high as 3.75 (Fig. 5c), which reflects their ability to effectively recall past information. Curiously, MC does not depend strongly on T which suggests that the reservoir's memory results from the

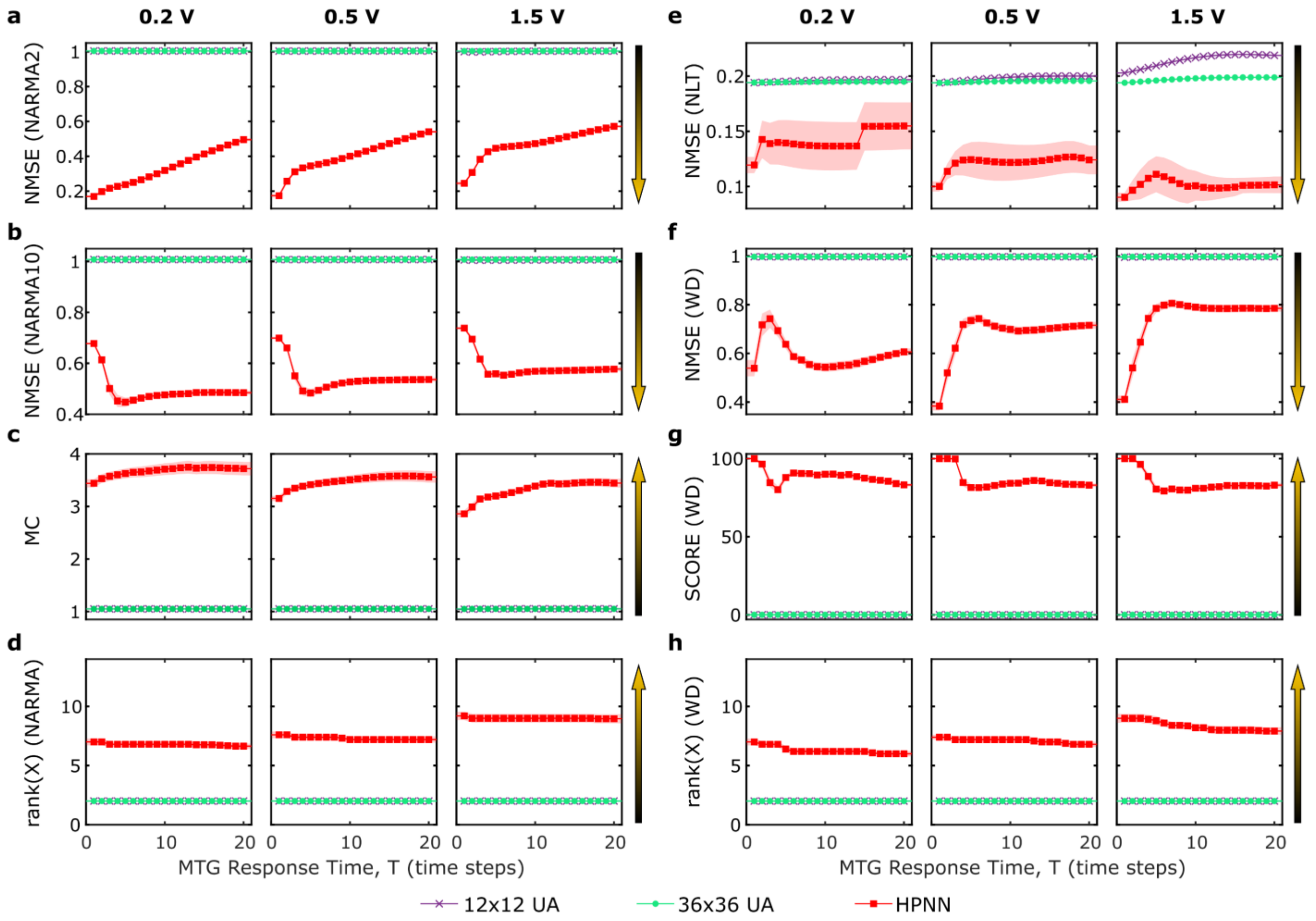

Fig. 5: Task performance for PNNs and Uniform Arrays. For both (a-d) and (e-h), the three columns correspond to three different input amplitudes (V_{max} = 0.2 V, 0.5 V, 1.5 V with V_{min} = 0.1 V for all V_{max}). The arrows indicate the direction of better performance for each panel. All results are the mean of five network realizations and five input sequences (25 trials total) except for NLT (e) which has only one input sequence. Shaded areas correspond to the standard error. All results are presented as a function of the network response time, T. (a) NARMA-2 and (b) NARMA-10 task performance expressed as the NMSE. (c) Linear memory capacity. (d) The rank of the predictor matrix X corresponding to (a-d). (e) NLT and (f) waveform discrimination task performance expressed as the NMSE. (g) Waveform discrimination task performance expressed as the % of correctly classified waveforms. (h) The rank of the predictor matrix X (during waveform discrimination), corresponding to the number of linearly independent outputs from the reservoir. The maximum possible rank is 13, corresponding to 12 electrode outputs and one constant bias term.

interaction of many MTGs (i.e. network dynamics) rather than the response time of individual MTGs.

