Identifying network topologies via quantum walk distributions

Genetic algorithms & Machine learning

Claudia Benedetti

University of Milan

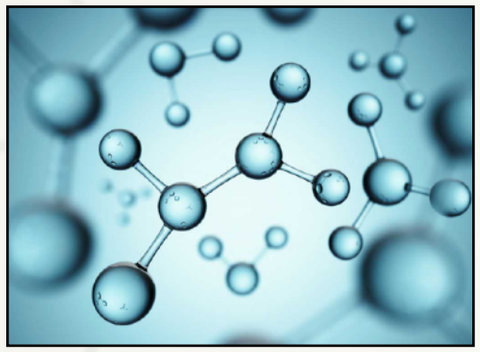
In collab with Ilaria Gianani (University of Roma Tre)

AVS Quantum Sci. 5, 014405 (2023)

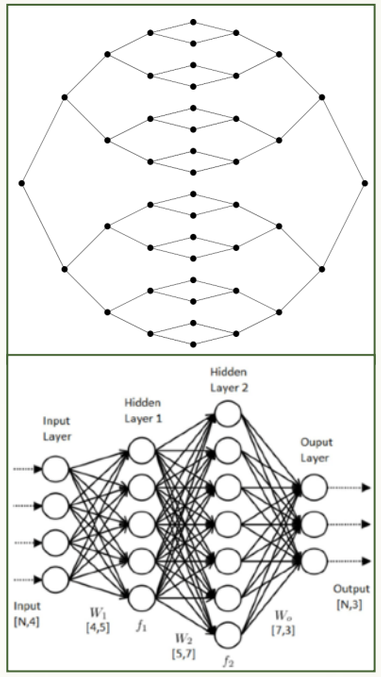
AVS Quantum Sci. 6, 014412 (2024)

