Quantum reservoir computing in finite dimensions

Physical Review E 107 (3), 035306

Rodrigo Martínez-Peña (IFISC)
Juan-Pablo Ortega (NTU, Singapore)

PhD supervisors:
Roberta Zambrini and Miguel C. Soriano

June 1st 2023

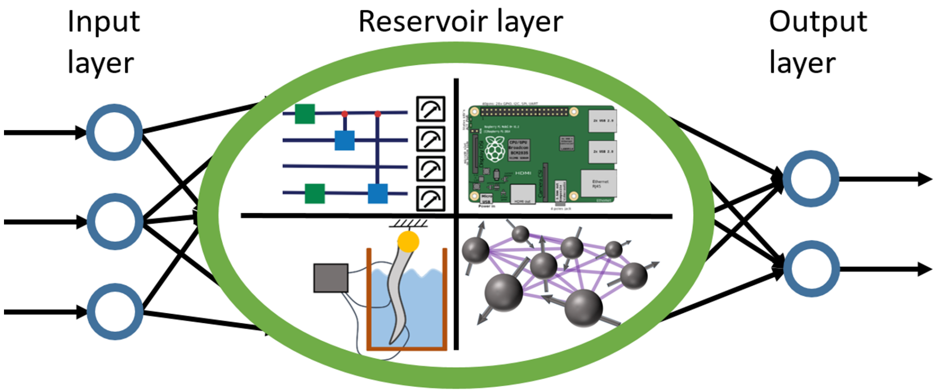
QuaReC
PRD2018/47

EXCELENCIA MARÍA DE MAEZTU
MDM-2017-0711

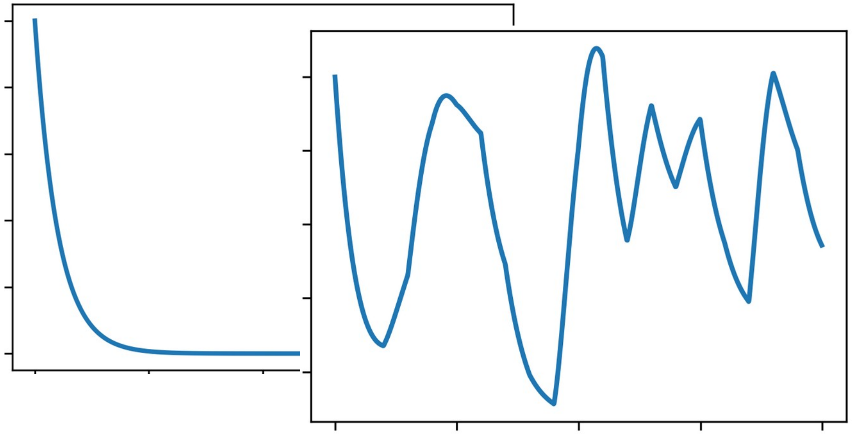
QuaResC
PID2019-109094GB-C21

- Introduction

- (Quantum) Reservoir Computing



- Motivation of this work



- Result 1
- Result 2 and examples
- Conclusions and outlook

Machine Learning is mainly developed for **digital computers** due to its **accessibility**:

- Ease of use
- Unified theory
- Universal (can emulate Turing machines)

Machine Learning is mainly developed for **digital computers** due to its **accessibility**:

- Ease of use
- Unified theory
- Universal (can emulate Turing machines)

But it has some problems:

- Energy footprint 10% and growing (also in deep learning)

Andrae, A. S., & Edler, T. *Challenges*, 2015

N. C. Thompson *et. al.* *ArXiv*:2007.05558, 2020

Machine Learning is mainly developed for **digital computers** due to its **accessibility**:

- Ease of use
- Unified theory
- Universal (can emulate Turing machines)

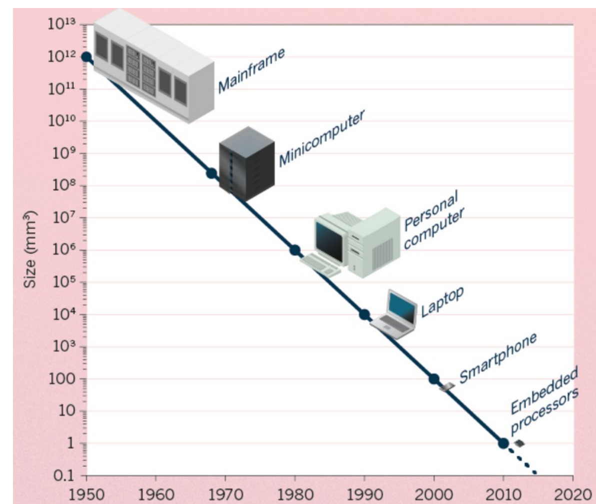
But it has some problems:

- Energy footprint 10% and growing (also in deep learning)

Andrae, A. S., & Edler, T. *Challenges*, 2015

N. C. Thompson *et. al.* *ArXiv*:2007.05558, 2020

- Miniaturization (end’s of Moore’s law)



Mitchell Waldrop *Nature* 2016

Machine Learning is mainly developed for **digital computers** due to its **accessibility**:

- Ease of use
- Unified theory
- Universal (can emulate Turing machines)

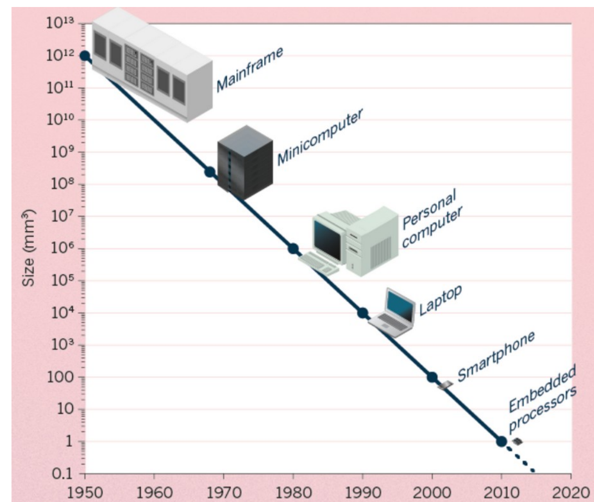
But it has some problems:

- Energy footprint 10% and growing (also in deep learning)

Andrae, A. S., & Edler, T. *Challenges*, 2015

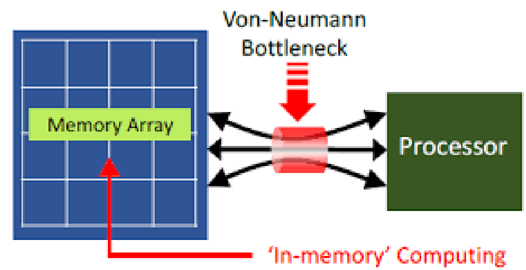
N. C. Thompson *et. al. ArXiv*:2007.05558, 2020

- Miniaturization (end's of Moore's law)



Mitchell Waldrop *Nature* 2016

- Von Neumann Bottleneck



G. Indiveri *et. al. Proceedings of the IEEE*, 2015

Tom Dillinger

- Possible alternative in **Neuromorphic Computing:** brain-inspired computing paradigm. **Computation and physical substrate go hand by hand**

- Possible alternative in **Neuromorphic Computing:** brain-inspired computing paradigm. **Computation and physical substrate go hand by hand**

Characteristics:

- Energy efficient
- Co-locate processing and memory
- Parallel computing

- Possible alternative in **Neuromorphic Computing:** brain-inspired computing paradigm. **Computation and physical substrate go hand by hand**

Characteristics:

- Energy efficient
- Co-locate processing and memory
- Parallel computing

Machine learning is already taking advantage of this progress. Example: Reservoir Computing

What is **Reservoir Computing**?

What is **Reservoir Computing**?

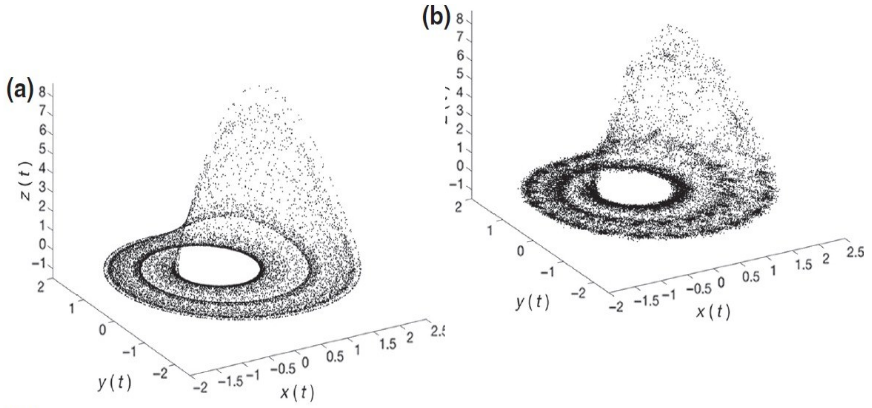
Short answer: machine learning technique that **exploits dynamical systems**

What is **Reservoir Computing**?

Short answer: machine learning technique that **exploits dynamical systems**

Target: Temporal tasks
(working with **temporal series**)

- Dynamical systems
- Noisy processes
- Input/output maps

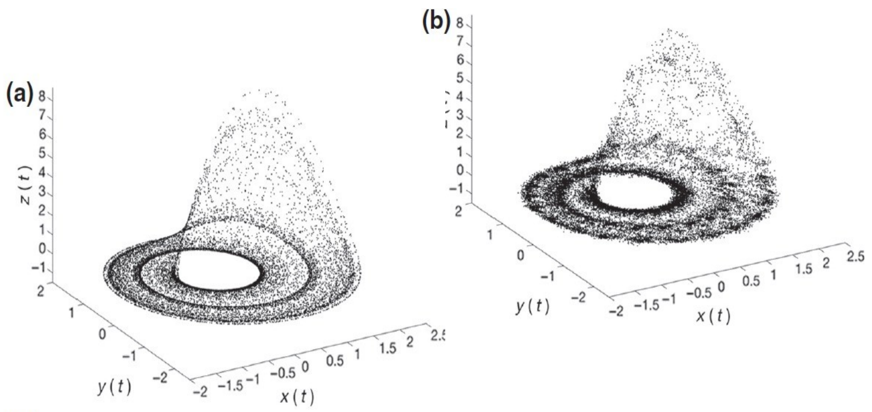


What is **Reservoir Computing**?

Short answer: machine learning technique that **exploits dynamical systems**

Target: Temporal tasks (working with **temporal series**)

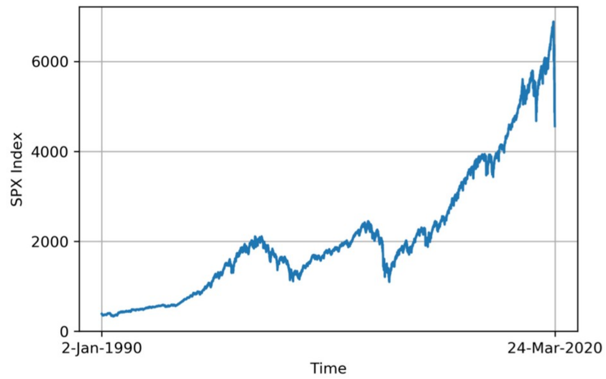
- Dynamical systems
- Noisy processes
- Input/output maps



Applications:

- Biomedicine
- Engineering
- Financial
- Communication...

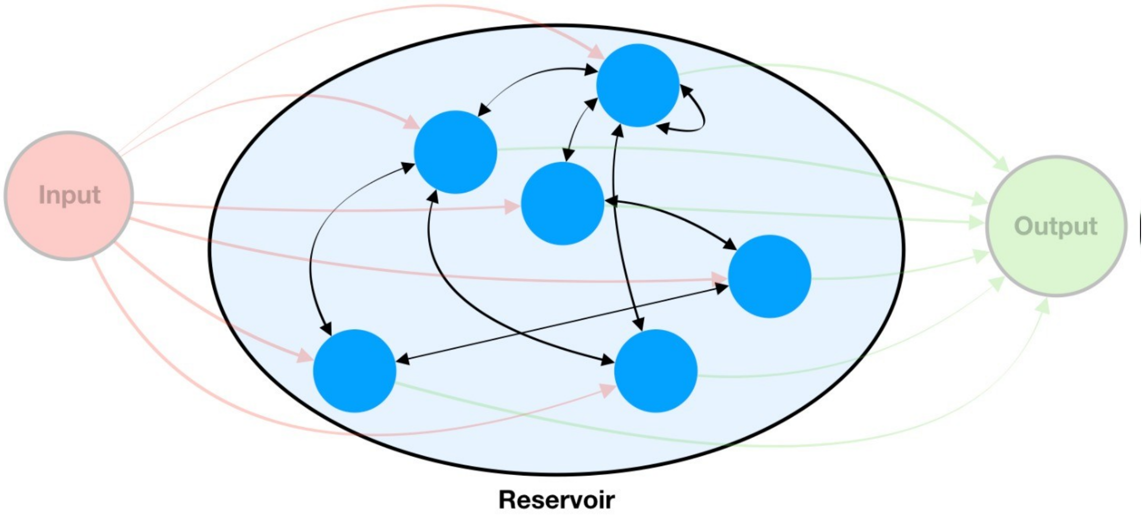
Gouhei Tanaka et al. *Neural Networks* 2019



How does it work?

