Tutorial: Reservoir Computing

D. Brunner, M. Jacquot, L. Larger
FEMTO-ST / Optics Dept., CNRS Univ. Bourgogne-Franche-Comté, Besançon, France

Spintronics meets Neuromorphics, Spice, Mainz, Germany
October 11th, 2018
What is the problem: I. Classification

Sci-Hub: Where is Einstein?

Postman: Cat /Dog?

Recurrent Neural Network (RNN)

Deep Neural Network (DNN)
Algorithmic calculation:

\[ y_{n+1} = \frac{y_n + \delta(0.2y_t \cdot \frac{1 + y_t^{10}}{1 + y_t} - 0.1y_t)}{n=110000 \Delta t<0.1 \text{s}} \]

Complex, ambiguous behavior:

What is the problem: II. Prediction
Reservoirs: Dynamical, Complex, Simple

- **Input**: Random injection
- **Reservoir**: Random connection
  - Resting state
- **Output**: Linear weights
  - Linear regression

- Simple: 2 random matrix multiplications
- Echo state property: working memory
- Excellent performance for prediction

Operating a reservoir

Input: $u(n + 1)$

Network state:

\[ x(n + 1) = f[Wx(n) + W^{\text{inj}}u(n + 1) + b] \]

Output: $y(n + 1) = W^{\text{out}}x(n + 1)$

1. Training data: set of inputs $u(n)$ for which $y(n)$ is known
2. Collect $x(n)$ for $n \in [1, ..., T]$
3. Cross-validate: randomly label instances of input data by $l \in [1, ..., L]$
4. Select one $l_c \in [1, ..., L]$
   - $M_x$: concatenated matrix of $x(n)$ in response to $u(n)$ for $l \neq l_c$
   - $T^T$: concatenated matrix of $y(n)$ for $l \neq l_c$
   - Obtain $W^{\text{out}} = (M_xM_x^T + \lambda I)^{-1}(M_xT^T)$
5. Measure error for $l_c$

For hardware: Don’t care about system ($x(n + 1) = f[x(n)]$)
- Good control of “distance” to criticality
- Good access to system state
Why novel hardware???
Big questions: What hardware

Neural Networks: What is special ????

- Large scale parallel dynamics
- Highly connected
- Direct, physical links (no address routing)
  - Fully parallel
  - Stand alone / unclocked
  - Ideally energy efficient (no routing overhead)

Even TPU: ~200 KHz global frame rate

\[
X \quad \text{with } N^2 = \sum_{j=1}^{N} \sum_{i=1}^{n+1} n \cdot i + j \cdot n 
\]
Photonic RC: delay networks

Spatially multiplexed reservoir


- Multiplexed in space:
  - Injection weights constant in time
  - Data-rate: bandwidth of nodes
- Hardware implementation challenging
- Flexible network-connectivity structure

Temporally multiplexed reservoir


- Multiplexed in time:
  - Injection weights based on temporal modulation
  - Data-rate: bandwidth / N
- Simplistic hardware implementation
- Potentially large memory
Neural Networks in Photonic Delay Systems
Delay systems 101

\[ \varepsilon \dot{x}(s) + x(s) = f(\beta x(s - 1), b) \]
\[ s = t/\tau, \ \varepsilon = T_R/\tau \]

- No feedback
- Driven response

- Feedback
- No drive

- Highly simplistic system
- High dimensional
- Critical transition

Soriano et al., RMP 85, 421 (2013).
Delay systems are ring networks

\[ \varepsilon \dot{x}(s) + x(s) = f(x(s - 1)) \]

Solving via Green’s function approach

\[ x(s) = \int_{-\infty}^{s} h(s - \xi) \cdot f(x(\xi - 1)) d\xi, \quad h(s): \text{Impulse response} \]

Temporal reorganization

\[ x_\sigma(n) = x(s), \sigma \in [0, 1 + \gamma], (1 + \gamma)n + \sigma = s, n = 0, 1, 2, \ldots \]

\[ n = \text{time}, \sigma = \text{node} \]

\[ x_\sigma(n) \approx \int_{\sigma-1}^{\sigma} h(\sigma - \xi) \cdot f[x(\xi - 1)] d\xi \]

Coupling = convolution with previous NL-transformed states

Delay reservoirs:

Input: \( W^{\text{inj}} u(n + 1) \), \( W^{\text{inj}} \) is temporal modulation

Network state: \( \varepsilon \dot{x}(s) + x(s) = f(x(s - 1), u(s)) \)

Output: \( W^{\text{out}} x(n + 1) \), \( W^{\text{out}} \) is temporal modulation
Tour de ville for delay substrates

**Electronic:**
- Mackey-Glass nonlinearity

**Opto-electronic:**
- Ikeda nonlinearity

**Nano-electronics:**
- Spin-torque oscillator

**All optical:**
- Semiconductor devices

Neural Networks in spatio-temporal photonic systems
Why Optics? Connections!

- Parallel
- 2D substrate not sufficient
- Negligible distance tradeoff
- Zero induction
Photonic neural network

Neurons: liquid crystal pixels

Network: holographic fan-out

- Parallel state, analogue
- Single element

- DOE > 90,000 nodes
- SLM > 480,000 nodes

Polarization filtering
Recurrent coupling:

\[ y_{n+1}^{i} \propto \sum_{i=1}^{N} W_{i,k}^{DOE} (E_{i}^{0} - E_{i}^{n+1})^{2} \]

**Electro-optical networks**

- **Nodes:** Pixels of SLM
  - Density: 6400 / mm\(^2\)
- **Coupling:** DOE + imaging
- **4f architecture:**
  - Self-coupling
- **2025 network nodes**
- **Dynamic evolution:** iterative update

Camera -> SLM

**Readout weights:**

- **Digital Micro-mirror Array**
- **Image SLM onto DMD**
- **Result:** analog power meter

**BUT**

- Full state not known
- State detection non-linear
- Matrix inversion not possible

Matlab:

\[ I_{i}^{n+1} = \sin^{2}(\beta) \sum_{j=1}^{N} \kappa_{i,j}^{DOE} E_{j}^{n} \kappa_{i}^{inj} u^{n+1} + \Theta_{0} \]
Coupling matrix

Diffractive networks: coupling a neighborhood

- For 45x45 nodes: 100 KFIOP / per state
- For 300x300 nodes: 4.5 MFLOP / per state
- More complex / larger range trivial to create

Bueno, et al., under review *Optica.*
Autonomous dynamics

\[ I_{i}^{n+1} = \sin^2(\beta I_{j}^{n} + \Theta_0) \]

\[ I_{i}^{n+1} = \sin^2(\beta(\sum_{j=1}^{N} \kappa_{i,j}^{DOE} E_{j}^{n})^2 + \Theta_0) \]
1. Training of Boolean readout

1. Select mirror with bias to “not yet modified”
2. Flip mirror (1 → 0, 0 → 1)
3. Record output for 200 steps
4. Compare current with previous NMSE

- Error reduces efficiently
- Close to state of the art (remember: Boolean weights only)

Bueno, et al., under review *Optica.*
No negative values (uni-polarity)

1. SLM: plane wave illumination
2. Polarization filtering: only constructive interaction
   - Only positive addition possible
3. Diffractive coupling: only constructive interaction
   - Only positive addition possible
4. Boolean readout weights: 0 / 1
   - Only positive addition possible

+ No phase effects: stable
- Connection weights always positive!
A way around unipolar weights

$$I_i^{n+1} = \sin^2(\beta| \sum_{j=1}^{N} \kappa_i^{DOE} E_j^n|^2 + \gamma \kappa_i^{inj} u^{n+1} + \Theta_0)$$

$\Theta_0$ is a Matrix:
- $(1 - \mu)$ – Values: $\Theta_0 = 42$
- $(\mu)$ – Values: $\Theta_0 = 109$

- Problem with unipolarity: no negative ‘slopes’
- Solution: harvest periodic nonlinearity

No negative weights

- Distribute operation points strongly aids performance
- Best points: close to 50/50 division

Bueno, et al., under review *Optica*
1. Training of Boolean readout

1. Clear orientation toward “one-step leader”
2. Divergence largely close to local extrema

Bueno, et al., under review *Optica*
2. Feedback of RNN output

- **Prediction target: future state**
  - Output approximates future input
- **Output feedback: self consistent, autonomous system**
- **Morphing into target system (Neuronal Network → Butterfly effect)**
- **Important: principle of motor control**

Image: © http://testoil.com/did-you-know/the-butterfly-effect/
2. Feedback of readout result

- RNN creates autonomous, nonlinear oscillator
- Period very close to target
2. Feedback of RNN output

- Output reassemble MG attractor
- Readout weights fully passive
- Readout weights have no bandwidth limitation
How do Neural Networks predict (chaos)
What is chaos?