Context and motivations

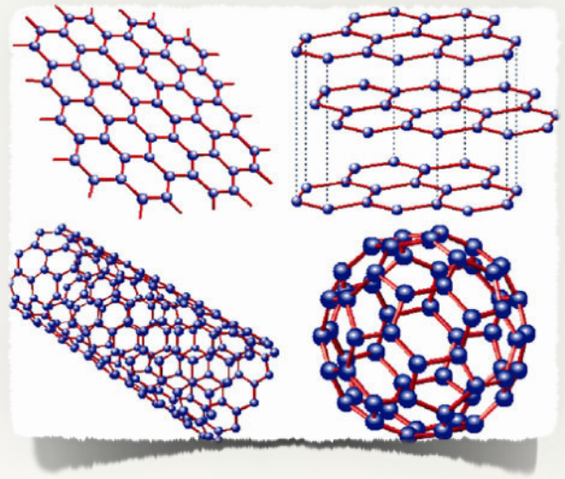
Different topologies



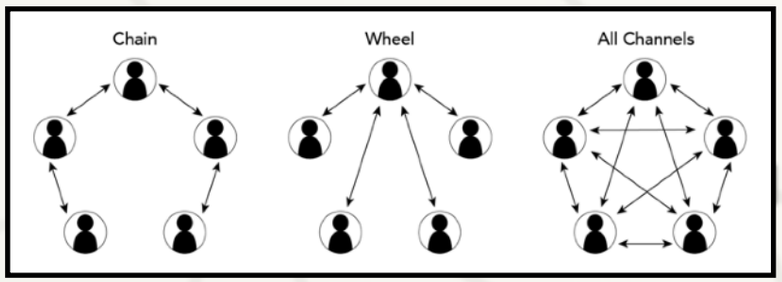
interactions



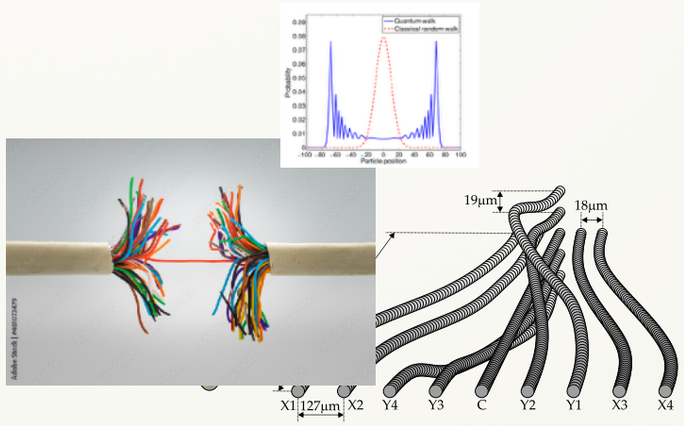
algorithms



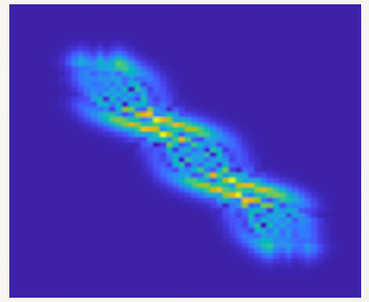
Transport properties



Communication



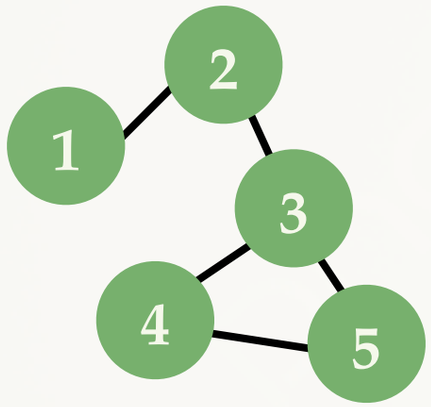
imperfections, defects



State preparation

Quantum walks on graphs

Graph $G(V, E)$ ⟷ Network



$$H = \begin{pmatrix} 0 & J_{12} & 0 & 0 & 0 \\ J_{12} & 0 & J_{23} & 0 & 0 \\ 0 & J_{23} & 0 & J_{34} & J_{35} \\ 0 & 0 & J_{34} & 0 & J_{45} \\ 0 & 0 & J_{35} & J_{45} & 0 \end{pmatrix}$$

Adjacency matrix: A

$$A_{jk} = \begin{cases} 1 \text{ if } (j,k) \in E(G) \\ 0 \text{ otherwise} \end{cases}$$

$$H_{yk} = -J_{yk} A_{yk}$$

$$|\psi(t)\rangle = e^{-iHt} |\psi_0\rangle$$

$$p(v,t) = |\langle v|\psi(t)\rangle|^2 = |\langle v| e^{-iHt} |\psi_0\rangle|^2$$

The task:

The QW evolution:

The task:

The QW evolution:

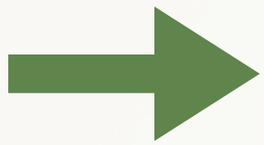
Our aim:

non-linear mapping between the H and the probability distributions

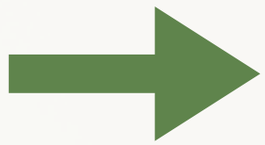
How?

$$|\langle v|\psi(t)\rangle|^2$$

$$|\psi_0\rangle$$



QW evolution??



$$H = ?$$

$$H = \begin{pmatrix} 0 & J_{12} & 0 & 0 & 0 \\ J_{12} & 0 & J_{23} & 0 & 0 \\ 0 & J_{23} & 0 & J_{34} & J_{35} \\ 0 & 0 & J_{34} & 0 & J_{45} \\ 0 & 0 & J_{35} & J_{45} & 0 \end{pmatrix}$$

How do we do this?

$$J_{yk} = 0,1$$

How?

$$|\langle v|\psi(t)\rangle|^2$$

$$|\psi_0\rangle$$

QW evolution??

$$H = ?$$

$$H = \begin{pmatrix} 0 & J_{12} & 0 & 0 & 0 \\ J_{12} & 0 & J_{23} & 0 & 0 \\ 0 & J_{23} & 0 & J_{34} & J_{35} \\ 0 & 0 & J_{34} & 0 & J_{45} \\ 0 & 0 & J_{35} & J_{45} & 0 \end{pmatrix}$$

How do we do this?

