Benchmarking the performance of quantum reservoir computing platforms of particles of distinct statistics

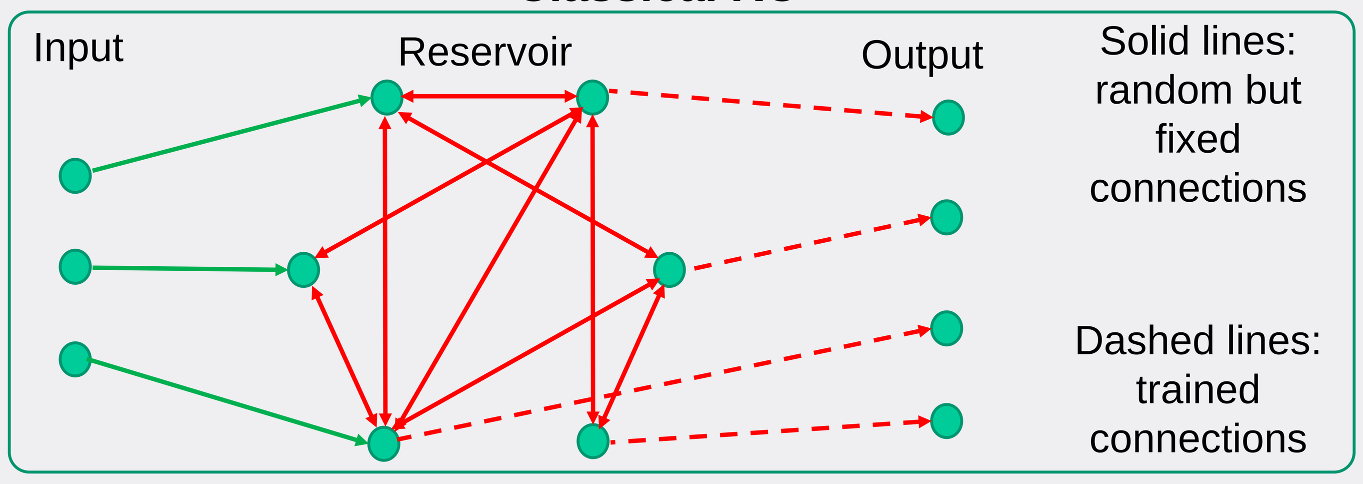
Christos Charalambous, Guillem Llodra, Rodrigo Martinez-Peña, Gian Luca Giorgi and Roberta Zambrini
IFISC (CSIC-UIB) Palma de Mallorca – Spain.
christos@ifisc.uib-csic.es

Abstract

Reservoir computing (RC) is a neuro-inspired machine learning approach to time series processing. As such, it forms an example of a natural unconventional analog computer designed to perform a given computational task. Its power in solving nonlinear and temporal tasks depends on the reservoir possessing a high dimensional state space and the ability to retain memory of information for sufficiently long time. Quantum systems, with their large number of degrees of freedom and their complex real time dynamics satisfy both requirements, and for this reason are good candidates to serve as substrates for RC. In addition, quantum effects such as superposition could lead to improvement in the performance of a RC. An important issue we explore here in order to establish the potential of quantum reservoirs computing (QRC) is the role of the particle statistics of the units composing the complex network reservoir. Considering the simplest interaction, we assess the performance of fermions bosons and the commonly used spins for QRC.

Introduction / Model

Classical RC

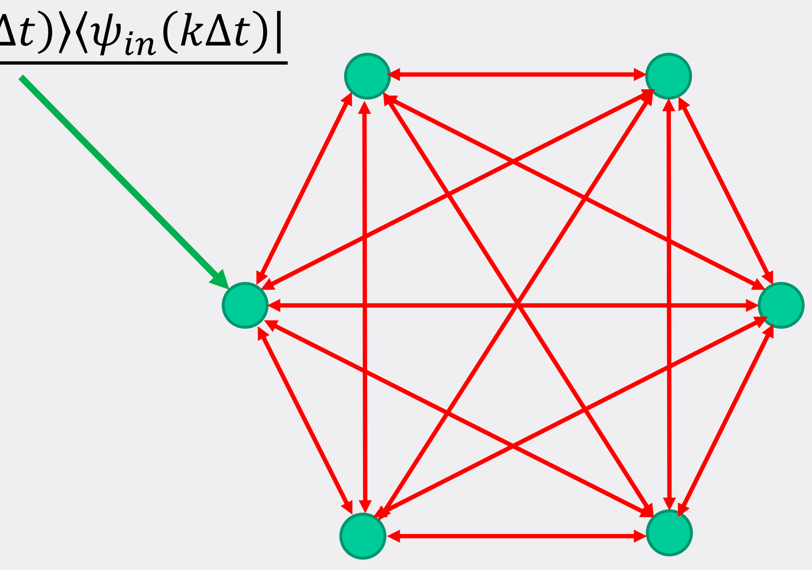


Solid lines: random but fixed connections

Dashed lines: trained connections

Advantages of RC:
1. Fast training
2. Easy to implement in practice
3. Possibility of solving many tasks with the same reservoir simply by training only the output layer independently for each task