How does it work?

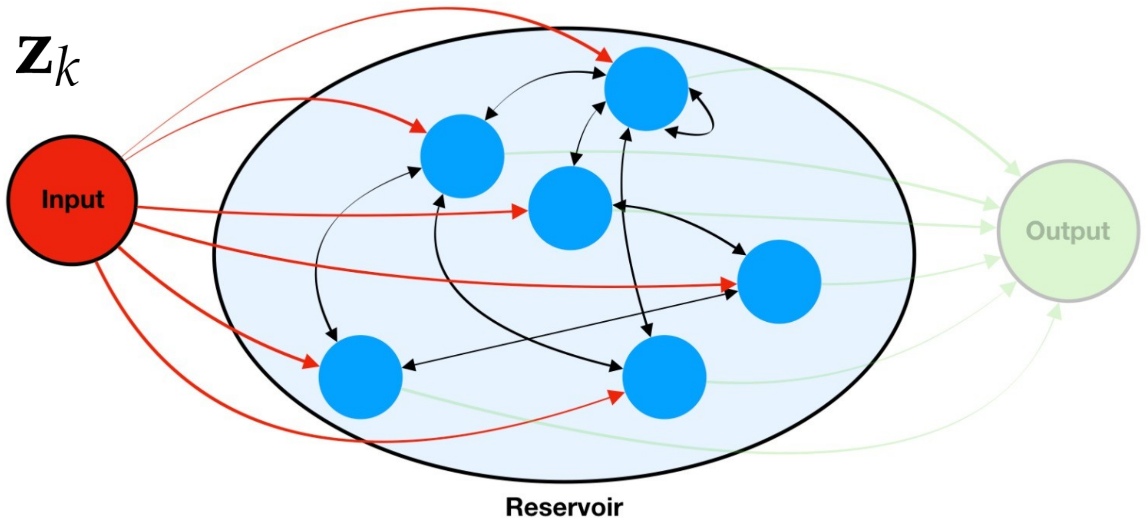
Take a dynamical system (**reservoir**)



How does it work?

Take a dynamical system (**reservoir**)

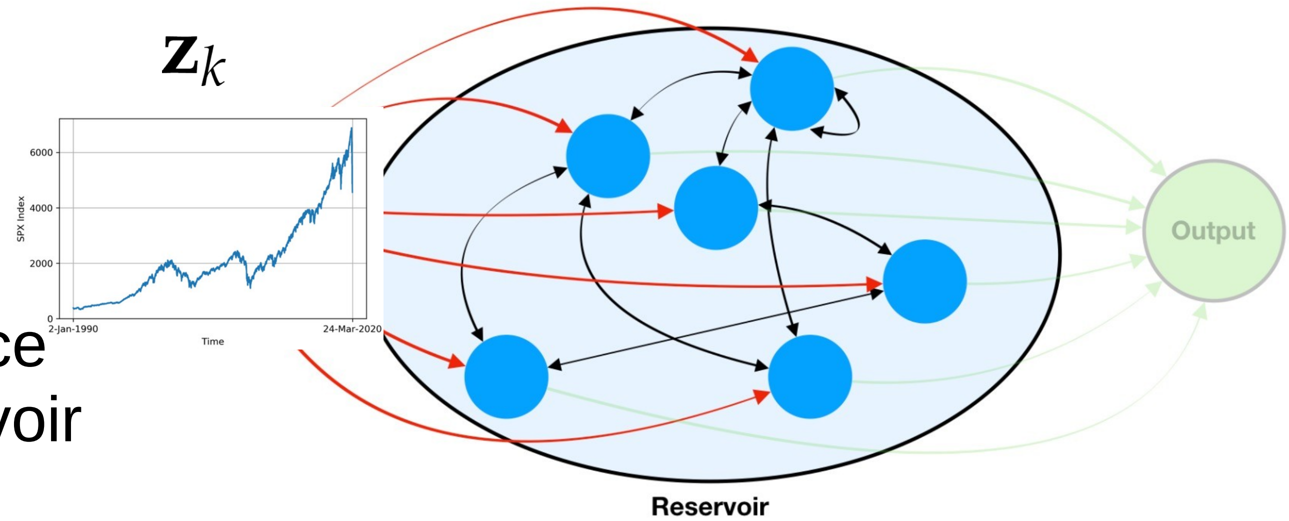
Every time step k, introduce the input $\mathbf{z}_k$ into the reservoir



How does it work?

Take a dynamical system (**reservoir**)

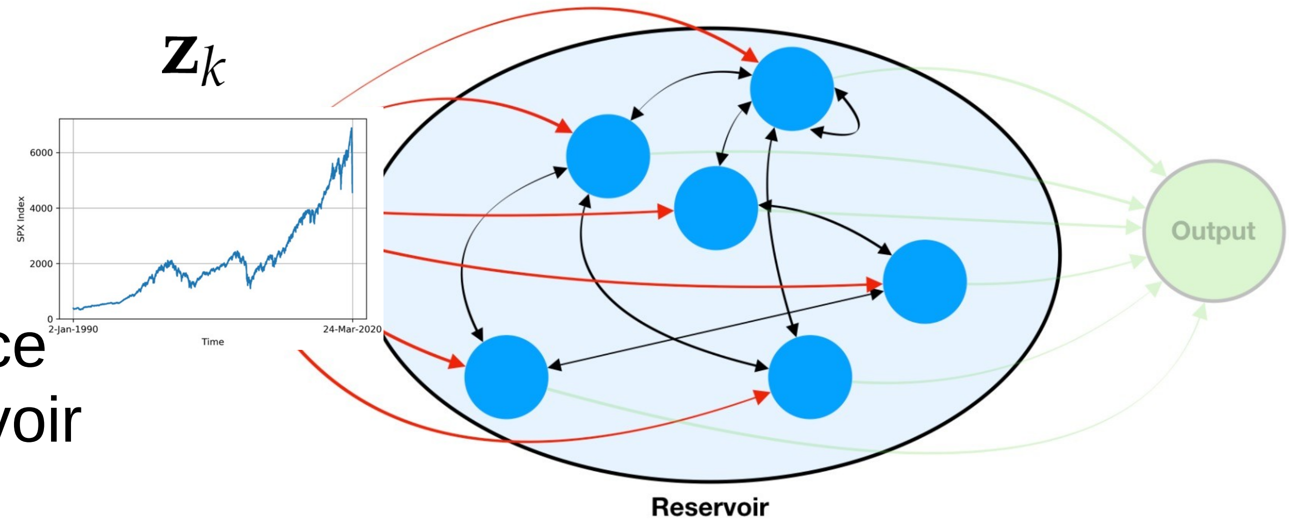
Every time step k, introduce the input $\mathbf{z}_k$ into the reservoir



How does it work?

$$\mathbf{x}_k = T(\mathbf{x}_{k-1}, \mathbf{z}_k)$$

$\mathbf{z}_k$



Take a dynamical system (**reservoir**)

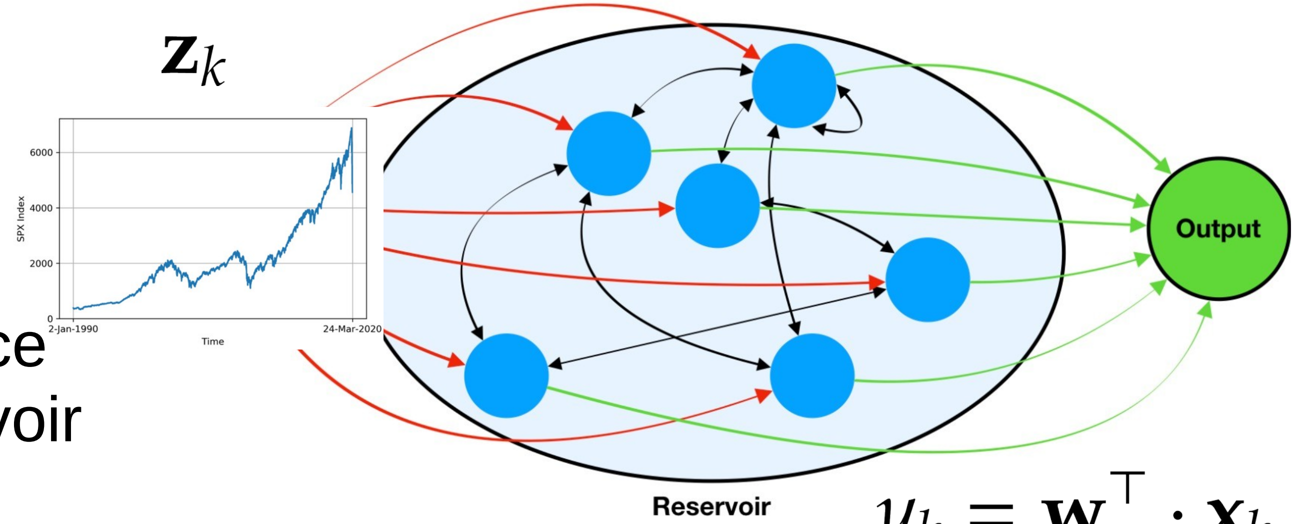
Every time step k, introduce the input $\mathbf{z}_k$ into the reservoir

Natural dynamics process the information ($T \Rightarrow \mathbf{x}_k$)

How does it work?

$$\mathbf{x}_k = T(\mathbf{x}_{k-1}, \mathbf{z}_k)$$

$\mathbf{z}_k$



$$y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$$

Take a dynamical system (**reservoir**)

Every time step k, introduce the input $\mathbf{z}_k$ into the reservoir

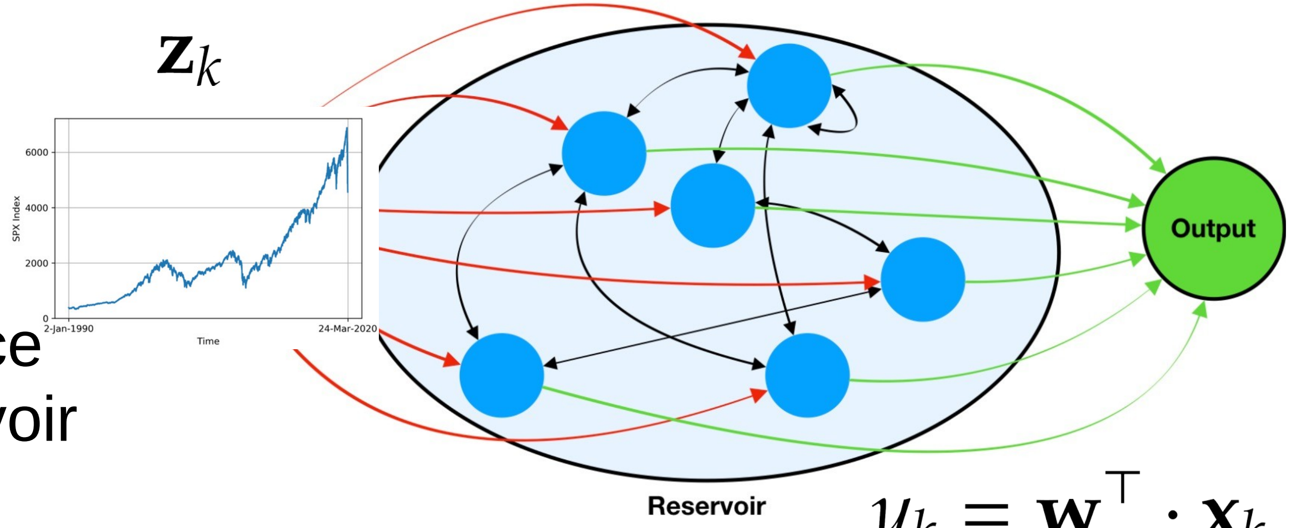
Natural dynamics process the information ($T \Rightarrow \mathbf{x}_k$)

Extract the output information from $\mathbf{x}_k$: **linear combination** y_k

How does it work?

$$\mathbf{x}_k = T(\mathbf{x}_{k-1}, \mathbf{z}_k)$$

$\mathbf{z}_k$



$$y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$$

Take a dynamical system (**reservoir**)

Every time step k, introduce the input $\mathbf{z}_k$ into the reservoir

Natural dynamics process the information ($T \Rightarrow \mathbf{x}_k$)

Extract the output information from $\mathbf{x}_k$: **linear combination** y_k

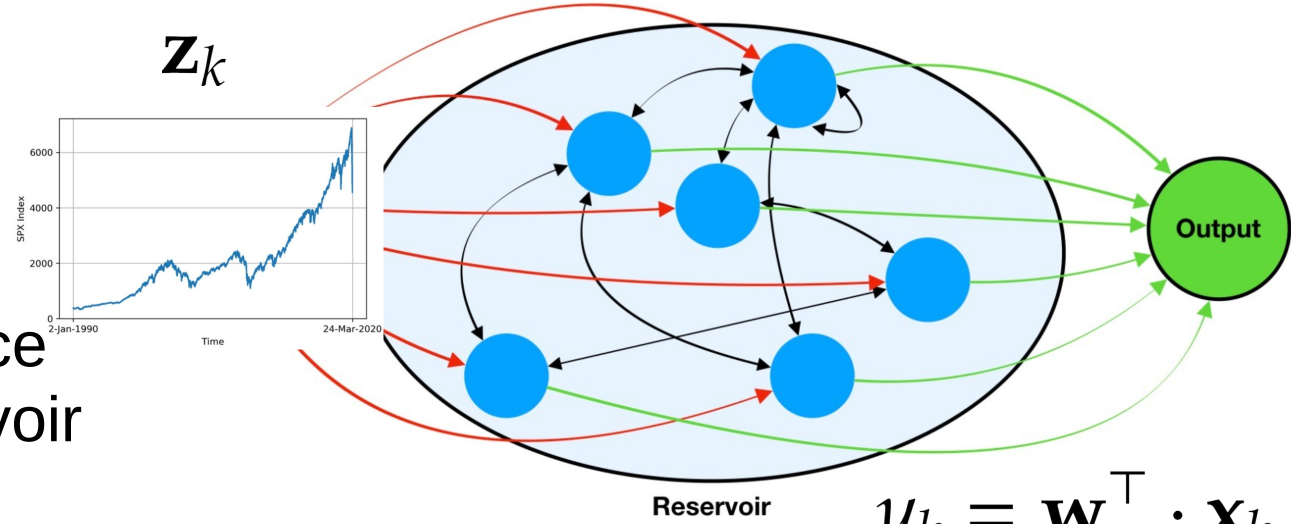
Supervised ML!