Determine the topology, with fixed J values

$$J_{yk} = 0,1$$

Search problem
Genetic algorithm

How?

$$|\langle v|\psi(t)\rangle|^2$$

$$|\psi_0\rangle$$

QW evolution??

$H = ?$

$$H = \begin{pmatrix} 0 & J_{12} & 0 & 0 & 0 \\ J_{12} & 0 & J_{23} & 0 & 0 \\ 0 & J_{23} & 0 & J_{34} & J_{35} \\ 0 & 0 & J_{34} & 0 & J_{45} \\ 0 & 0 & J_{35} & J_{45} & 0 \end{pmatrix}$$

How do we do this?

Determine the topology, with fixed J values

$$J_{yk} = 0,1$$

Search problem
Genetic algorithm

Coupling estimation, at fixed topology

$$J_{yk} \in [0,1]$$

Multiparameter estimation
Neural network

How?

$$|\langle v|\psi(t)\rangle|^2$$

$$|\psi_0\rangle$$

QW evolution??

$H = ?$

$$H = \begin{pmatrix} 0 & J_{12} & 0 & 0 & 0 \\ J_{12} & 0 & J_{23} & 0 & 0 \\ 0 & J_{23} & 0 & J_{34} & J_{35} \\ 0 & 0 & J_{34} & 0 & J_{45} \\ 0 & 0 & J_{35} & J_{45} & 0 \end{pmatrix}$$

How do we do this?

Determine the topology, with fixed J values

$$J_{yk} = 0,1$$

Search problem
Genetic algorithm

Coupling estimation, at fixed topology

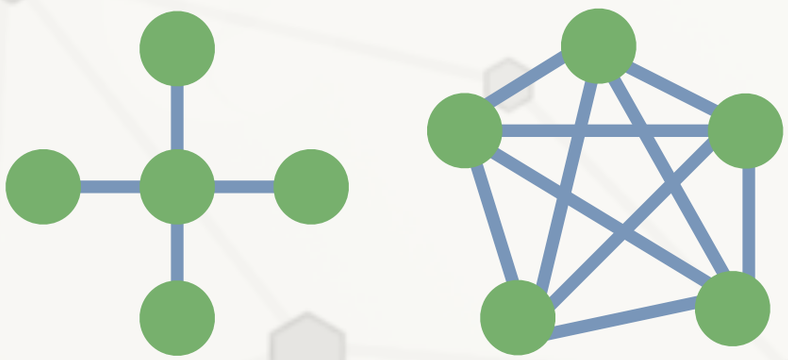
$$J_{yk} \in [0,1]$$

Multiparameter estimation
Neural network

Task1: Identifying topologies

The problem: Given an undirected graph of N sites, retrieve the adjacency matrix of the graph having access only to the initial state of the walker and probability distributions $p(v, t_k)$ over the nodes at times t_k.

The coupling values are fixed $J_{yk} = 0,1$



Task1: Identifying topologies

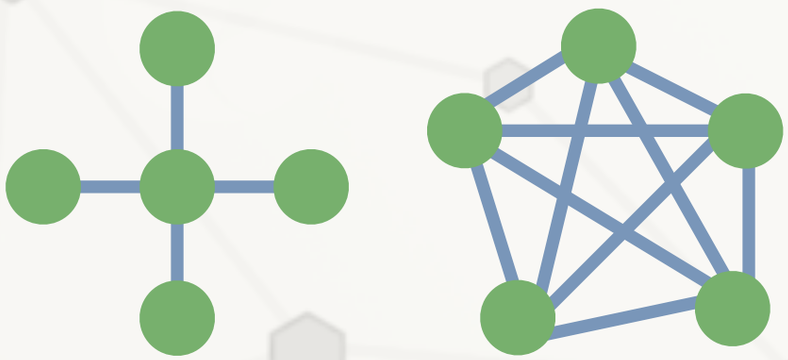
The problem: Given an undirected graph of N sites, retrieve the adjacency matrix of the graph having access only to the initial state of the walker and probability distributions $p(v, t_k)$ over the nodes at times t_k.