$\rho_{in} = |\psi_{in}(k\Delta t)\rangle\langle\psi_{in}(k\Delta t)|$



Comments:
1. $J_{ij} \sim U[0,1]$
2. Input:

$$|\psi_{in}^{r}(k\Delta t)\rangle = \sqrt{u_k}|0\rangle + \sqrt{1-u_k}|r\rangle$$

where $u_k \in u = (u_0, \dots, u_L)$ with $u_k \sim U[0,1]$ is the classical input and $r = \begin{cases} 1 & fermions \\ 1 \text{ or } 2 & bosons \end{cases}$

3. Input nodes: subset of physical nodes
4. Output nodes: All of physical nodes

Algorithm:
1. Measurement every Δt: $x_k = Tr[O_j \rho(k\Delta t)]$ obtained using all particles and observables O_j $\rightarrow \; y_k^{out} = W_{out} f(x_k)$
2. State of reservoir prepared after measurement: $\tilde{\rho}(k\Delta t) = \rho_{in} \otimes Tr_1[\rho((k-1)\Delta t)]$
3. Unitary evolution during time Δt $\rho(k\Delta t) = e^{-i\Delta t H}[\tilde{\rho}(k\Delta t)]e^{i\Delta t H}$

Train network (regression $\min_{W}(y_k^{out} - y_k)^2$) to be able to reproduce a target function of the input y_k, e.g. $y_k = u_{k-delay}$

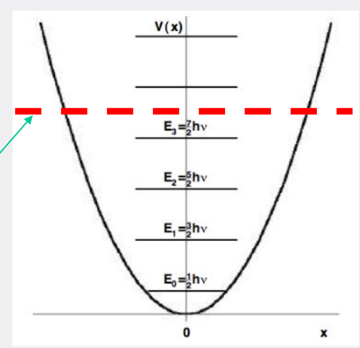
Goal: Evaluate its performance on reproducing y_k using a new set of outputs y_k^{out} obtained for new input numbers u_k $\quad PearsonCorr = \frac{Cov^2(y_k^{out}, y_k)}{\sigma^2(y_k^{out})\sigma^2(y_k)}$

Normal Hamiltonian $\quad H = \sum_{i,j} J_{ij}\, a_i^{\dagger} a_j$

Fermions: $a_i = \underbrace{\sigma^z \otimes \sigma^z \otimes \dots \otimes \sigma^z}_{i-1} \otimes \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \otimes \underbrace{I \otimes I \otimes \dots \otimes I}_{N-i}$

Bosons: $a_i = \underbrace{I \otimes I \otimes \dots \otimes I}_{i-1} \otimes \hat{a}_i \otimes \underbrace{I \otimes I \otimes \dots \otimes I}_{N-i}$