HPNNs also perform well at temporal classification and transformation tasks (i.e. WD and NLT). For the NLT task (Fig. 5e), HPNNs achieve a mean NMSE as low as 0.09. Smaller T gives slightly better performance than larger T since the NLT task does not require a significant amount of memory. For the WD task, the lowest NMSE ~0.38 (Fig. 5f) and highest classification score of 100% (Fig. 5g) are again obtained for small T. This is because the reservoir state must capture memory of at least one previous input value (waveforms each have eight values) in order to successfully separate the sine and square waveforms.[42] However, separability suffers at larger T as the reservoir state depends too strongly on previous elements of the randomised sequence of sine/square waveforms. Thus, it is clear that the optimal characteristic time scale of the MTG response T is highly dependent on the memory requirements of the task.

We now consider how the optimal performance for each task relates to the amplitude of the input signal (i.e. V_{max}). The optimal performance for NARMA and MC becomes poorer with increasing V_{max} (NMSE for the NARMA tasks increases in Fig. 5a and b, while the MCs decrease in Fig. 5c). On the other hand, the NLT performance improves with increasing V_{max} (NMSE decreases in Fig. 5e), and the WD performance is optimized at the intermediate V_{max} = 0.5 V (Fig. 5f and g, though these results appear somewhat less sensitive to V_{max}). The dependence of performance on V_{max} for the various tasks can be understood as a consequence of the "memory-nonlinearity trade-off" previously observed in echo-state networks.[43,44] As the nonlinearity of the input transformation increases with V_{max} (Fig. 4, bottom right panel), information about past inputs that is retained within the reservoir is further perturbed. This makes it more difficult to faithfully recall past inputs, as illustrated by the decreasing MC (from left to right) in the panels in Fig. 5c. MC is a linear measure of memory and it is therefore expected that more nonlinear reservoirs have lower MC. However, the NARMA tasks require both memory *and* nonlinearity in order to emulate the dynamical systems described by Eq. (5) and (6). In this case, the trade-off favours memory over nonlinearity as noted in ref. 43, and consequently the task performance is