The coupling values are fixed $J_{yk} = 0,1$

Hamiltonian H

↓

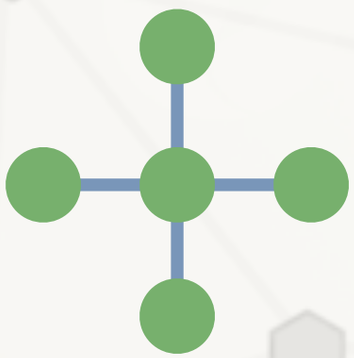
Adjacency matrix: A

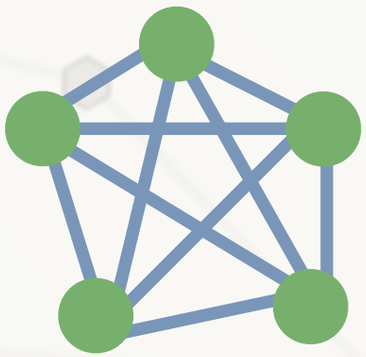


Task1: Identifying topologies

The problem: Given an undirected graph of N sites, retrieve the adjacency matrix of the graph having access only to the initial state of the walker and probability distributions $p(v, t_k)$ over the nodes at times t_k.

The coupling values are fixed $J_{yk} = 0,1$





Hamiltonian H

Adjacency matrix: A

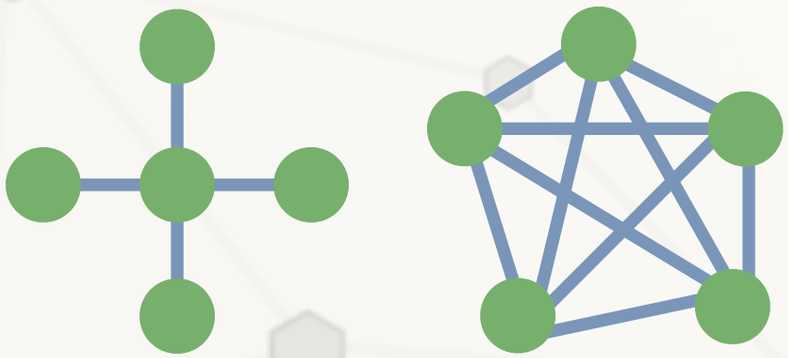
$$H = \begin{pmatrix} 0 & J_{12} & J_{13} & J_{14} & J_{15} \\ J_{12} & 0 & J_{23} & J_{24} & J_{25} \\ J_{13} & J_{23} & 0 & J_{34} & J_{35} \\ J_{14} & J_{24} & J_{34} & 0 & J_{45} \\ J_{15} & J_{25} & J_{35} & J_{45} & 0 \end{pmatrix}$$

Real Symmetric

Task1: Identifying topologies

The problem: Given an undirected graph of N sites, retrieve the adjacency matrix of the graph having access only to the initial state of the walker and probability distributions $p(v, t_k)$ over the nodes at times t_k.

The coupling values are fixed $J_{yk} = 0,1$



Hamiltonian H

$$H = \begin{pmatrix} 0 & J_{12} & J_{13} & J_{14} & J_{15} \\ J_{12} & 0 & J_{23} & J_{24} & J_{25} \\ J_{13} & J_{23} & 0 & J_{34} & J_{35} \\ J_{14} & J_{24} & J_{34} & 0 & J_{45} \\ J_{15} & J_{25} & J_{35} & J_{45} & 0 \end{pmatrix}$$

Adjacency matrix: A

Real Symmetric

$$\Lambda = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots, J_{45}] = [0,0,1,1,\ldots,1,0]$$

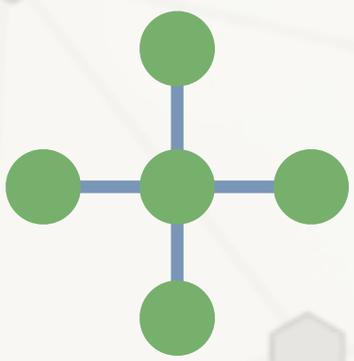
Determining the topology is equivalent to retrieving a binary string S of length $n_c = N(N-1)/2$