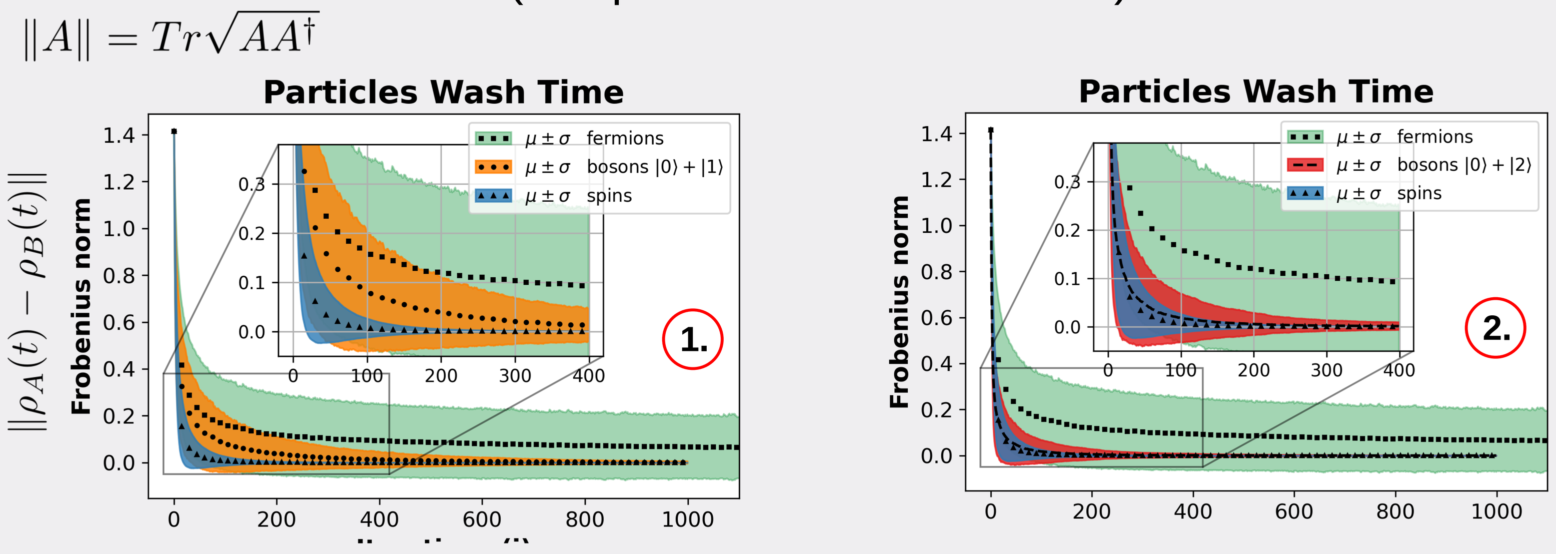
where: $\hat{a}_i = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & \cdots \\ \sqrt{1} & 0 & 0 & 0 & 0 & \cdots \\ 0 & \sqrt{2} & 0 & 0 & 0 & \cdots \\ 0 & 0 & \sqrt{3} & 0 & 0 & \cdots \\ 0 & 0 & 0 & \sqrt{4} & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{pmatrix}$



Energy spectrum cutoff for bosons at n_{cutoff} level

Echo state property (independence on initial state)

$\|A\| = Tr\sqrt{AA^{\dagger}}$



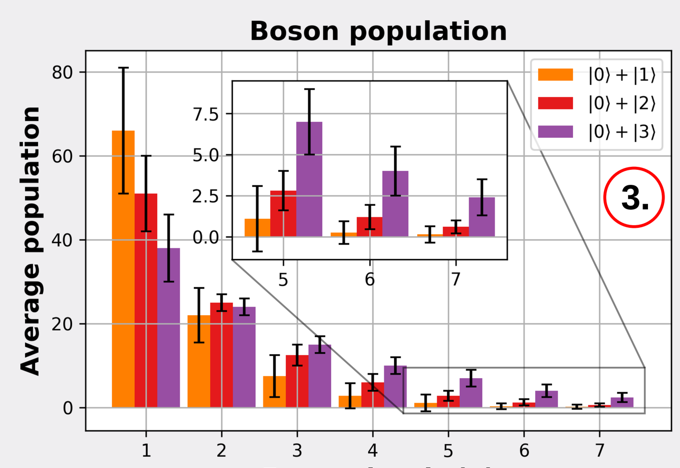
$$\rho_A(0) = \underbrace{|0\rangle\langle 0| \otimes |0\rangle\langle 0| \otimes \dots \otimes |0\rangle\langle 0|}_{N}$$

$$\rho_B(0) = \sum_{i,j=1}^{n^N} K_{ij}\,|i\rangle\langle j| \qquad K_{ij} \sim Uniform[0,1]$$

Check boson's levels' occupation

$P_i^j = tr\left(\rho_t \hat{P}_i^j\right)$ where:

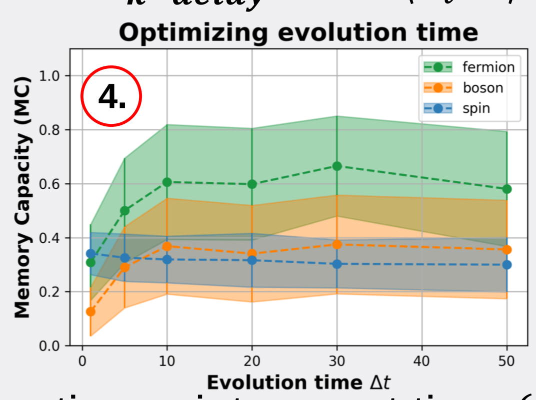
$\hat{P}_i^j = \underbrace{I \otimes \dots \otimes I}_{i-1} \otimes |j\rangle\langle j| \otimes \dots \otimes I$