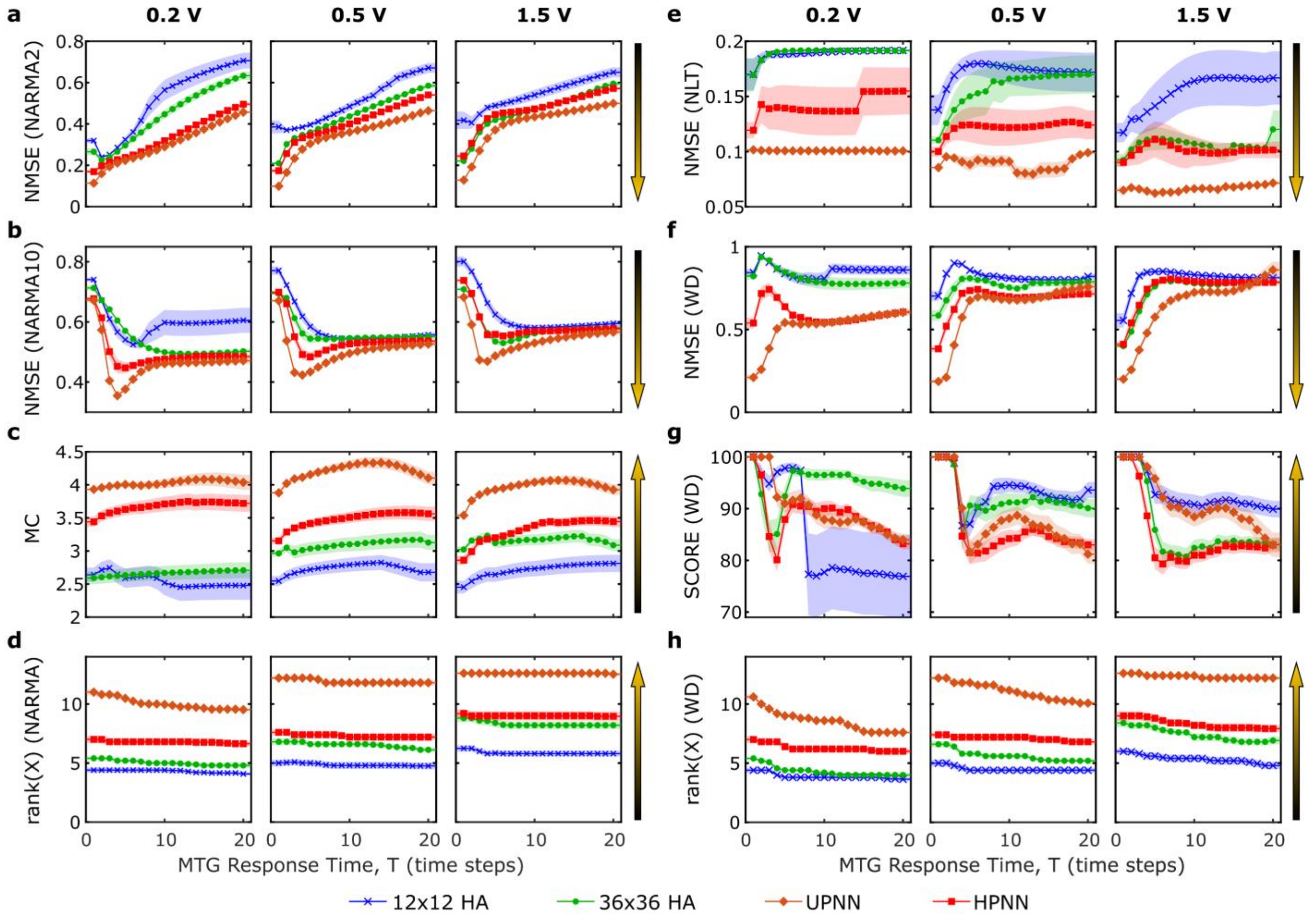

Fig. 6: Task performance for PNNs and Heterogeneous Arrays. For both (a-d) and (e-h), the three columns correspond to three different input amplitudes ($\mathrm{V_{max}} = 0.2$ V, 0.5 V, 1.5 V with $\mathrm{V_{min}} = 0.1$ V for all $\mathrm{V_{max}}$). The arrows indicate the direction of better performance for each panel. All results are the mean of five network realizations and five input sequences (25 trials total) except for NLT (e) which has only one input sequence. Shaded areas correspond to the standard error. All results are presented as a function of the network response time, T. (a) NARMA-2 and (b) NARMA-10 task performance expressed as the NMSE. (c) Linear memory capacity. (d) The rank of the predictor matrix X corresponding to (a-d). (e) NLT and (f) waveform discrimination task performance expressed as the NMSE. (g) Waveform discrimination task performance expressed as the % of correctly classified waveforms. (h) The rank of the predictor matrix X (during waveform discrimination), corresponding to the number of linearly independent outputs from the reservoir. The maximum possible rank is 13, corresponding to 12 electrode outputs and one constant bias term.

optimized by a lower input amplitude. Nonlinearity and memory are both required for WD, but the memory requirements are relatively small as compared with the NARMA tasks and so an intermediate $\mathrm{V_{max}}$ is optimal. In the NLT task, where nonlinearity requirements are dominant over memory requirements, the performance is optimized at larger $\mathrm{V_{max}}$. Thus, it is clear that the input amplitude which produces optimal performance is dependent on the particular task.

We now focus on a comparison of the performance of the HPNNs and UAs. The data presented in Fig. 5 directly compares the performance of the HPNNs with that of the 12×12 and 36×36 UAs: the UAs perform poorly at all five tasks, for all T. They achieve NMSE ~ 1 for both NARMA tasks, which means that they entirely fail to predict the required outputs (see Fig. S4 and Fig. S5, ESI†). Additionally, UAs exhibit MC ~ 1, meaning that the RC system cannot reproduce the delayed input signal ($\tau > 0$, as shown in Fig. S6, ESI†). The NMSE $\sim$0.2 for the NLT task is about twice as large as the NMSE obtained for the HPNNs and corresponds to the NMSE when a sine and square wave of equal amplitude and period are compared (see Fig. S7, ESI†). The UAs fail at the WD task with NMSE $\sim$1 and 0% of waveforms correctly classified.