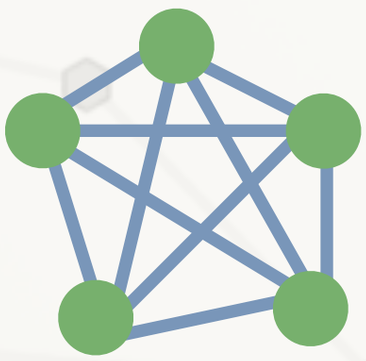
Task1: Identifying topologies

The problem: Given an undirected graph of N sites, retrieve the adjacency matrix of the graph having access only to the initial state of the walker and probability distributions $p(v, t_k)$ over the nodes at times t_k.

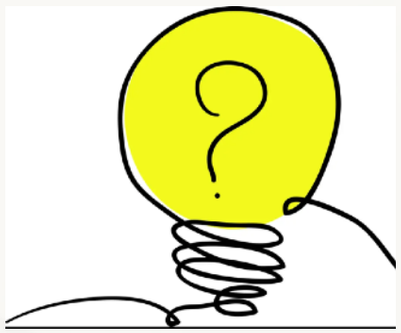
The coupling values are fixed $J_{yk} = 0,1$

$$\Lambda = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots] = [0,0,1,1,\ldots,1,0]$$





Search problem

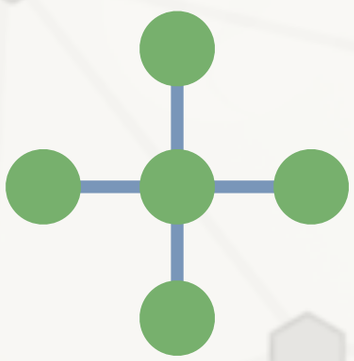


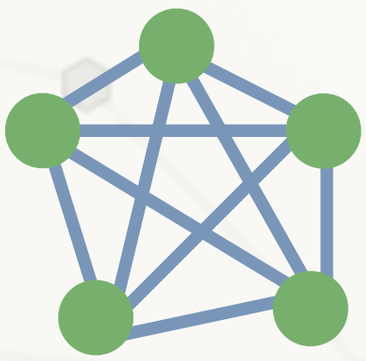
Task1: Identifying topologies

The problem: Given an undirected graph of N sites, retrieve the adjacency matrix of the graph having access only to the initial state of the walker and probability distributions $p(v, t_k)$ over the nodes at times t_k.

The coupling values are fixed $J_{yk} = 0,1$

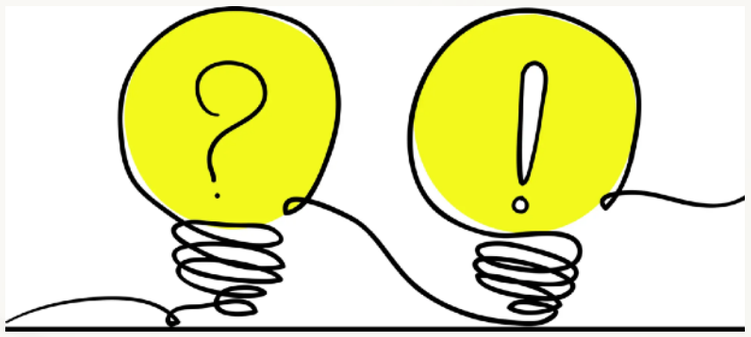
$$\Lambda = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots] = [0,0,1,1,\ldots,1,0]$$





Search problem

Genetic algorithm



The genetic algorithm

$$\Sigma = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots] \qquad (J_{jk} = 0,1)$$

GAs inspired by biological evolution to find optimal solutions to problems.
They mimic the way species evolve over generations to adapt to their environments