$P_i^0 \gg P_i^{n_{cutoff}} \quad \forall\, i \in \{1, \dots, N\}$

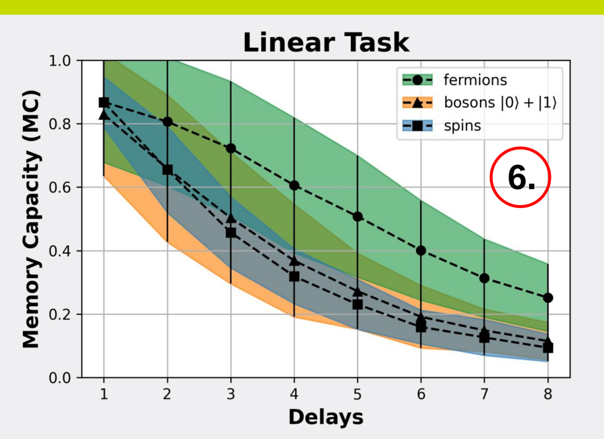
Results

$y_k = u_{k-delay}^{q}$, $O_i = \langle a_i^{\dagger} a_i \rangle$

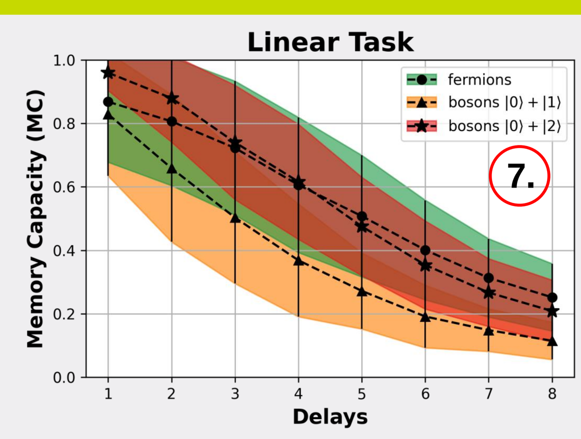


$q = 1$

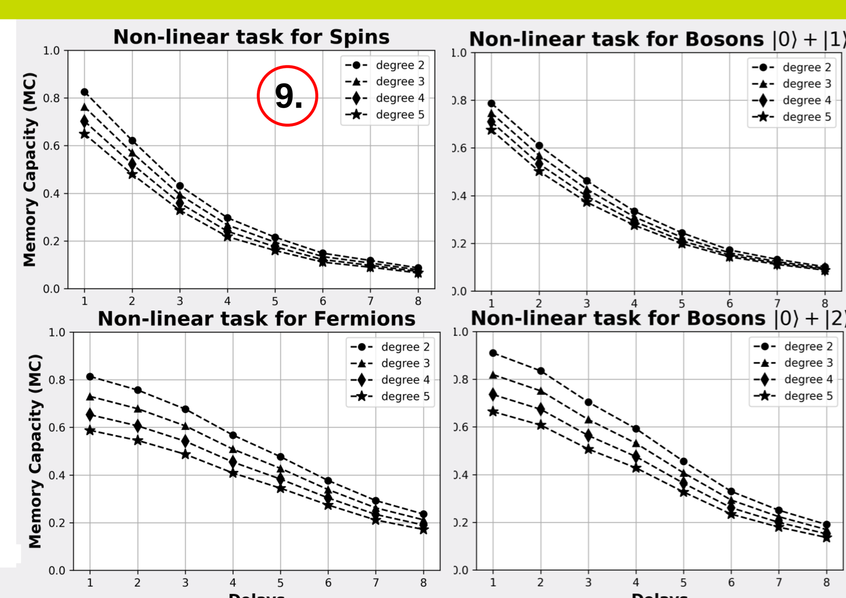
optimum interevent time (Δt)



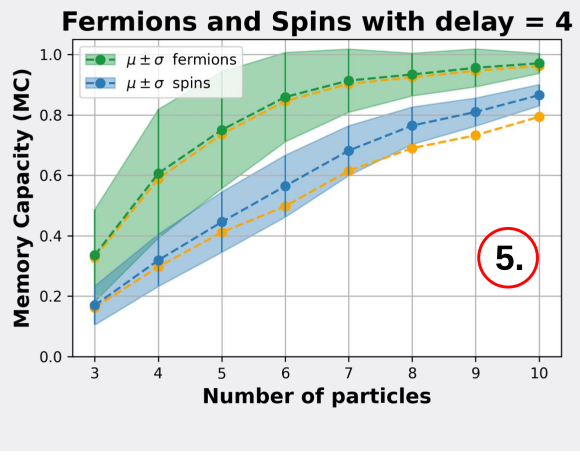
Bosons with input in 1[st] excited state perform similarly to spins