The stark difference in performance between UAs and HPNNs is perhaps somewhat surprising given that the MTGs in both types of system have the same properties (same response time T (Eq. (2)) and nonlinearity (Eq. (1)). However, Fig. 5d and h show that the reservoir dynamics of the UAs lack the richness that is required for RC. The rank of the predictor matrix X indicates the number of linearly independent output currents from the reservoirs and reflects the (lack of) diversity of the outputs. The outputs of the 12×12 and 36×36 UAs both have $\mathrm{rank}(X) = 2$ for the NARMA, MC (Fig. 5d) and WD (Fig. 5h) tasks. As the bias (constant offset in $\hat{y}(k)$) is included in X, this indicates that all of the output currents are collinear. By comparison, HPNNs produce reservoir outputs with mean $\mathrm{rank}(X)$ values as high as 9, indicating significantly more diversity among the output current dynamics. This diversity can be seen in Fig. S2f (ESI†), where the different electrode currents exhibit different levels of nonlinearity and hysteresis. Because HPNNs have both asymmetric (scale-free) connectivity *and* heterogeneous initial MTG sizes, there is a wide distribution of the local potential differences across the MTGs at any given time, which produce a broad range of responses from the individual MTGs. The resulting reservoir dynamics become more diverse as the range of applied voltages (i.e. $\mathrm{V}_{\mathrm{max}}$) increases (e.g. Fig. S2f, ESI†), thus explaining the observed increase in $\mathrm{rank}(X)$ for the HPNNs in Fig. 5d and h.

The diversity of the outputs is related to the effective dimensionality of the nonlinear transformation of the input signals. When an input signal is transformed into an N-dimensional reservoir output, this transformation is only truly N-dimensional when each of the N reservoir outputs are linearly independent of the others. Collinear outputs do not improve the transformed representation of the input. Furthermore, training may be impeded by redundant predictors as the linear regression problem becomes increasingly ill-conditioned the more collinearity exists among predictor variables.

PNNs and Heterogeneous MTG Arrays

It is clear from the results presented so far that the HPNNs have greater information processing capability than the UAs. However, it is unclear whether it is the HPNN's scale-free topology, or the heterogeneity of the MTG sizes (or a combination of the two) which bestow this computational advantage. To investigate these possibilities, we inverted the relationship between topology and MTG heterogeneity to create two variants of the systems studied so far. Arrays were assigned heterogeneous initial MTG sizes (HAs, see above) and PNNs were forced to have uniform initial MTG sizes $D_{\mathrm{uniform}} = 0.05$ particle diameters (UPNNs, note that the natural scale-free connectivity structure was not perturbed). The same set of tasks, using the same voltage ranges and characteristic time scales, were repeated using the HAs and UPNNs.

Fig. 6 shows the task performance for the HAs (12×12, blue; 36×36, green) and UPNNs (amber) using the same layout as Fig. 5: NARMA-2 (Fig. 6a), NARMA-10 (Fig. 6b), MC (Fig. 6c) NLT (Fig. 6e), WD (Fig. 6f and g). Fig. 6d and h show the rank of the predictor matrix X corresponding to the NARMA and WD tasks respectively. The performance of the HPNNs shown in Fig. 5 is reproduced in Fig. 6 for reference (red); note that the y-axis scales in Fig. 6 have been reduced relative to Fig. 5 to allow visualisation of more detail. S3 (ESI†) contains examples of target function $y(k)$ overlaid with the final output $\hat{y}(k)$ for each type of network and for each task.

It is immediately obvious in Fig. 6 that the performance of the arrays improves significantly when endowed with heterogeneous MTG sizes (compare purple (light green) in Fig. 5 with blue (green) in Fig. 6). The performance of the HAs in each task shows a strong dependence on T just like the HPNNs, albeit with slightly different optimal T values in some cases. The optimal values of T and $\mathrm{V}_{\mathrm{max}}$ depend on the task, as for the HPNNs discussed above. These results indicate that the heterogeneity of MTGs plays a crucial role in breaking the symmetry of the UAs, leading to richer reservoir dynamics and more diverse outputs, and consequential improved task performance. This effect is illustrated by the diversity among the electrode outputs in Fig. S2d and e (ESI†) which is not present for the UAs in Fig. S2a and b. This increase in output diversity is also reflected in the larger $\mathrm{rank}(X)$ for the HAs than for the UAs (compare Fig. 6d and h with Fig. 5d and h). There is clearly a direct relationship between output diversity, as represented by $\mathrm{rank}(X)$, and the corresponding task performance in Fig. 6.