The genetic algorithm

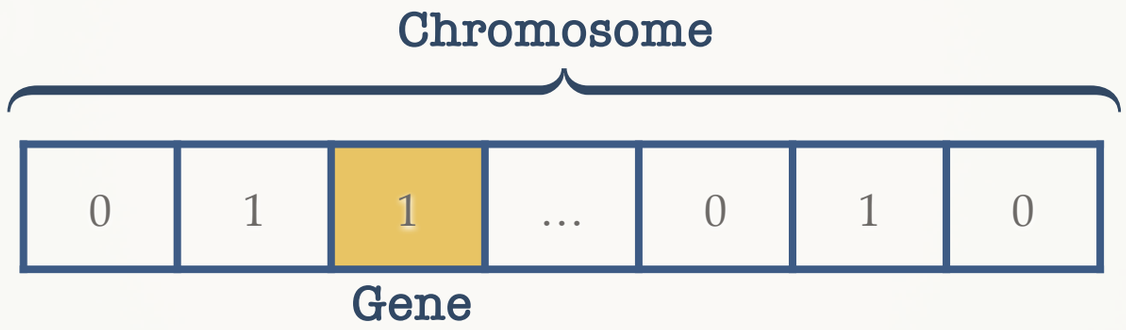
$$\Sigma = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots] \qquad (J_{jk} = 0,1)$$ → Chromosome, made of genes

GAs inspired by biological evolution to find optimal solutions to problems.
They mimic the way species evolve over generations to adapt to their environments

The genetic algorithm

$$\Sigma = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots] \quad (J_{jk} = 0,1)$$ → Chromosome, made of genes

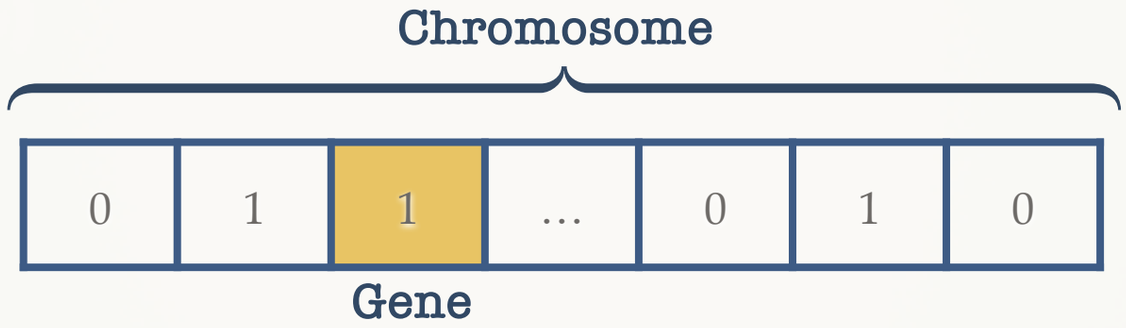
GAs inspired by biological evolution to find optimal solutions to problems.
They mimic the way species evolve over generations to adapt to their environments