Train **only** the linear output y_k, **don't need to tune the dynamical system**!!

How does it work?

$$\mathbf{x}_k = T(\mathbf{x}_{k-1}, \mathbf{z}_k)$$

$\mathbf{z}_k$



$$y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$$

Take a dynamical system (**reservoir**)

Every time step k, introduce the input $\mathbf{z}_k$ into the reservoir

Natural dynamics process the information ($T \Rightarrow \mathbf{x}_k$)

Extract the output information from $\mathbf{x}_k$: **linear combination** y_k

Supervised ML!

Train **only** the linear output y_k, **don't need to tune the dynamical system**!!

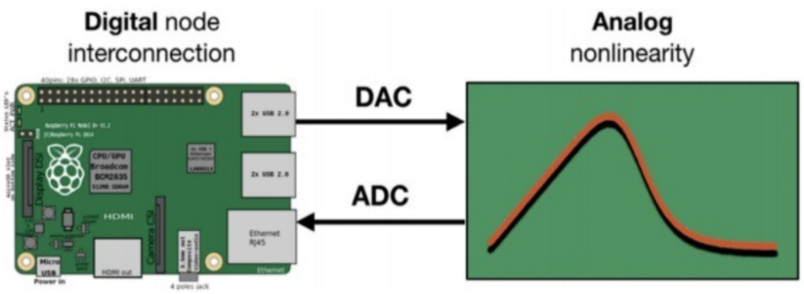
Ready to use!

Why Reservoir Computing?

- **Fast training** (linear regression is enough!) $y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$
- **Multitasking** (don’t need to tune the dynamical system!!)
- **Implementable in hardware**

Why Reservoir Computing?

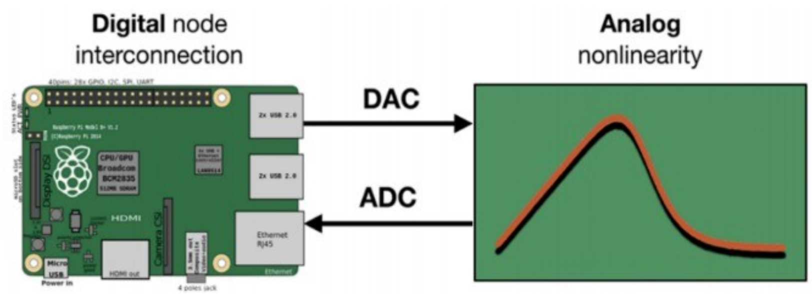
- **Fast training** (linear regression is enough!) $y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$
- **Multitasking** (don’t need to tune the dynamical system!!)
- **Implementable in hardware**



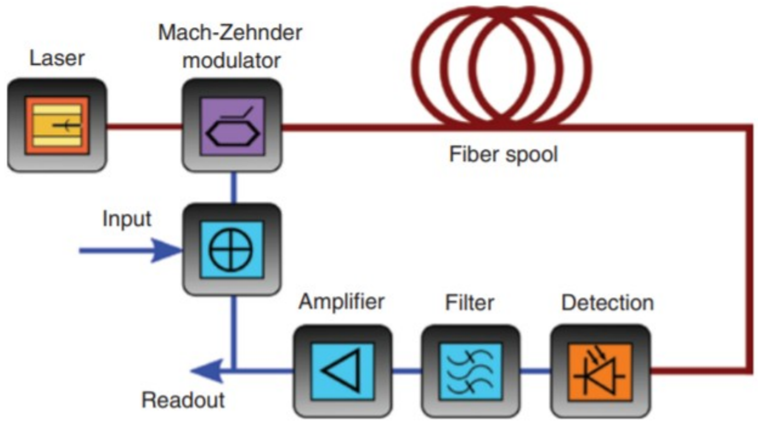
Irene Estébanez et al. *Phys. Rev. App.* 2019

Why Reservoir Computing?

- **Fast training** (linear regression is enough!) $y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$
- **Multitasking** (don’t need to tune the dynamical system!!)
- **<u>Implementable in hardware</u>**



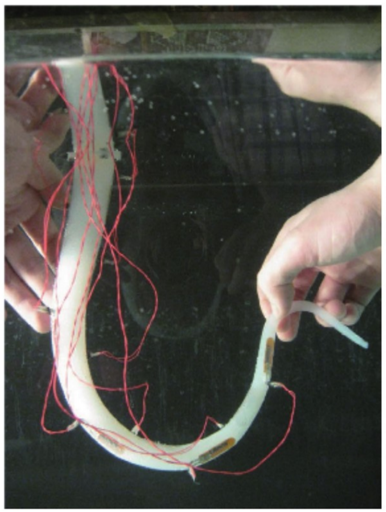
Irene Estébanez et al. *Phys. Rev. App.* 2019



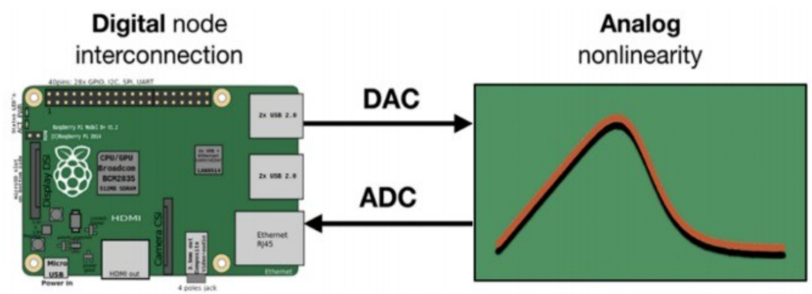
Van der Sande et al. *Nanophotonics* 2017

Why Reservoir Computing?

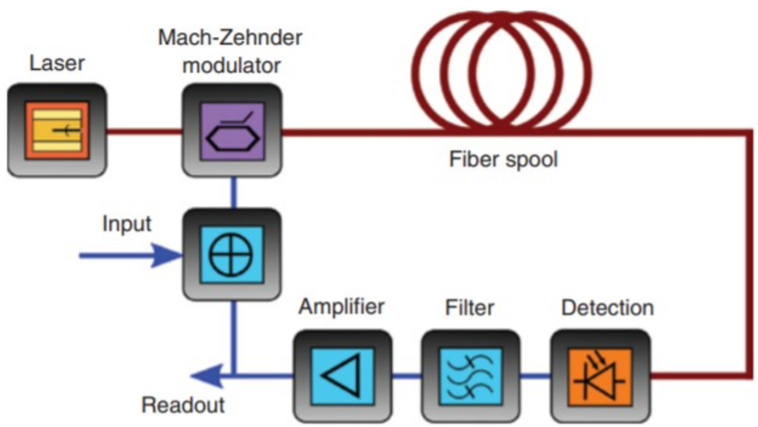
- **Fast training** (linear regression is enough!) $y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$
- **Multitasking** (don’t need to tune the dynamical system!!)
- <u>**Implementable in hardware**</u>



Nakajima et al.
Scientific Reports
2015



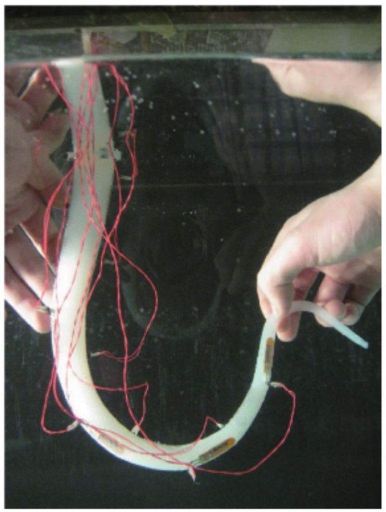
Irene Estébanez et al.
Phys. Rev. App.
2019



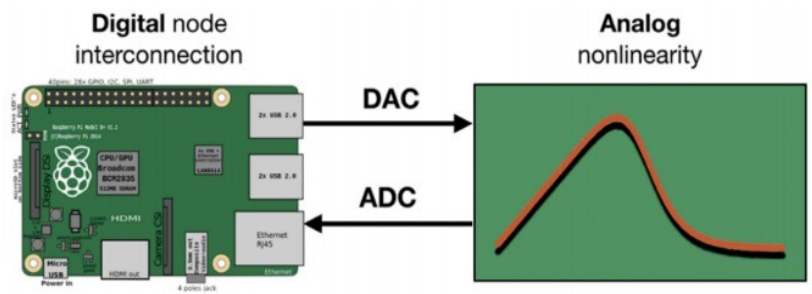
Van der Sande et al.
Nanophotonics
2017

Why Reservoir Computing?

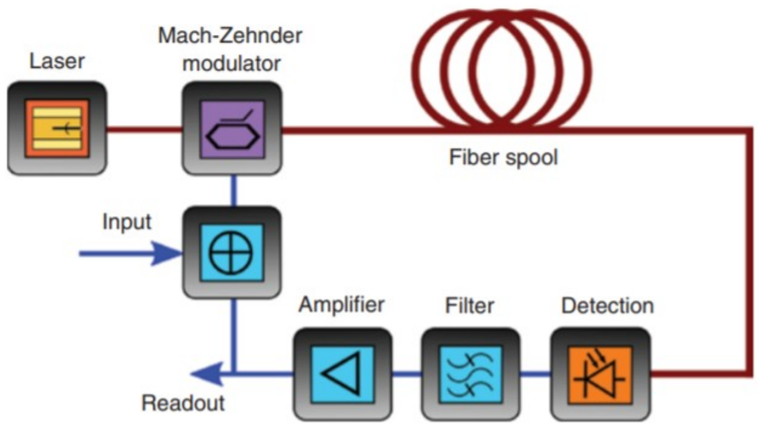
- **Fast training** (linear regression is enough!) $y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$
- **Multitasking** (don't need to tune the dynamical system!!)
- **<u>Implementable in hardware</u>**



Nakajima et al.
Scientific Reports
2015



Irene Estébanez et al.
Phys. Rev. App.
2019

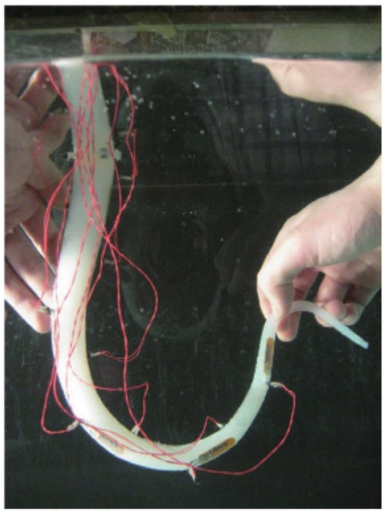


Van der Sande et al.
Nanophotonics
2017

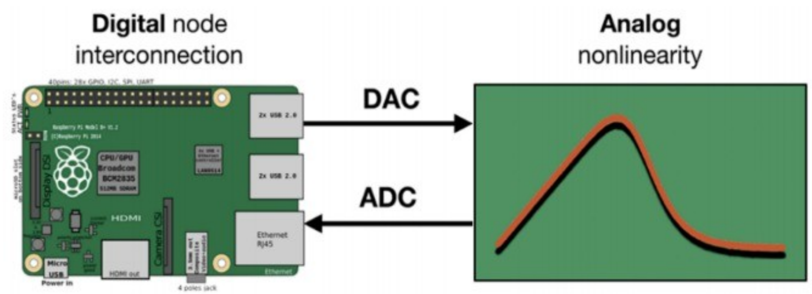
C. Fernando and S.Sojakka
European Conference on Artificial Life 2003

Why Reservoir Computing?