The 36×36 HAs generally outperform the 12×12 HAs and it is clear that this is due to their greater output diversity (larger $\mathrm{rank}(X)$). This indicates that some network properties play a greater role in producing diverse output signals (and greater computational performance) than others. For example, since the number of MTGs (N_{MTG}) in the 36x36 HAs is similar in that of the PNNs, while the mean path length ($\langle L_P \rangle$) of the 12×12 HAs is similar to the PNNs, this suggests that N_{MTG} is more computationally relevant than $\langle L_P \rangle$ (see network modelling section and S1, ESI†). Making a similar consideration of network topology, both $\mathrm{rank}(X)$ and task performance for the 36×36 HAs is very similar to those of the HPNNs at $\mathrm{V}_{\mathrm{max}} = 1.5$ V. This would seem to suggest that, when MTG sizes are heterogeneous, the computational performance of memristive electronic reservoirs is not strongly affected by reservoir topology.

The roles of MTG heterogeneity and reservoir topology are further elucidated by results for the UPNN, shown in Fig. 6. Perhaps counterintuitively, forcing the PNNs to have uniform MTG sizes results in an *improvement* in performance over those with heterogenous MTG sizes, opposite to the trend observed for the arrays. This indicates that there is interplay between the roles of MTG heterogeneity and reservoir topology, which is now discussed. In the case of UAs, the structure is regular and the MTG sizes are uniform, resulting in spatially symmetrical and temporally synchronized local potentials and tunnelling currents within the reservoir, leading to low diversity among the reservoir outputs. By either adding heterogeneity to the MTG sizes (HAs) or by replacing the regular connectivity with a scale-free topology (UPNNs), that symmetry is broken, resulting in widely distributed local potentials and tunnelling currents throughout the reservoir. This variation contributes richness to the reservoir dynamics and leads to diverse reservoir outputs, thus facilitating greater computational performance. In S4 (ESI†) we show that the symmetry of UA dynamics can also be broken by randomly weighting the inputs (so that each of the left-hand electrodes receives a different signal amplitude), or by applying the input to a single left-hand electrode. Fig. S10 (ESI†) shows task performance for the 12×12 and 36×36 UAs using both randomly weighted inputs to all input electrodes, and using a single input electrode. The performance of the 12×12 UAs with these input configurations lies between the HPNN and UPNN performance for all five tasks, and the performance of the 36× 36 UA becomes similar to that of the HPNNs. Hence approximately the same level of performance gain can be attained using either a scale-free topology or input configurations that break the symmetry of the uniform arrays.

The better performance of UPNNs compared to HPNNs can be explained by the presence of "choke-points" in the HPNNs. The scale-free structure means that there are few current paths through the network in parallel with some key MTGs; if those MTGs happen to have large sizes (and therefore high resistance), the output currents measured at electrodes beyond that point in the network may then be relatively small and well-synchronized, resulting in reduced richness. In the UPNNs, choke points do not exist because all of the MTGs have identical initial sizes. This fact is reflected by a wider range of results (denoted by larger error bands in Fig. 6) for the HPNNs as compared with the UPNNs. We expect that the difference in performance between HPNNs and UPNNs will be significantly reduced in much larger PNNs, such as those produced experimentally, but further studies are needed to confirm that hypothesis. It is worth noting that nanowire networks have irregular topologies that are small-world but not scale-free.[45] They also have a high degree of MTG uniformity due to the controlled thickness of the insulating coatings surrounding the nanowires (which forms the memristive junction). Thus, choke points are less likely and it would be interesting to compare RC performance of NWNs to that of the UPNNs.

Conclusions

We have demonstrated the effects of both network topology and memristor heterogeneity on computational capability by comparing the performance of scale-free and regular arrays of memristors, and have considered both uniform and heterogeneous memristor properties. The results from a range of benchmark tasks lead us to the following conclusions: (i) In the presence of rich (or at least non-trivial) dynamics, the optimal response time scale and input amplitudes depend on the memory and nonlinearity requirements for each particular task, due to the so-called "memory-nonlinearity trade-off". (ii) The best RC performance is achieved using scale-free networks of uniform memristive elements, while the worst performance is observed from uniform arrays. (iii) The performance of the arrays improves significantly when the memristors are heterogeneous, and approaches that of the HPNNs. (iv) HPNN performance is limited (compared to UPNNs) by "choke points" which restrict information flow. (iv) The most important factor contributing to computational performance is the diversity of the reservoir outputs. (v) Scale-free topologies result in rich dynamics and diverse outputs even for uniform MTGs and symmetrical input configurations, whereas arrays can only produce diverse outputs if their symmetry is broken (by heterogeneity in the memristors, or by restricting the number of input nodes).