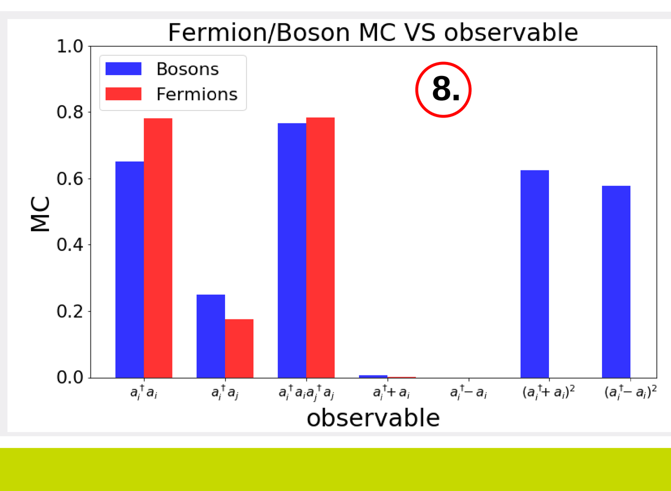
Bosons with input in 2[nd] excited state perform similarly to fermions



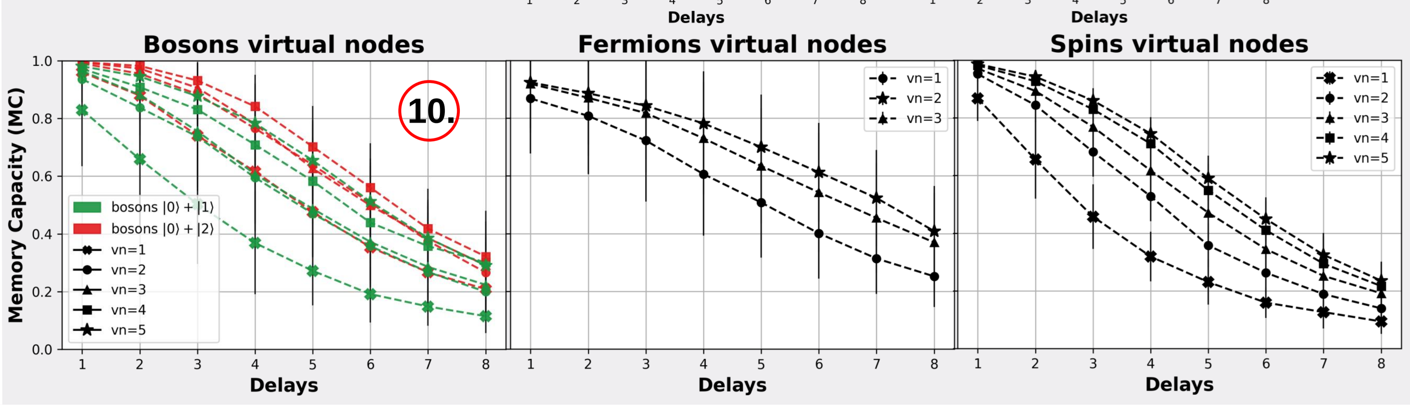
1) performance decreases for more nonlinear tasks.
2) For small (large) delays bosons $|\psi_{in}^{2}(k\Delta t)\rangle$ (fermions) outperform fermions (bosons).



Performance improves with the number of particles



The performance of QRC using different observables in the output is examined.



The introduction of virtual nodes improves the performance

Conclusions

- Performance improves up to a saturation level, as the interevent time increases for bosons and fermions
- Fermions and spins perform better as the number of particle increases
- The performance decreases for longer memory tasks. For the input state $|\psi_{in}^{1}(k\Delta t)\rangle$ bosons perform as good as spins. For the input state $|\psi_{in}^{2}(k\Delta t)\rangle$ they perform as good as fermions. The increase in the performance observed is attributed to the utilization of the larger Hilbert space of bosons.
- We studied the performance of bosons and spins for various observables.
- We saw that the performance decreases for more non-linear tasks. Fermions performance is more robust as delay increases.
- Virtual nodes improve the performance in all cases.