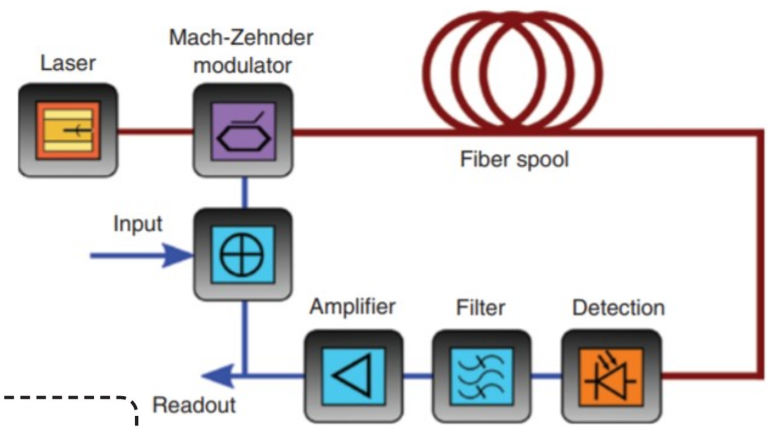
- **Fast training** (linear regression is enough!) $y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$
- **Multitasking** (don't need to tune the dynamical system!!)
- <u>**Implementable in hardware**</u>



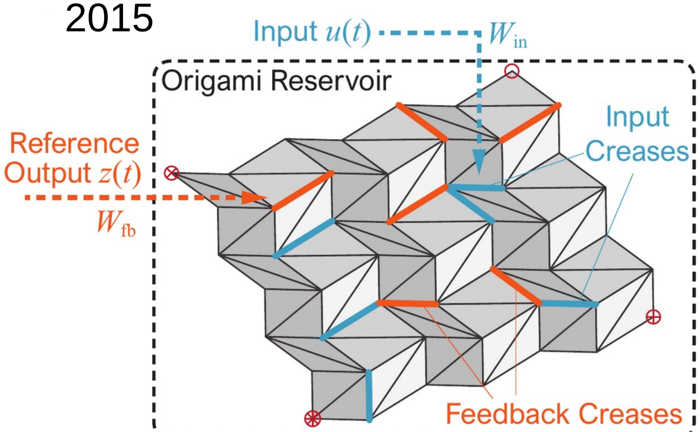
Nakajima et al.
Scientific Reports
2015



Irene Estébanez
et al.
Phys. Rev. App.
2019

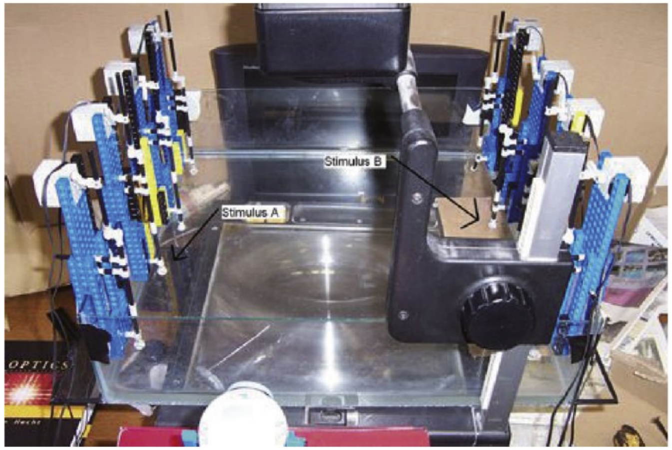


Van der Sande
et al.
Nanophotonics
2017



P. Bhovad and S. Li.
Scientific Reports 2021

C. Fernando and S.Sojakka
European Conference on Artificial Life 2003



Why Reservoir Computing?

- **Fast training** (linear regression is enough!) $y_k = \mathbf{w}^\top \cdot \mathbf{x}_k$
- **Multitasking** (don’t need to tune the dynamical system!!)
- **Implementable in hardware**

But what about **quantum systems**?

What is **Quantum Reservoir Computing**?

PERSPECTIVE

ADVANCED QUANTUM TECHNOLOGIES
www.advquantumtech.com

Opportunities in Quantum Reservoir Computing and Extreme Learning Machines

2021!

*Pere Mujal, Rodrigo Martínez-Peña, Johannes Nokkala, Jorge García-Beni, Gian Luca Giorgi, Miguel C. Soriano, and Roberta Zambrini**

What is **Quantum Reservoir Computing**?

PERSPECTIVE

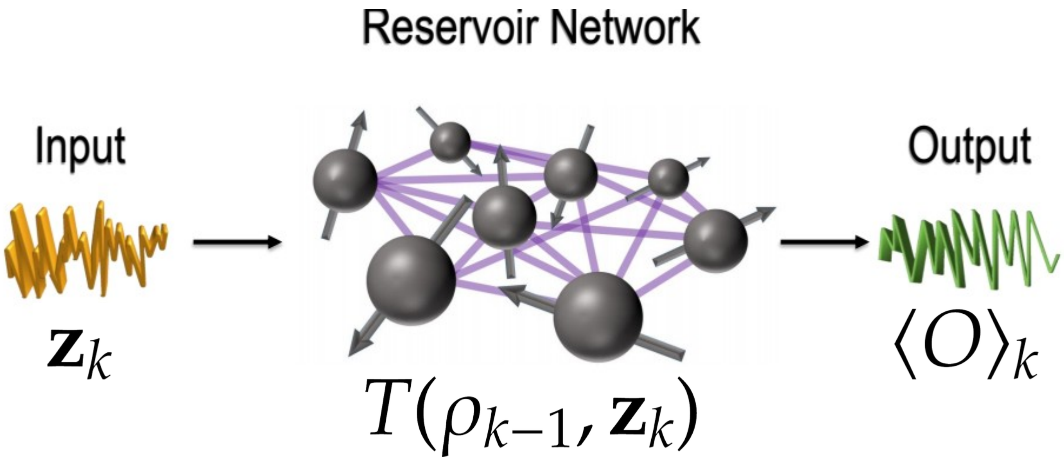
ADVANCED QUANTUM TECHNOLOGIES
www.advquantumtech.com

Opportunities in Quantum Reservoir Computing and Extreme Learning Machines

2021!

*Pere Mujal, Rodrigo Martínez-Peña, Johannes Nokkala, Jorge García-Beni, Gian Luca Giorgi, Miguel C. Soriano, and Roberta Zambrini**

Use a quantum system as the dynamical system (**reservoir**)



What is **Quantum Reservoir Computing**?

PERSPECTIVE

ADVANCED QUANTUM TECHNOLOGIES
www.advquantumtech.com

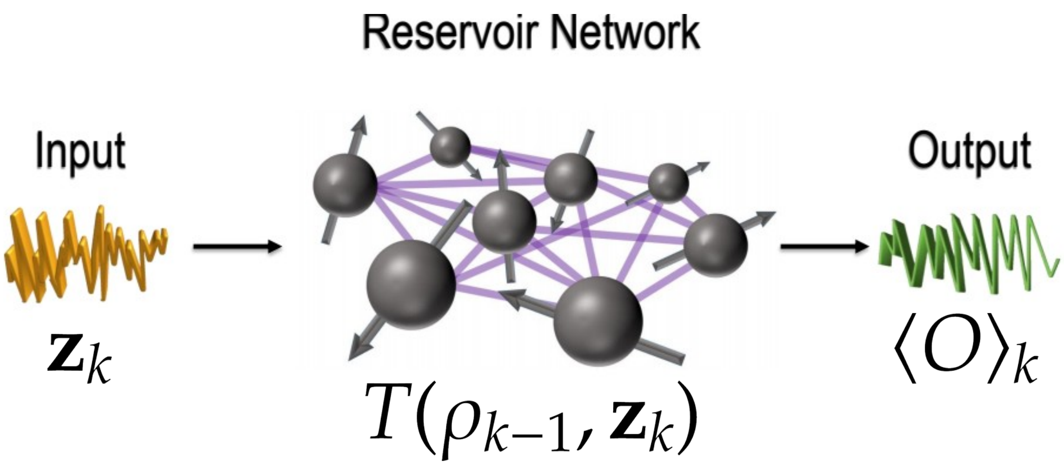
Opportunities in Quantum Reservoir Computing and Extreme Learning Machines

2021!

*Pere Mujal, Rodrigo Martínez-Peña, Johannes Nokkala, Jorge García-Beni, Gian Luca Giorgi, Miguel C. Soriano, and Roberta Zambrini**

Use a quantum system as the dynamical system (**reservoir**)

Use the **observables** for the **output**!



What is **Quantum Reservoir Computing**?

PERSPECTIVE

ADVANCED QUANTUM TECHNOLOGIES

www.advquantumtech.com

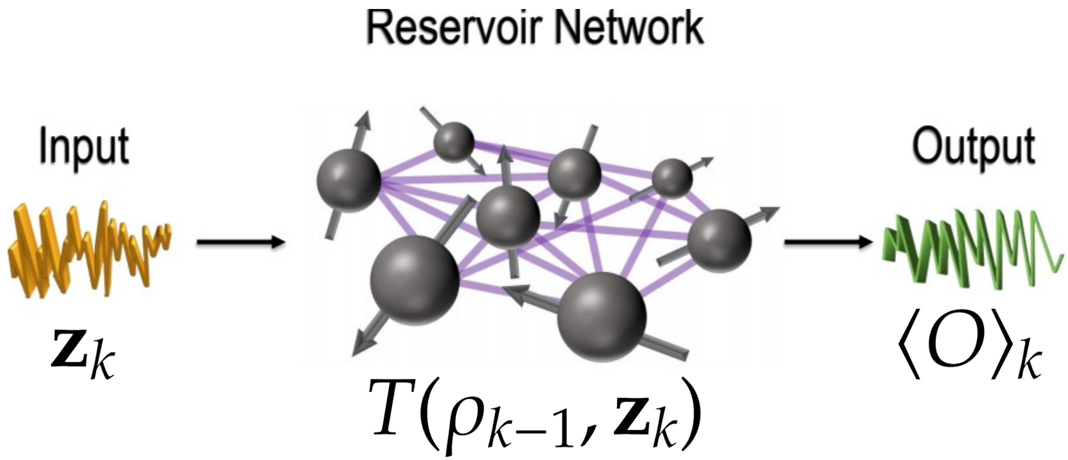
Opportunities in Quantum Reservoir Computing and Extreme Learning Machines

*Pere Mujal, Rodrigo Martínez-Peña, Johannes Nokkala, Jorge García-Beni, Gian Luca Giorgi, Miguel C. Soriano, and Roberta Zambrini**

2021!

Use a quantum system as the dynamical system (**reservoir**)

Use the **observables** for the **output**!



Why quantum?

- Number of degrees of freedom **increases exponentially in few body systems**
- Accessible on **noisy quantum devices**
- Extension to process **quantum data**

- **Context**: only a **few theoretical results** in the field. Mainly **sufficient conditions for useful reservoir systems**.

- <u>**Context**</u>: only a **few theoretical results** in the field. Mainly **sufficient conditions for useful reservoir systems**.

- <u>**Useful reservoirs**</u>: those that ensure a **minimum level of performance**.

Motivation of this work

- **Context**: only a **few theoretical results** in the field. Mainly **sufficient conditions for useful reservoir systems**.

- **Useful reservoirs**: those that ensure a **minimum level of performance**.

Quantum Information Processing (2019) 18:198
https://doi.org/10.1007/s11128-019-2311-9

Learning nonlinear input–output maps with dissipative quantum systems

Jiayin Chen[1] · Hendra I. Nurdin[1]

ARTICLE
https://doi.org/10.1038/s42005-021-00556-w OPEN
Communication Physics, 2021
Check for updates

Gaussian states of continuous-variable quantum systems provide universal and versatile reservoir computing

Johannes Nokkala[1✉], Rodrigo Martínez-Peña[1], Gian Luca Giorgi[1], Valentina Parigi[2], Miguel C. Soriano[1] & Roberta Zambrini[1✉]

PHYSICAL REVIEW LETTERS **127,** 260401 (2021)

Learning Temporal Quantum Tomography

Quoc Hoan Tran[1,*] and Kohei Nakajima[1,2,†]
[1]*Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan*
[2]*Next Generation Artificial Intelligence Research Center, The University of Tokyo, Tokyo 113-8656, Japan*

(Received 5 April 2021; revised 11 October 2021; accepted 30 November 2021; published 22 December 2021)

Dissipation as a resource for Quantum Reservoir Computing

Antonio Sannia, Rodrigo Martínez-Peña, Miguel C. Soriano, Gian Luca Giorgi, and Roberta Zambrini

Institute for Cross-Disciplinary Physics and Complex Systems (IFISC) UIB-CSIC, Campus Universitat Illes Balears, 07122, Palma de Mallorca, Spain.

ArXiv: 2212.12078

Motivation of this work

- **<u>Context</u>**: only a **few theoretical results** in the field. Mainly **sufficient conditions for useful reservoir systems**.