Methods

Reservoir Training

Training is performed by calculating $N + 1$ linear coefficients w_i, such that the final output $\hat{y}(k)$ approximates the desired target output $y(k)$:

$$\hat{y}(k) = \sum_{i=1}^{N+1} w_i\, x_i(k) \approx y(k). \tag{3}$$

The vector of linear coefficients w is found using linear regression via the Moore-Penrose pseudo-inverse †:

$$w = (X^{\dagger} y). \tag{4}$$

Reservoir Computing Tasks

Nonlinear auto-regressive moving average (NARMA) tasks[46] are a set of challenging time series prediction benchmarks commonly used to quantify the computational capabilities of brain-like information processing systems. The aim of these tasks is to emulate a particular nonlinear dynamical system by learning the association between a discrete white noise signal $u(k)$ and the resulting chaotic time series produced by that dynamical system when driven by $u(k)$. Here we consider two NARMA tasks: NARMA-2 and NARMA-10.

The NARMA-2 time series is produced by the dynamical system:

$$y_{k+1} = \alpha y_k + \beta y_k y_{k-1} + \gamma u_k^3 + \delta, \tag{5}$$

where the constants $[\alpha,\ \beta,\ \gamma,\ \delta] =$ [0.4, 0.4, 0.6, 0.1].[46] Each point y_{k+1}depends nonlinearly on both the present input u_k and the previous *two* output values y_k and y_{k-1}; hence the name NARMA-2. The NARMA10 time series is produced by the dynamical system:

$$y_{k+1} = \alpha y_k + \beta y_k \left[\sum_{i=0}^{9} y_{k-i}\right] + \gamma u_k u_{k-9} + \delta, \tag{6}$$

where the constants $[\alpha,\ \beta,\ \gamma,\ \delta] =$ [0.3, 0.05, 1.5, 0.1].[46] The NARMA-10 task is significantly more challenging than the NARMA-2 task because each point y_{k+1} now depends on the *ten* previous values of both $u(k)$ and $y(k)$. For both NARMA tasks we use input signals $u(k)$ in the range [0, 0.5] with length $K = 4000$, where the first half of the sequence is used for training and the second half is used to test the trained RC system.

The performance of the RC systems on the NARMA tasks is quantified by the error between the RC system output $\hat{y}(k)$ and the target output $y(k)$, with smaller error corresponding to better performance. Several error metrics can be used, among which there is little consistency in the literature, often making direct comparison of results difficult. We use the normalized mean-square error (NMSE), where the sum of squared residuals is normalized by the variance of the target function $\sigma^2(y_k)$:

$$NMSE = \frac{1}{K}\frac{\sum_{k=1}^{K}(\hat{y}_k - y_k)^2}{\sigma^2(y_k)}. \tag{7}$$

The third task that we consider is the short-term memory capacity (MC) task.[47] The MC task measures the ability of an RC system to faithfully reproduce delayed copies of a white noise input signal $u(k)$. The memory capacity for a delay τ is given by:

$$MC_\tau = \frac{cov(u(k-\tau), \hat{y}_\tau)^2}{\sigma^2(u(k-\tau))\sigma^2(\hat{y}_\tau)}, \tag{8}$$

where σ^2 and cov represent the variance and covariance respectively and $\hat{y}_\tau$ is the reservoir output trained to approximate the delayed input. The total memory capacity of the RC system is then the sum of MC_τ over all possible delays:

$$MC = \sum_{\tau=0}^{\infty} MC_\tau. \tag{9}$$

In practice we limit the sum in Eq. (9) to $\tau = 100$, which is more than sufficient for the reservoirs considered here.

The fourth task considered is waveform discrimination (WD)[48] where randomized sequences of complete sine and square waveforms are to be classified as either 'sine' or 'square' by outputting a different binary value (one and minus one respectively) for each waveform. As we consider input waveforms of period $P = 8$, the resulting binary target function $y(k)$ (and reservoir output $\hat{y}(k)$) has eight values for each input waveform.[42] In order to make a classification, the mean of these eight output values $\langle \hat{y}(k) \rangle_{\mathrm{wf}}$ is compared with the decision boundary at $y = 0$, i.e. positive (negative) $\langle \hat{y}(k) \rangle_{\mathrm{wf}}$ indicates a 'sine' ('square') classification. The performance can be measured both by the NMSE between $\hat{y}(k)$ and $y(k)$ (Eq. (7)) and by the percentage of correctly classified waveforms. Compared with the NARMA and MC tasks which require strong memory from the reservoir, WD requires less memory and more nonlinearity.

The final task that we consider is nonlinear transformation (NLT)[49] in which an input sine wave of period $P = 100$ must be transformed (nonlinearly) into a square wave of the same period. The performance is measured using the NMSE ((Eq. (7)). Compared with the other tasks, NLT utilizes a high degree of nonlinearity without requiring much memory.