Chaotic systems are sensitive to initial conditions: point in phase space not sufficient

This sensitivity creates unpredictability: impossibility to determine future development for all times.

State space chaotic systems

Takens embedding theorem

**Theorem 1.** The time delayed version of one time series suffices to reveal the structure of an attractor. Let us represent the data in $M$-dimensional space by the vectors $\mathbf{x} = [y(t), y(t - \tau_0), \ldots, y(t - (M - 1)\tau_0)]^\dagger$. Where $(\cdot)^\dagger$ as transpose matrix. The pair dimension-delay for the embedding $(M, \tau_0)$ contributes to reconstruct the right object in the state space.

Temporal position of delay vectors:
- $\min |(AC(x(t)))|
- \min (MI(x(t), x(t - n)))$

Attractor reconstruction in rRNNs

Cross-correlation analysis for Mackey-Glass for $\mu=1.3$

According to Cross-correlation analysis, RNN does:

1. Takes-like embedding
2. Increases sampling
Characterize nearest neighbors: Random Projections theory

**Proposition**  For any positive constant values $\epsilon_1, \epsilon_2$. Let $V$ be a collection of $S$ points $\{y^{(1)}, y^{(2)}, \ldots, y^{(S)}\} \in \mathbb{R}^q$, with distances computed under $L_2$ norm. There is a map $\varphi : \mathbb{R}^q \to \mathbb{R}^h$, such that for all $y^{(i)}, y^{(j)} \in V$,

$$(1 - \epsilon_1)\|y^{(i)} - y^{(j)}\| \leq \|\varphi(y^{(i)}) - \varphi(y^{(j)})\| \leq (1 + \epsilon_2)\|y^{(i)} - y^{(j)}\|.$$  

This proposition states that the **distances between two consecutive states** of the attractor are bound to the range 

$$[(1 - \epsilon_1), (1 + \epsilon_2)]$$

where $\epsilon_1, \epsilon_2$ are arbitrary constant values.

Example of a distribution of nearest neighbor states in rRNN
NN limits according to Random Projection:

I. Sampling dense but short-range

II. Sampling dense and long range

III. Sampling not dense but long-range

Limits of nearest neighbour-distances bind good prediction conditions
NN limits according to Random Projection:
New scheme: Taken RNN

Introduce first-in first-out memory specifically with depth of Taken embedding delay
Operation at BAD parameters

Cross-correlation analysis for Mackey-Glass for $\mu=0.2$

rRNN

TrRNN
Applied to model for cardiac arrest

Interspike intervals (ISI) of an arrhythmic excitable system comparable to a heart

Comparison between the stabilized mean of the TrRNN and the classical rRNN

Stabilization of the system based on our TrRNN

The TrRNN requires 15 times less nodes, simultaneously achieving superior performance.


International conference on:
Cognitive Computing: Merging Concepts and Hardware
http://www.cognitive-comp.org/
18th - 20th of December 2018
at Herrenhausen Castle, Hannover, Germany

Topical Sessions:

- Cognitive neurosciences: how they may guide novel computing technologies
- Cognitive applications of current systems
- Theoretical concepts and mathematical foundations
- Towards neuronal hardware networks
- Novel substrates

Organizing Committee:
- Dr. Daniel Brunner
- Prof. Herbert Jaeger
- Prof. Gordon Pipa
- Prof. Stuart Parkin

• Kwabena Boahen
• Joanna J. Bryson
• Chris Eliasmith
• Edward A. Lee
• Demetri Psaltis
• Pieter Roelfsema
• Susan Stepney
• Ipke Wachsmuth
• David Wolpert
Summary

- 2025 EO network nodes, much larger to be expected
- Learning / analog, passive readout fully implemented
- Stability (?), noise (?)
- Approaching understanding of prediction in ANN

http://neuroqnet.com/