- **<u>Useful reservoirs</u>**: those that ensure a **minimum level of performance**.

Quantum Information Processing (2019) 18:198
https://doi.org/10.1007/s11128-019-2311-9

Learning nonlinear input–output maps with dissipative quantum systems

Jiayin Chen[1] · Hendra I. Nurdin[1]

PHYSICAL REVIEW LETTERS **127,** 260401 (2021)

Learning Temporal Quantum Tomography

Quoc Hoan Tran[1,*] and Kohei Nakajima[1,2,†]
[1]*Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan*
[2]*Next Generation Artificial Intelligence Research Center, The University of Tokyo, Tokyo 113-8656, Japan*

(Received 5 April 2021; revised 11 October 2021; accepted 30 November 2021; published 22 December 2021)

ARTICLE
https://doi.org/10.1038/s42005-021-00556-w OPEN

Communication Physics, 2021

Check for updates

Gaussian states of continuous-variable quantum systems provide universal and versatile reservoir computing

Johannes Nokkala[1✉], Rodrigo Martínez-Peña[1], Gian Luca Giorgi[1], Valentina Parigi[2], Miguel C. Soriano[1] & Roberta Zambrini[1✉]

Dissipation as a resource for Quantum Reservoir Computing

Antonio Sannia, Rodrigo Martínez-Peña, Miguel C. Soriano, Gian Luca Giorgi, and Roberta Zambrini

Institute for Cross-Disciplinary Physics and Complex Systems (IFISC) UIB-CSIC, Campus Universitat Illes Balears, 07122, Palma de Mallorca, Spain.

ArXiv: 2212.12078

- **<u>What we did</u>**: find **necessary and sufficient conditions** of **useful reservoir systems.**

- **Context**: only a **few theoretical results** in the field. Mainly **sufficient conditions for useful reservoir systems**.

- **Useful reservoirs**: those that ensure a **minimum level of performance**.

Quantum Information Processing (2019) 18:198
https://doi.org/10.1007/s11128-019-2311-9

Learning nonlinear input–output maps with dissipative quantum systems

Jiayin Chen[1] · Hendra I. Nurdin[1]

ARTICLE
https://doi.org/10.1038/s42005-021-00556-w OPEN

Communication Physics, 2021

Check for updates

Gaussian states of continuous-variable quantum systems provide universal and versatile reservoir computing

Johannes Nokkala[1✉], Rodrigo Martínez-Peña[1], Gian Luca Giorgi[1], Valentina Parigi[2], Miguel C. Soriano[1] & Roberta Zambrini[1✉]

PHYSICAL REVIEW LETTERS **127**, 260401 (2021)

Learning Temporal Quantum Tomography

Quoc Hoan Tran[1,*] and Kohei Nakajima[1,2,†]
[1]*Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan*
[2]*Next Generation Artificial Intelligence Research Center, The University of Tokyo, Tokyo 113-8656, Japan*

(Received 5 April 2021; revised 11 October 2021; accepted 30 November 2021; published 22 December 2021)

Dissipation as a resource for Quantum Reservoir Computing

Antonio Sannia, Rodrigo Martínez-Peña, Miguel C. Soriano, Gian Luca Giorgi, and Roberta Zambrini

Institute for Cross-Disciplinary Physics and Complex Systems (IFISC) UIB-CSIC, Campus Universitat Illes Balears, 07122, Palma de Mallorca, Spain.

ArXiv: 2212.12078

- **What we did**: find **necessary and sufficient conditions** of **useful reservoir systems.**

- **Why it is important**: **universal approximation property** and connection with **experimental design.**

Setup:

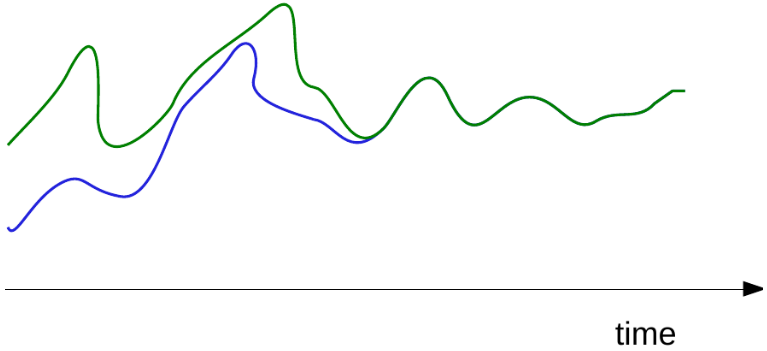
- Finite dimensional system
- Classical input information
- Ideal measurements (Some experimental works can be approximated by this limit)

Setup:

- Finite dimensional system
- Classical input information
- Ideal measurements (Some experimental works can be approximated by this limit)

Result:

A quantum reservoir is **useful if and only** if the dynamics of the quantum channel **converges** towards **input-dependent** fixed points



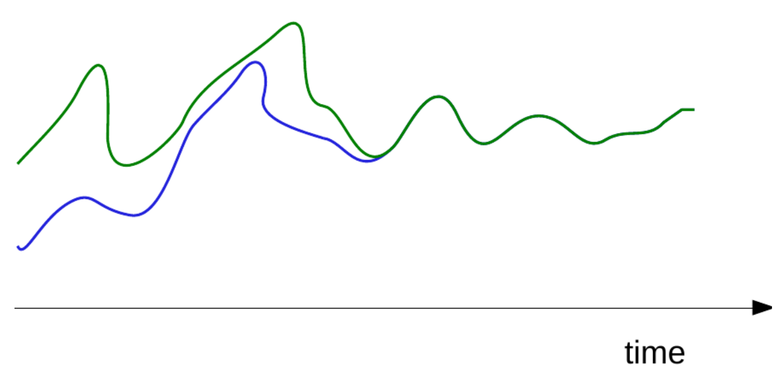
Setup:

- Finite dimensional system
- Classical input information
- Ideal measurements (Some experimental works can be approximated by this limit)

Result:

A quantum reservoir is **useful if and only** if the dynamics of the quantum channel **converges** towards **input-dependent** fixed points

Result 2



Result 1

Result 1

There are **two “scales”** of convergence:

There are **two “scales”** of convergence:

- At **each time step**: convergence of the quantum channel towards its fixed point

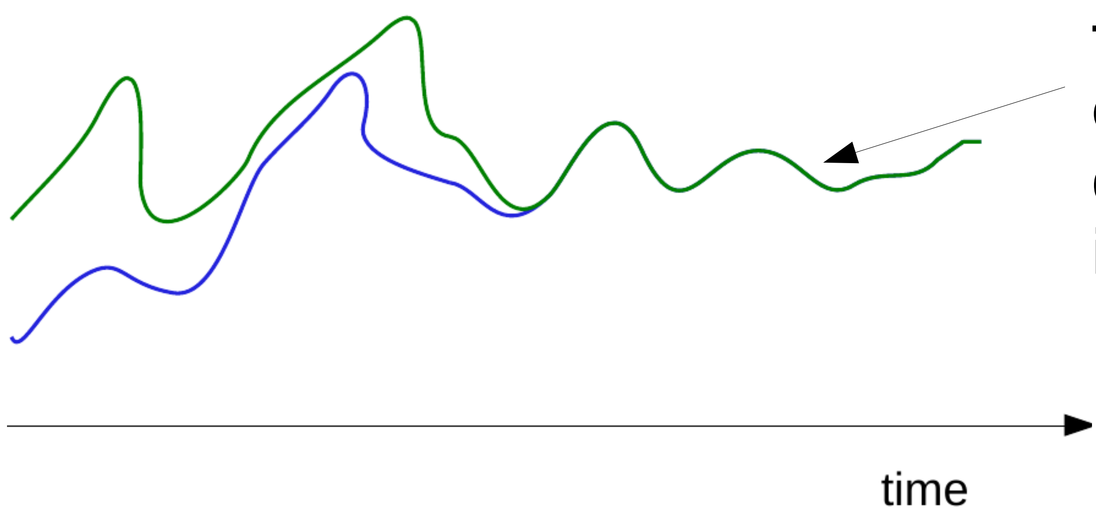
There are **two “scales”** of convergence:

- At **each time step**: convergence of the quantum channel towards its fixed point
- **Over time steps**: fading memory property

There are **two “scales”** of convergence:

- At **each time step**: convergence of the quantum channel towards its fixed point
- **Over time steps**: fading memory property

- Fading Memory Property (FMP): **memory** of past inputs.



Reservoir dynamics **forgets the initial conditions and dissipates past inputs**

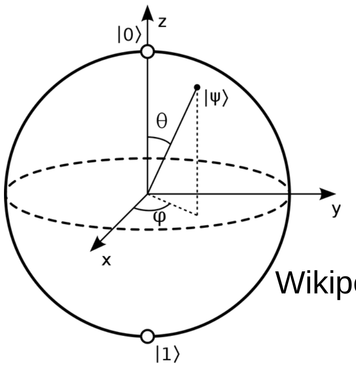
- **Result 1: merging the two scales of convergence.** A quantum reservoir has the fading memory property **if and only if** the dynamics of channel is contracted in some norm at each time step.

- **<u>Result 1: merging the two scales of convergence.</u>** A quantum reservoir has the fading memory property **if and only if** the dynamics of channel is contracted in some norm at each time step.

Example of one qubit:

$$\rho \Rightarrow \mathbf{x} = (1, \langle \sigma^x \rangle, \langle \sigma^y \rangle, \langle \sigma^z \rangle)$$

$$\{B_i\} = \{I, \sigma^x, \sigma^y, \sigma^z\}$$



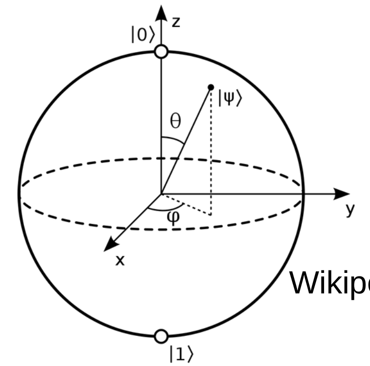
Wikipedia

- **<u>Result 1: merging the two scales of convergence.</u>** A quantum reservoir has the fading memory property **if and only if** the dynamics of channel is contracted in some norm at each time step.

Example of one qubit:

$$\rho \Rightarrow \mathbf{x} = (1, \langle\sigma^x\rangle, \langle\sigma^y\rangle, \langle\sigma^z\rangle)$$

$$\{B_i\} = \{I, \sigma^x, \sigma^y, \sigma^z\}$$



Quantum channel in observables form:

$$x_{k,j} = (\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0})_{i,j} x_{k-1,j} + \widehat{T}(\mathbf{z}_k)_{0j}$$

$\mathbf{z}_k$: input

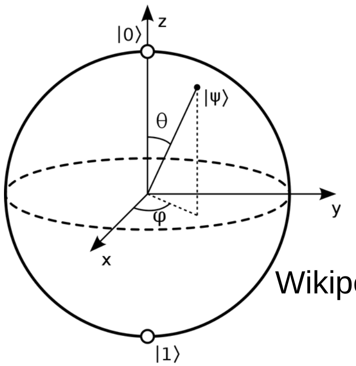
$$\mathcal{H}_0 = \{\sigma^x, \sigma^y, \sigma^z\}$$

$$\widehat{T}(\mathbf{z}_k)_{ij} = \frac{1}{2}\mathrm{tr}(B_i T(B_j, \mathbf{z}_k))$$

- **<u>Result 1: merging the two scales of convergence.</u>** A quantum reservoir has the fading memory property **if and only if** the dynamics of channel is contracted in some norm at each time step.

Example of one qubit:

$$\rho \Rightarrow \mathbf{x} = (1, \langle \sigma^x \rangle, \langle \sigma^y \rangle, \langle \sigma^z \rangle)$$
$$\{B_i\} = \{I, \sigma^x, \sigma^y, \sigma^z\}$$



Wikipedia

Quantum channel in observables form:

$$x_{k,j} = (\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0})_{i,j} x_{k-1,j} + \widehat{T}(\mathbf{z}_k)_{0j} \qquad \mathcal{H}_0 = \{\sigma^x, \sigma^y, \sigma^z\}$$

input

$$\widehat{T}(\mathbf{z}_k)_{ij} = \frac{1}{2}\mathrm{tr}(B_i T(B_j, \mathbf{z}_k))$$

$$|||\widehat{A}||| := \sup_{||\mathbf{x}|| \neq 0} \frac{||\widehat{A\mathbf{x}}||}{||\mathbf{x}||}$$

Induced norm $|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| < 1$

D. J. Hartfiel, *Nonhomogeneous matrix products*, 2002

There is an equivalent condition in density matrix form

- **<u>Result 2:</u>** a quantum reservoir becomes **input-dependent (useful in the long-term) if and only if** the quantum channel have a unique **input-dependent fixed point**.

Quantum channel in matrix form:

$$T(\rho^*(\mathbf{z}_k), \mathbf{z}_k) = \rho^*(\mathbf{z}_k)$$

input

fixed point

- **Result 2:** a quantum reservoir becomes **input-dependent (useful in the long-term) if and only if** the quantum channel have a unique **input-dependent fixed point**.