Conflicts of interest

There are no conflicts to declare.

Acknowledgements

This project was financially supported by The MacDiarmid Institute for Advanced Materials and Nanotechnology and the Marsden Fund.

References

1 D. Marković, A. Mizrahi, D. Querlioz and J. Grollier, *Nat. Rev. Phys.*, 2020, **2**, 499–510.

2 P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, B. Brezzo, I. Vo, S. K. Esser, R. Appuswamy, B. Taba, A. Amir, M. D. Flickner, W. P. Risk, R. Manohar and D. S. Modha, *Science*, 2014, **345**, 668–673.

3 M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain, Y. Liao, C.-K. Lin, A. Lines, R. Liu, D. Mathaikutty, S. McCoy, A. Paul, J. Tse, G. Venkataramanan, Y.-H. Weng, A. Wild, Y. Yang and H. Wang, *IEEE Micro*, 2018, **38**, 82–99.

4 S. Furber, *J. Neural Eng.*, 2016, **13**, 051001.

5 J. Torrejon, M. Riou, F. A. Araujo, S. Tsunegi, G. Khalsa, D. Querlioz, P. Bortolotti, V. Cros, K. Yakushiji, A. Fukushima, H. Kubota, S. Yuasa, M. D. Stiles and J. Grollier, *Nature*, 2017, **547**, 428–431.

6 Y. Li, Z. Wang, R. Midya, Q. Xia and J. J. Yang, *J. Phys. D. Appl. Phys.*, 2018, **51**, 503002.

7 Z. Wang, S. Joshi, S. E. Savel'ev, H. Jiang, R. Midya, P. Lin, M. Hu, N. Ge, J. P. Strachan, Z. Li, Q. Wu, M. Barnell, G. L. Li, H. L. Xin, R. S. Williams, Q. Xia and J. J. Yang, *Nat. Mater.*, 2017, **16**, 101–108.

8 M. Dale, J. F. Miller, S. Stepney and M. A. Trefzer, in *Unconventional Computation and Natural Computation. UCNC 2016. Lecture Notes in Computer Science*, eds. M. Amos and A. CONDON, Springer, Cham., 2016, pp. 49–61.

9 M. Akai-Kasaya, Y. Takeshima, S. Kan, K. Nakajima, T. Oya and T. Asai, *Neuromorphic Comput. Eng.*, 2022, **2**, 014003.

10 R. K. Daniels, J. B. Mallinson, Z. E. Heywood, P. J. Bones, M. D. Arnold and S. A. Brown, *Neural Networks*, 2022, **154**, 122–130.

11 J. Hochstetter, R. Zhu, A. Loeffler, A. Diaz-Alvarez, T. Nakayama and Z. Kuncic, *Nat. Commun.*, 2021, **12**, 4008.

12 H. O. Sillin, R. Aguilera, H. H. Shieh, A. V. Avizienis, M. Aono, A. Z. Stieg and J. K. Gimzewski, *Nanotechnology*, 2013, **24**, 384004.

13 G. Milano, G. Pedretti, K. Montano, S. Ricci, S. Hashemkhani, L. Boarino, D. Ielmini and C. Ricciardi, *Nat. Mater.*, 2022, **21**, 195–202.

14 H. Tanaka, S. Azhari, Y. Usami, D. Banerjee, T. Kotooka, O. Srikimkaew, T.-T. Dang, S. Murazoe, R. Oyabu, K. Kimizuka and M. Hakoshima, *Neuromorphic Comput. Eng.*, 2022, **2**, 022002.

15 S. K. Bose, J. B. Mallinson, R. M. Gazoni and S. A. Brown, *IEEE Trans. Electron Devices*, 2017, **64**, 5194–5201.

16 J. B. Mallinson, S. Shirai, S. K. Acharya, S. K. Bose, E. Galli and S. A. Brown, *Sci. Adv.*, 2019, **5**, eaaw8438.

17 C. Minnai, A. Bellacicca, S. A. Brown and P. Milani, *Sci. Rep.*, 2017, **7**, 7955.

18 N. Carstens, B. Adejube, T. Strunskus, F. Faupel, S. Brown and A. Vahl, *Nanoscale Adv.*, 2022, **4**, 3149–3160.

19 H. Jaeger, *The " echo state " approach to analysing and training recurrent neural networks – with an Erratum note.*, 2001.