Quantum channel in matrix form:

$$T(\rho^*(\mathbf{z}_k), \mathbf{z}_k) = \rho^*(\mathbf{z}_k)$$

input

fixed point

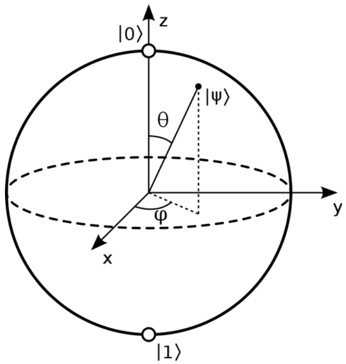
- **Intuition:** convergence makes the system approach an attractor. To **drive** a system **over time steps** you need **attractors that depends on the input.**

Is example 1 a useful quantum reservoir?

Example 1:

$$\dot{\rho} = -i[H,\rho] + \gamma L\rho L^\dagger - \frac{\gamma}{2}\{L^\dagger L, \rho\}$$

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^z \qquad L = \sigma^-$$



Solution 1:

$$\dot{\rho} = -i[H, \rho] + \gamma L \rho L^\dagger - \frac{\gamma}{2}\{L^\dagger L, \rho\}$$

Does this system fading memory property?

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^z \qquad L = \sigma^-$$

Solution 1:

$$\dot{\rho} = -i[H, \rho] + \gamma L \rho L^\dagger - \frac{\gamma}{2}\{L^\dagger L, \rho\}$$

Does this system fading memory property?

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^z \qquad L = \sigma^-$$

Bloch vector representation:

Solution 1:

$$\dot{\rho} = -i[H,\rho] + \gamma L\rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L,\rho\}$$

Does this system fading memory property?

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^z \qquad L = \sigma^-$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-\frac{\gamma\Delta t}{2}}\cos(h_k\Delta t) & e^{-\frac{\gamma\Delta t}{2}}\sin(h_k\Delta t) & 0 \\ 0 & -e^{-\frac{\gamma\Delta t}{2}}\sin(h_k\Delta t) & e^{-\frac{\gamma\Delta t}{2}}\cos(h_k\Delta t) & 0 \\ e^{-\gamma\Delta t}-1 & 0 & 0 & e^{-\gamma\Delta t} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$$x_{k,j} = (\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0})_{i,j}x_{k-1,j} + \widehat{T}(\mathbf{z}_k)_{0j}$$

Solution 1:

$$\dot{\rho} = -i[H,\rho] + \gamma L\rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L,\rho\}$$

Does this system fading memory property?

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^z \qquad L = \sigma^-$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-\frac{\gamma\Delta t}{2}}\cos(h_k\Delta t) & e^{-\frac{\gamma\Delta t}{2}}\sin(h_k\Delta t) & 0 \\ 0 & -e^{-\frac{\gamma\Delta t}{2}}\sin(h_k\Delta t) & e^{-\frac{\gamma\Delta t}{2}}\cos(h_k\Delta t) & 0 \\ e^{-\gamma\Delta t}-1 & 0 & 0 & e^{-\gamma\Delta t} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$\widehat{T}|_{\mathcal{H}_0}$

?

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| < 1$$

$$x_{k,j} = (\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0})_{i,j}x_{k-1,j} + \widehat{T}(\mathbf{z}_k)_{0j}$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-\frac{\gamma\Delta t}{2}}\cos(h_k\Delta t) & e^{-\frac{\gamma\Delta t}{2}}\sin(h_k\Delta t) & 0 \\ 0 & -e^{-\frac{\gamma\Delta t}{2}}\sin(h_k\Delta t) & e^{-\frac{\gamma\Delta t}{2}}\cos(h_k\Delta t) & 0 \\ e^{-\gamma\Delta t}-1 & 0 & 0 & e^{-\gamma\Delta t} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$\widehat{T}|_{\mathcal{H}_0}$

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| \overset{?}{<} 1$$

Solution 1

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle \sigma^x \rangle_k \\ \langle \sigma^y \rangle_k \\ \langle \sigma^z \rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-\frac{\gamma \Delta t}{2}} \cos(h_k \Delta t) & e^{-\frac{\gamma \Delta t}{2}} \sin(h_k \Delta t) & 0 \\ 0 & -e^{-\frac{\gamma \Delta t}{2}} \sin(h_k \Delta t) & e^{-\frac{\gamma \Delta t}{2}} \cos(h_k \Delta t) & 0 \\ e^{-\gamma \Delta t} - 1 & 0 & 0 & e^{-\gamma \Delta t} \end{pmatrix} \begin{pmatrix} 1 \\ \langle \sigma^x \rangle_{k-1} \\ \langle \sigma^y \rangle_{k-1} \\ \langle \sigma^z \rangle_{k-1} \end{pmatrix}$$

$\widehat{T}|_{\mathcal{H}_0}$

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| \overset{?}{<} 1$$

Induced Frobenius norm:
maximum singular value

$$|||\widehat{T}|_{\mathcal{H}_0}|||_F = s_{\max}(\widehat{T}|_{\mathcal{H}_0})$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle \sigma^x \rangle_k \\ \langle \sigma^y \rangle_k \\ \langle \sigma^z \rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-\frac{\gamma \Delta t}{2}} \cos(h_k \Delta t) & e^{-\frac{\gamma \Delta t}{2}} \sin(h_k \Delta t) & 0 \\ 0 & -e^{-\frac{\gamma \Delta t}{2}} \sin(h_k \Delta t) & e^{-\frac{\gamma \Delta t}{2}} \cos(h_k \Delta t) & 0 \\ e^{-\gamma \Delta t} - 1 & 0 & 0 & e^{-\gamma \Delta t} \end{pmatrix} \begin{pmatrix} 1 \\ \langle \sigma^x \rangle_{k-1} \\ \langle \sigma^y \rangle_{k-1} \\ \langle \sigma^z \rangle_{k-1} \end{pmatrix}$$

$$\widehat{T}|_{\mathcal{H}_0}$$

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| \overset{?}{<} 1$$

Induced Frobenius norm:
maximum singular value

$$|||\widehat{T}|_{\mathcal{H}_0}|||_F = s_{\max}(\widehat{T}|_{\mathcal{H}_0})$$

$$s_1 = e^{-\gamma \Delta t} < 1$$

$$s_2 = s_3 = e^{-\gamma \Delta t / 2} < 1$$

Solution 1

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle \sigma^x \rangle_k \\ \langle \sigma^y \rangle_k \\ \langle \sigma^z \rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-\frac{\gamma \Delta t}{2}} \cos(h_k \Delta t) & e^{-\frac{\gamma \Delta t}{2}} \sin(h_k \Delta t) & 0 \\ 0 & -e^{-\frac{\gamma \Delta t}{2}} \sin(h_k \Delta t) & e^{-\frac{\gamma \Delta t}{2}} \cos(h_k \Delta t) & 0 \\ e^{-\gamma \Delta t} - 1 & 0 & 0 & e^{-\gamma \Delta t} \end{pmatrix} \begin{pmatrix} 1 \\ \langle \sigma^x \rangle_{k-1} \\ \langle \sigma^y \rangle_{k-1} \\ \langle \sigma^z \rangle_{k-1} \end{pmatrix}$$

$$\widehat{T}|_{\mathcal{H}_0}$$

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| \stackrel{?}{<} 1$$

Induced Frobenius norm:
maximum singular value

$$|||\widehat{T}|_{\mathcal{H}_0}|||_F = s_{\max}(\widehat{T}|_{\mathcal{H}_0})$$

$$s_1 = e^{-\gamma \Delta t} < 1$$

$$s_2 = s_3 = e^{-\gamma \Delta t/2} < 1$$

This system has fading memory

Solution 1

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle \sigma^x \rangle_k \\ \langle \sigma^y \rangle_k \\ \langle \sigma^z \rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-\frac{\gamma \Delta t}{2}} \cos(h_k \Delta t) & e^{-\frac{\gamma \Delta t}{2}} \sin(h_k \Delta t) & 0 \\ 0 & -e^{-\frac{\gamma \Delta t}{2}} \sin(h_k \Delta t) & e^{-\frac{\gamma \Delta t}{2}} \cos(h_k \Delta t) & 0 \\ e^{-\gamma \Delta t} - 1 & 0 & 0 & e^{-\gamma \Delta t} \end{pmatrix} \begin{pmatrix} 1 \\ \langle \sigma^x \rangle_{k-1} \\ \langle \sigma^y \rangle_{k-1} \\ \langle \sigma^z \rangle_{k-1} \end{pmatrix}$$

$\widehat{T}|_{\mathcal{H}_0}$

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| \overset{?}{<} 1$$

Induced Frobenius norm:
maximum singular value

$$|||\widehat{T}|_{\mathcal{H}_0}|||_F = s_{\max}(\widehat{T}|_{\mathcal{H}_0})$$

$$s_1 = e^{-\gamma \Delta t} < 1$$

$$s_2 = s_3 = e^{-\gamma \Delta t/2} < 1$$

This system has fading memory

But...

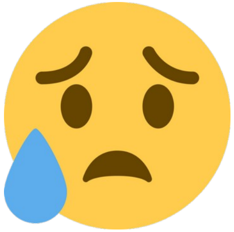
$$\dot{\rho} = -i[H,\rho] + \gamma L \rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L,\rho\}$$

<u>Solution 1</u>:

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^z \qquad L = \sigma^-$$

Input-independent fixed point:

$$\rho^* = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$



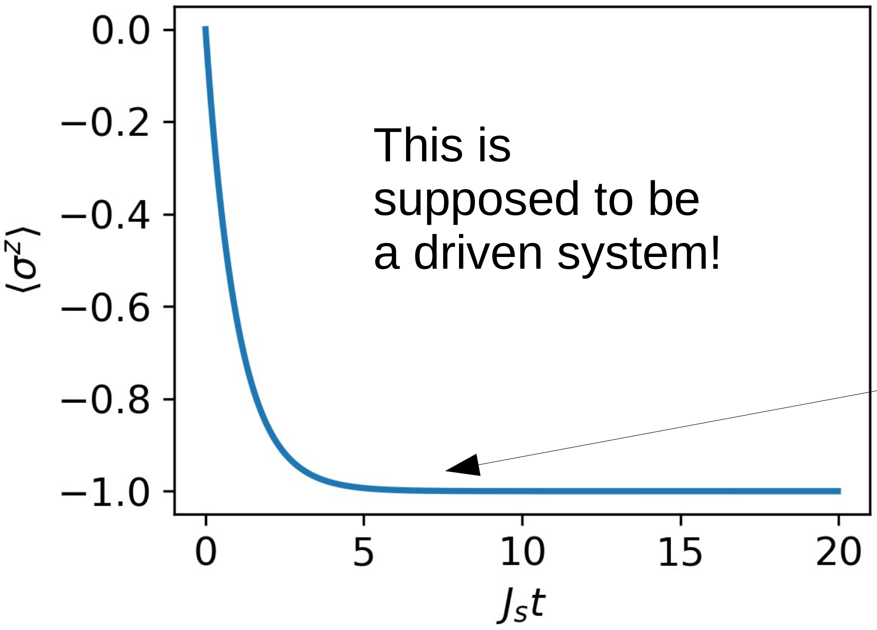
Solution 1:

$$\dot{\rho} = -i[H,\rho] + \gamma L\rho L^\dagger - \frac{\gamma}{2}\{L^\dagger L, \rho\}$$

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^z \qquad L = \sigma^-$$

Input-independent fixed point:

$$\rho^* = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$



Result 2 says that this system is **useless** for long-input sequences (or not that long...)

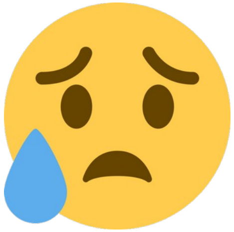
<u>Solution 1</u>:

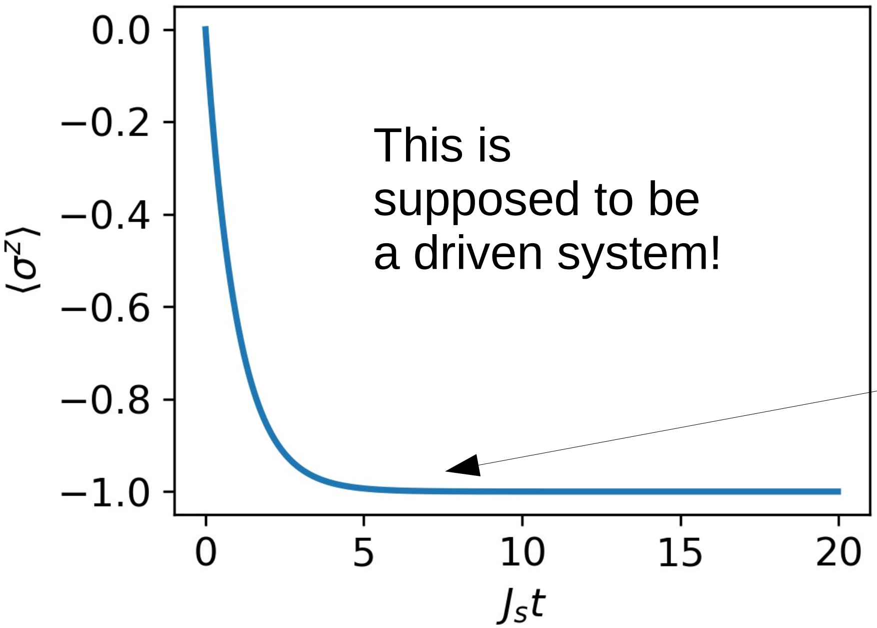
$$\dot{\rho} = -i[H,\rho] + \gamma L\rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L,\rho\}$$

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^z \qquad L = \sigma^-$$

Input-independent fixed point:

$$\rho^* = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$





Result 2 says that this system is **useless** for long-input sequences (or not that long...)

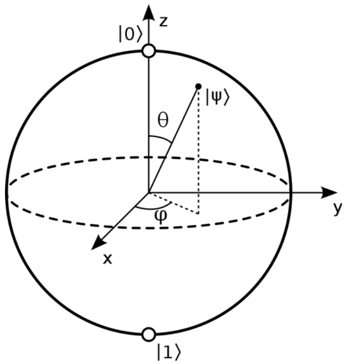
<u>Conclusion:</u> fading memory is important, but the design is very important as well!

Is example 2 a useful quantum reservoir?

<u>Example 2</u>:

$$\dot{\rho} = -i[H,\rho] + \gamma L \rho L^\dagger - \frac{\gamma}{2}\{L^\dagger L, \rho\}$$

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^x \qquad L = \sigma^-$$



Solution 2:

$$\dot{\rho} = -i[H, \rho] + \gamma L \rho L^\dagger - \frac{\gamma}{2}\{L^\dagger L, \rho\}$$

Does this system have fading memory property?

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^x \qquad L = \sigma^-$$

Solution 2:

$$\dot{\rho} = -i[H, \rho] + \gamma L \rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L, \rho\}$$

Does this system have fading memory property?

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^x \qquad L = \sigma^-$$

Bloch vector representation:

Solution 2:

$$\dot{\rho} = -i[H, \rho] + \gamma L \rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L, \rho\}$$

Does this system have fading memory property?

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^x \qquad L = \sigma^-$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle \sigma^x \rangle_k \\ \langle \sigma^y \rangle_k \\ \langle \sigma^z \rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{T}_{22} & 0 & 0 \\ \widehat{T}_{31} & 0 & \widehat{T}_{33} & \widehat{T}_{34} \\ \widehat{T}_{41} & 0 & \widehat{T}_{43} & \widehat{T}_{44} \end{pmatrix} \begin{pmatrix} 1 \\ \langle \sigma^x \rangle_{k-1} \\ \langle \sigma^y \rangle_{k-1} \\ \langle \sigma^z \rangle_{k-1} \end{pmatrix}$$

$$x_{k,j} = (\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0})_{i,j} x_{k-1,j} + \widehat{T}(\mathbf{z}_k)_{0j}$$

Solution 2:

$$\dot{\rho} = -i[H,\rho] + \gamma L \rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L, \rho\}$$

Does this system have fading memory property?

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^x \qquad L = \sigma^-$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{T}_{22} & 0 & 0 \\ \widehat{T}_{31} & 0 & \widehat{T}_{33} & \widehat{T}_{34} \\ \widehat{T}_{41} & 0 & \widehat{T}_{43} & \widehat{T}_{44} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$$\widehat{T}|_{\mathcal{H}_0}$$

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| < 1 \quad ?$$

$$x_{k,j} = (\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0})_{i,j}\, x_{k-1,j} + \widehat{T}(\mathbf{z}_k)_{0j}$$

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{T}_{22} & 0 & 0 \\ \widehat{T}_{31} & 0 & \widehat{T}_{33} & \widehat{T}_{34} \\ \widehat{T}_{41} & 0 & \widehat{T}_{43} & \widehat{T}_{44} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$$\widehat{T}_{22} = e^{-\frac{\gamma\Delta t}{2}}$$

$$\widehat{T}_{33} = e^{-\frac{3\gamma\Delta t}{4}}\left(\cosh\left(\frac{\Delta t}{4}\sqrt{\gamma^2-16h_k^2}\right) + \frac{\gamma}{\sqrt{\gamma^2-16h_k^2}}\sinh\left(\frac{\Delta t}{4}\sqrt{\gamma^2-16h_k^2}\right)\right)$$

$$\widehat{T}_{44} = e^{-\frac{3\gamma\Delta t}{4}}\left(\cosh\left(\frac{\Delta t}{4}\sqrt{\gamma^2-16h_k^2}\right) - \frac{\gamma}{\sqrt{\gamma^2-16h_k^2}}\sinh\left(\frac{\Delta t}{4}\sqrt{\gamma^2-16h_k^2}\right)\right)$$

$$\widehat{T}_{34} = \frac{4h_k e^{-\frac{3\gamma\Delta t}{4}}}{\sqrt{\gamma^2-16h_k^2}}\sinh\left(\frac{\Delta t}{4}\sqrt{\gamma^2-16h_k^2}\right)$$

$$\widehat{T}_{43} = -\widehat{T}_{34}$$

$$\widehat{T}_{31} = \frac{2\gamma h_k}{\gamma^2+2h_k^2}\left\{-1 + e^{-\frac{3\gamma\Delta t}{4}}\left(\cosh\left(\frac{\Delta t}{4}\sqrt{\gamma^2-16h_k^2}\right) + \frac{3\gamma}{\sqrt{\gamma^2-16h_k^2}}\sinh\left(\frac{\Delta t}{4}\sqrt{\gamma^2-16h_k^2}\right)\right)\right\}$$

$$\widehat{T}_{41} = \frac{\gamma}{\gamma^2+2h_k^2}\left\{-\gamma + e^{-\frac{3\gamma\Delta t}{4}}\left(\gamma\cosh\left(\frac{\Delta t}{4}\sqrt{\gamma^2-16h_k^2}\right) - \frac{\gamma^2+8h_k^2}{\sqrt{\gamma^2-16h_k^2}}\sinh\left(\frac{\Delta t}{4}\sqrt{\gamma^2-16h_k^2}\right)\right)\right\}$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{T}_{22} & 0 & 0 \\ \widehat{T}_{31} & 0 & \widehat{T}_{33} & \widehat{T}_{34} \\ \widehat{T}_{41} & 0 & \widehat{T}_{43} & \widehat{T}_{44} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$$\widehat{T}|_{\mathcal{H}_0}$$

?

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| < 1$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{T}_{22} & 0 & 0 \\ \widehat{T}_{31} & 0 & \widehat{T}_{33} & \widehat{T}_{34} \\ \widehat{T}_{41} & 0 & \widehat{T}_{43} & \widehat{T}_{44} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$$\widehat{T}|_{\mathcal{H}_0}$$

Induced Frobenius norm:
maximum singular value

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| \overset{?}{<} 1$$

$$|||\widehat{T}|_{\mathcal{H}_0}|||_F = s_{\max}(\widehat{T}|_{\mathcal{H}_0})$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{T}_{22} & 0 & 0 \\ \widehat{T}_{31} & 0 & \widehat{T}_{33} & \widehat{T}_{34} \\ \widehat{T}_{41} & 0 & \widehat{T}_{43} & \widehat{T}_{44} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$$\widehat{T}|_{\mathcal{H}_0}$$

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| < 1 \quad ?$$

Induced Frobenius norm:
maximum singular value

$$|||\widehat{T}|_{\mathcal{H}_0}|||_F = s_{\max}(\widehat{T}|_{\mathcal{H}_0})$$

$$s_1 = e^{-\gamma\Delta t/2} < 1$$

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{T}_{22} & 0 & 0 \\ \widehat{T}_{31} & 0 & \widehat{T}_{33} & \widehat{T}_{34} \\ \widehat{T}_{41} & 0 & \widehat{T}_{43} & \widehat{T}_{44} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$\widehat{T}|_{\mathcal{H}_0}$

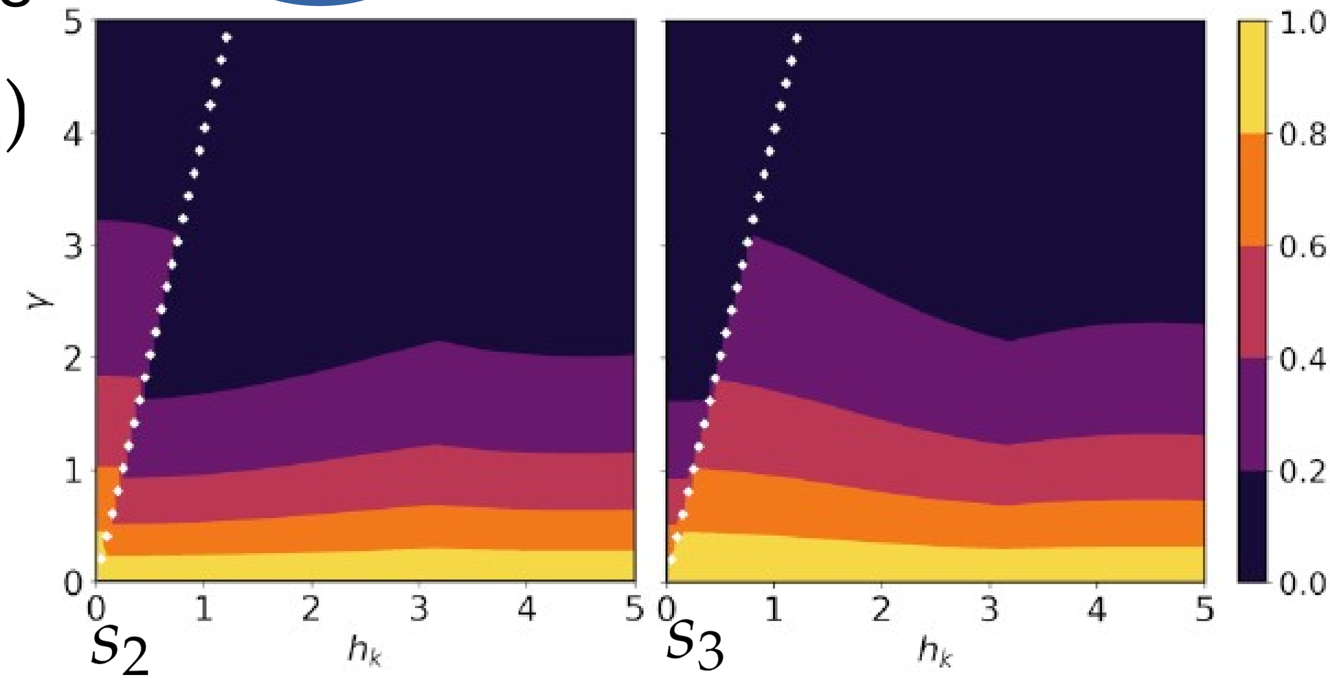
?

$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| < 1$$

Induced Frobenius norm:
maximum singular value

$$|||\widehat{T}|_{\mathcal{H}_0}|||_F = s_{\max}(\widehat{T}|_{\mathcal{H}_0})$$

$$s_1 = e^{-\gamma\Delta t/2} < 1$$



Solution 2

Bloch vector representation:

$$\begin{pmatrix} 1 \\ \langle\sigma^x\rangle_k \\ \langle\sigma^y\rangle_k \\ \langle\sigma^z\rangle_k \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \widehat{T}_{22} & 0 & 0 \\ \widehat{T}_{31} & 0 & \widehat{T}_{33} & \widehat{T}_{34} \\ \widehat{T}_{41} & 0 & \widehat{T}_{43} & \widehat{T}_{44} \end{pmatrix} \begin{pmatrix} 1 \\ \langle\sigma^x\rangle_{k-1} \\ \langle\sigma^y\rangle_{k-1} \\ \langle\sigma^z\rangle_{k-1} \end{pmatrix}$$

$\widehat{T}|_{\mathcal{H}_0}$

?