20 W. Maass, T. Natschläger and H. Markram, *Neural Comput.*, 2002, **14**, 2531–2560.

21 M. Lukoševičius and H. Jaeger, *Comput. Sci. Rev.*, 2009, **3**, 127–149.

22 G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano and A. Hirose, *Neural Networks*, 2019, **115**, 100–123.

23 K. Nakajima, *Jpn. J. Appl. Phys.*, 2020, **59**, 060501.

24 M. D. Pike, S. K. Bose, J. B. Mallinson, S. K. Acharya, S. Shirai, E. Galli, S. J. Weddell, P. J. Bones, M. D. Arnold and S. A. Brown, *Nano Lett.*, 2020, **20**, 3935–3942.

25 S. Shirai, S. K. Acharya, S. K. Bose, J. B. Mallinson, E. Galli, M. D. Pike, M. D. Arnold and S. A. Brown, *Netw. Neurosci.*, 2020, **4**, 432–447.

26 Z. Heywood, J. Mallinson, E. Galli, S. Acharya, S. Bose, M. Arnold, P. Bones and S. Brown, *Neuromorphic Comput. Eng.*, 2022, **2**, 024009.

27 E. Bullmore and O. Sporns, *Nat. Rev. Neurosci.*, 2009, **10**, 186–198.

28 E. Bullmore and O. Sporns, *Nat. Rev. Neurosci.*, 2012, **13**, 336–349.

29 H.-J. Park and K. Friston, *Science*, 2013, **342**, 1238411–1238411.

30 Z. Deng and Y. Zhang, *IEEE Trans. Neural Networks*, 2007, **18**, 1364–1375.

31 Y. Kawai, J. Park and M. Asada, *Neural Networks*, 2019, **112**, 15–23.

32 M. Dale, S. O'Keefe, A. Sebald, S. Stepney and M. A. Trefzer, *Nat. Comput.*, 2021, **20**, 205–216.

33 M. Dale, J. Dewhirst, S. O'Keefe, A. Sebald, S. Stepney and M. A. Trefzer, in *Unconventional Computation and Natural Computation. UCNC 2019. Lecture Notes in Computer Science*, ed. S. McQuillan, I., Seki, Springer, Cham, 2019, vol. 11493, pp. 52–

64.

34 S. Bose, S. Shirai, J. Mallinson, S. Acharya, E. Galli and S. Brown, in *2018 IEEE 18th International Conference on Nanotechnology (IEEE-NANO)*, IEEE, 2018, pp. 1–2.

35 S. Fostner, R. Brown, J. Carr and S. A. Brown, *Phys. Rev. B*, 2014, **89**, 075402.

36 S. Fostner and S. A. Brown, *Phys. Rev. E*, 2015, **92**, 052134.

37 A. Sattar, S. Fostner and S. A. Brown, *Phys. Rev. Lett.*, 2013, **111**, 136808.

38 A. D. F. Dunbar, J. G. Partridge, M. Schulze and S. A. Brown, *Eur. Phys. J. D*, 2006, **39**, 415–422.

39 D. Stauffer and A. Aharony, *Introduction To Percolation Theory*, Taylor & Francis, 2018.

40 D. B. Strukov, G. S. Snider, D. R. Stewart and R. S. Williams, *Nature*, 2008, **453**, 80–83.

41 W. Wang, M. Wang, E. Ambrosi, A. Bricalli, M. Laudato, Z. Sun, X. Chen and D. Ielmini, *Nat. Commun.*, 2019, **10**, 81.

42 M. Riou, J. Torrejon, B. Garitaine, F. Abreu Araujo, P. Bortolotti, V. Cros, S. Tsunegi, K. Yakushiji, A. Fukushima, H. Kubota, S. Yuasa, D. Querlioz, M. D. Stiles and J. Grollier, *Phys. Rev. Appl.*, 2019, **12**, 024049.

43 D. Verstraeten, J. Dambre, X. Dutoit and B. Schrauwen, in *The 2010 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2010, pp. 1–8.

44 M. Inubushi and K. Yoshimura, *Sci. Rep.*, 2017, **7**, 10199.

45 R. K. Daniels and S. A. Brown, *Nanoscale Horizons*, 2021, **6**, 482–488.

46 A. F. Atiya and A. G. Parlos, *IEEE Trans. Neural Networks*, 2000, **11**, 697–709.

47 H. Jaeger, *Short Term Memory in Echo State Networks*, GMD Forschungszentrum Informationstechnik, Sankt Augustin, 2001.

48 Y. Paquot, F. Duport, A. Smerieri, J. Dambre, B. Schrauwen, M. Haelterman and S. Massar, *Sci. Rep.*, 2012, **2**, 287.

49 E. C. Demis, R. Aguilera, K. Scharnhorst, M. Aono, A. Z. Stieg and J. K. Gimzewski, *Jpn. J. Appl. Phys.*, 2016, **55**, 1102B2.