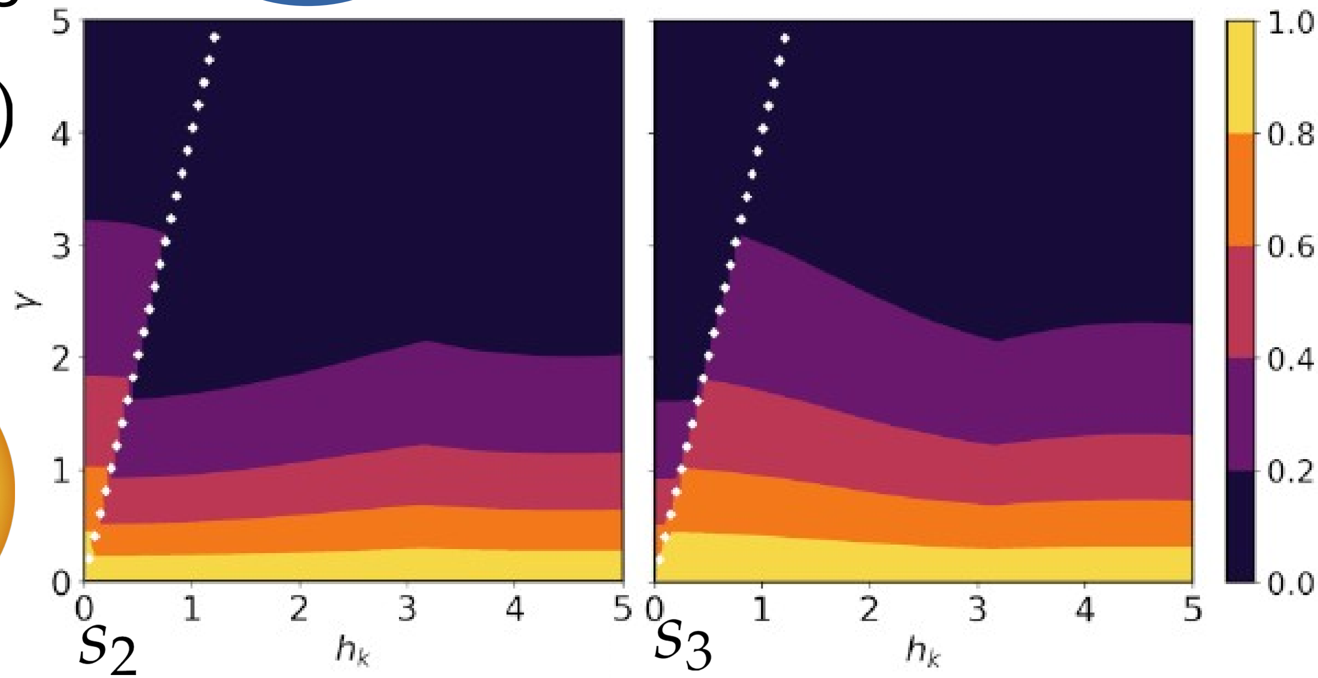
$$|||\widehat{T}(\mathbf{z}_k)|_{\mathcal{H}_0}||| < 1$$

Induced Frobenius norm:
maximum singular value

$$|||\widehat{T}|_{\mathcal{H}_0}|||_F = s_{\max}(\widehat{T}|_{\mathcal{H}_0})$$

$$s_1 = e^{-\gamma \Delta t/2} < 1$$

This system has fading memory!!



Solution 2:

$$\dot{\rho} = -i[H, \rho] + \gamma L \rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L, \rho\}$$

Input-dependent fixed point:

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^x \qquad L = \sigma^-$$

$$\rho^* = \frac{1}{\gamma^2 + 2h_k^2}\begin{pmatrix} h_k^2 & i\gamma h_k \\ -i\gamma h_k & \gamma^2 + h_k^2 \end{pmatrix}$$

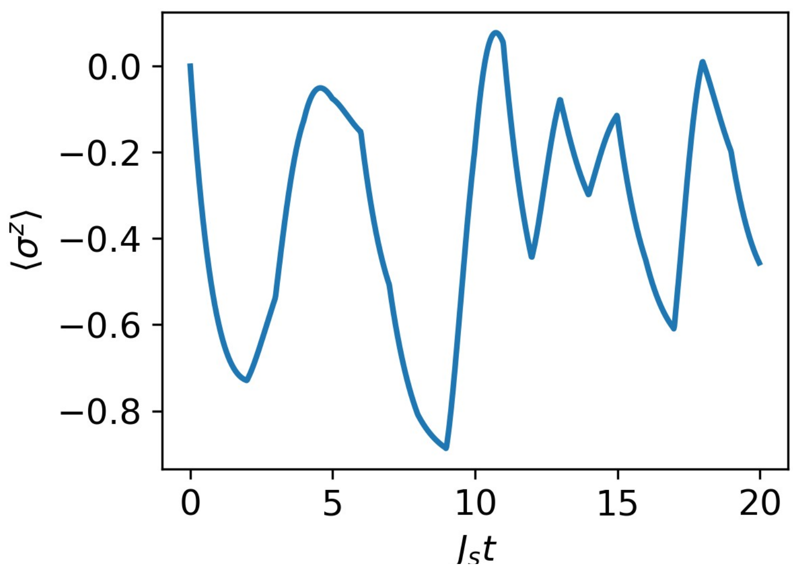
<u>Solution 2</u>:

$$\dot{\rho} = -i[H,\rho] + \gamma L \rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L, \rho\}$$

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^x \qquad L = \sigma^-$$

Input-dependent fixed point:

$$\rho^* = \frac{1}{\gamma^2 + 2h_k^2}\begin{pmatrix} h_k^2 & i\gamma h_k \\ -i\gamma h_k & \gamma^2 + h_k^2 \end{pmatrix}$$



Result 2 states that this system will be always **input-dependent**!

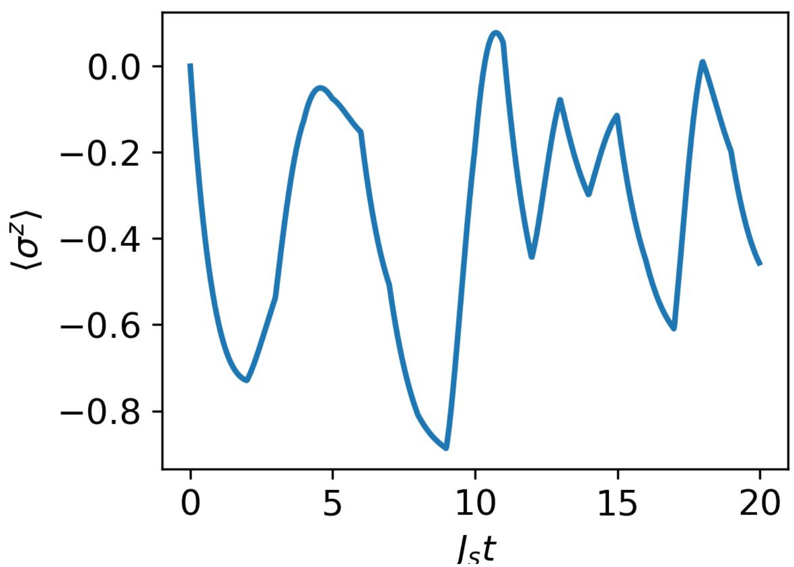
<u>Solution 2</u>:

$$\dot{\rho} = -i[H,\rho] + \gamma L \rho L^{\dagger} - \frac{\gamma}{2}\{L^{\dagger}L,\rho\}$$

$$H = \frac{h(\mathbf{z}_k)}{2}\sigma^x \qquad L = \sigma^-$$

Input-dependent fixed point:

$$\rho^* = \frac{1}{\gamma^2 + 2h_k^2}\begin{pmatrix} h_k^2 & i\gamma h_k \\ -i\gamma h_k & \gamma^2 + h_k^2 \end{pmatrix}$$

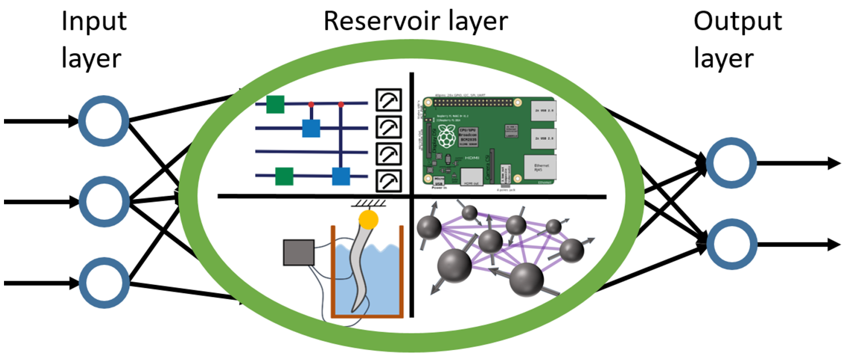


Result 2 states that this system will be always **input-dependent**!

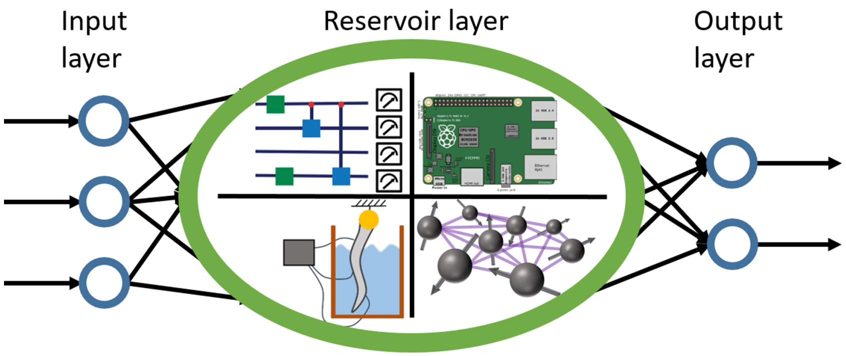
<u>Conclusion:</u> we **don't know how good** the reservoir is for specific tasks, **but it works!**

Conclusions

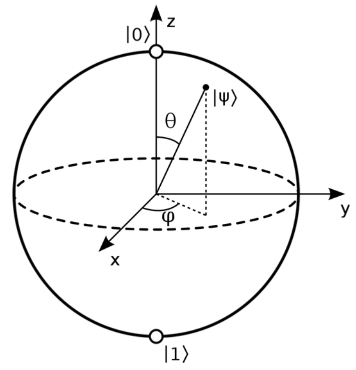
(Quantum) Reservoir Computing might be a good **alternative** to **conventional computers** to process **temporal** data



(Quantum) Reservoir Computing might be a good **alternative** to **conventional computers** to process **temporal** data

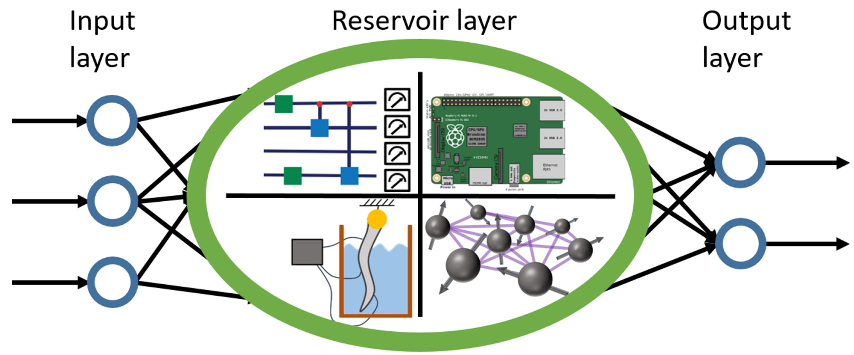


Different representations of quantum systems can provide a **better insight** of **reservoir computing properties**

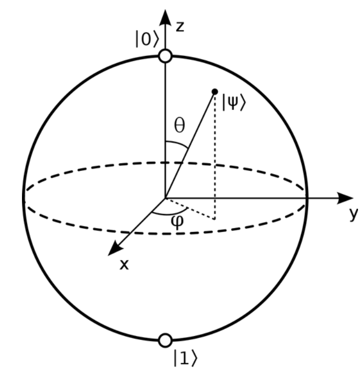


Conclusions

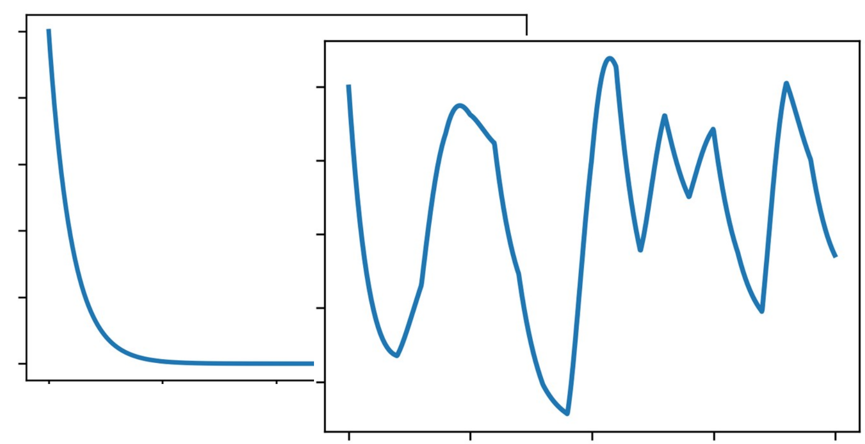
(Quantum) Reservoir Computing might be a good **alternative** to **conventional computers** to process **temporal** data



Different representations of quantum systems can provide a **better insight** of **reservoir computing properties**



A quantum reservoir is **useful if and only if** the dynamics **converges** towards **input-dependent** fixed points



Outlook & Thanks!

Studying infinite dimensional systems

Extending the theory to non-ideal situations: finite number of measurements, POVMs...

Finding the most general conditions for universal approximation property

R. Martínez-Peña, J. P. Ortega, *Quantum reservoir computing in finite dimensions*, Physical Review E 107 (3), 035306

IFISC

THANK YOU

for your attention

QuaReC
PRD2018/47

QuaResC
PID2019-109094GB-C21