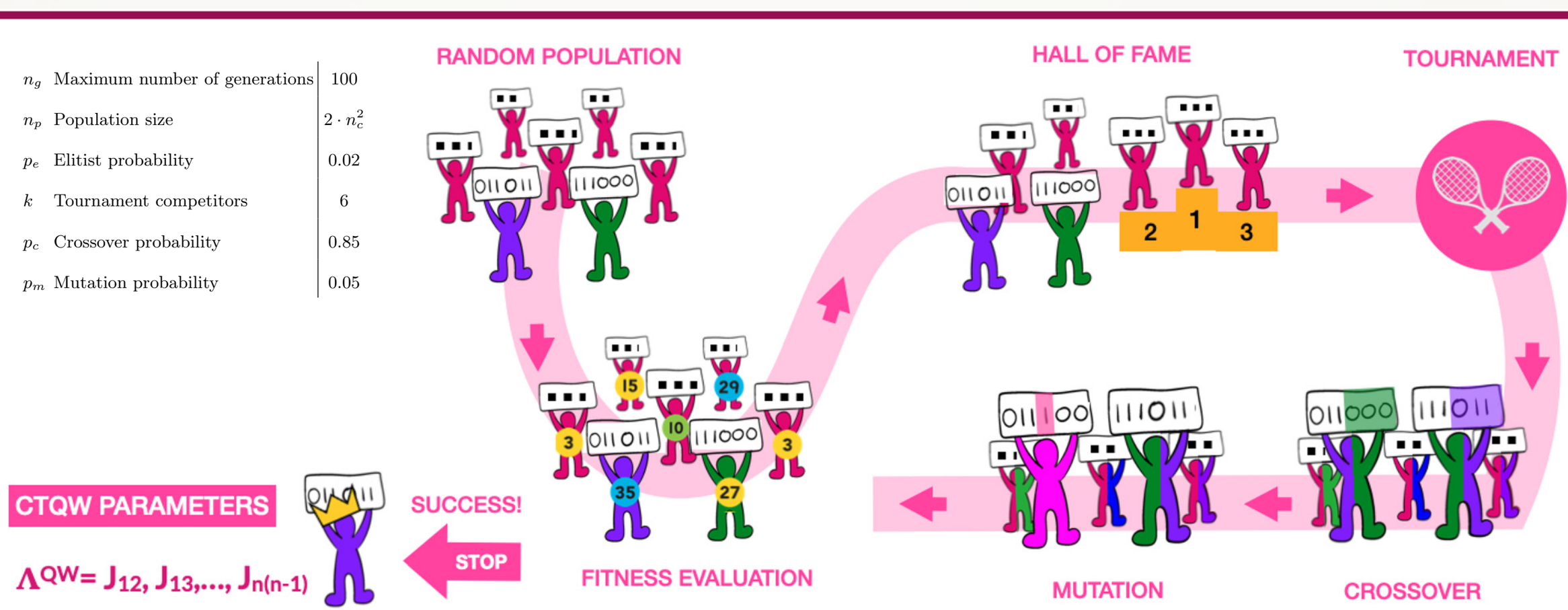
The genetic algorithm

$$\Sigma = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots] \quad (J_{jk} = 0,1)$$ → Chromosome, made of genes

GAs inspired by biological evolution to find optimal solutions to problems.
They mimic the way species evolve over generations to adapt to their environments



n_g	Maximum number of generations	100
n_p	Population size	$2 \cdot n_c^2$
p_e	Elitist probability	0.02
k	Tournament competitors	6
p_c	Crossover probability	0.85
p_m	Mutation probability	0.05



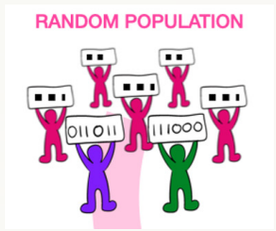
Genetic operations

$$\Sigma = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots] \qquad (J_{jk} = 0,1)$$ → **Chromosome, made of genes**

Genetic operations

$$\Sigma = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots] \qquad (J_{jk} = 0,1)$$ → Chromosome, made of genes

1. Generate the 0^{th} generation of chromosomes $\{\Lambda_i\}_{n_p}$



Genetic operations

$$\Sigma = [J_{12}, J_{13}, \ldots J_{23}, J_{24} \ldots] \qquad (J_{jk} = 0,1)$$ → Chromosome, made of genes

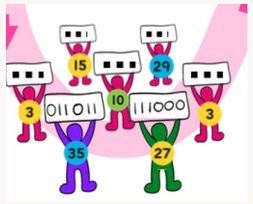
1. Generate the 0^{th} generation of chromosomes $\{\Lambda_i\}_{n_p}$

0	1	1	…	1	1
0	0	1	…	0	1
1	0	1	…	1	0
0	1	0	…	0	1

n_p

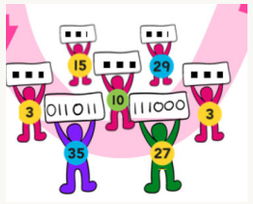
Genetic operations

2. Evaluate the scores S_i, according to the **fitness** function



Genetic operations

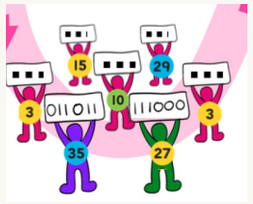
2. Evaluate the scores S_i, according to the **fitness** function



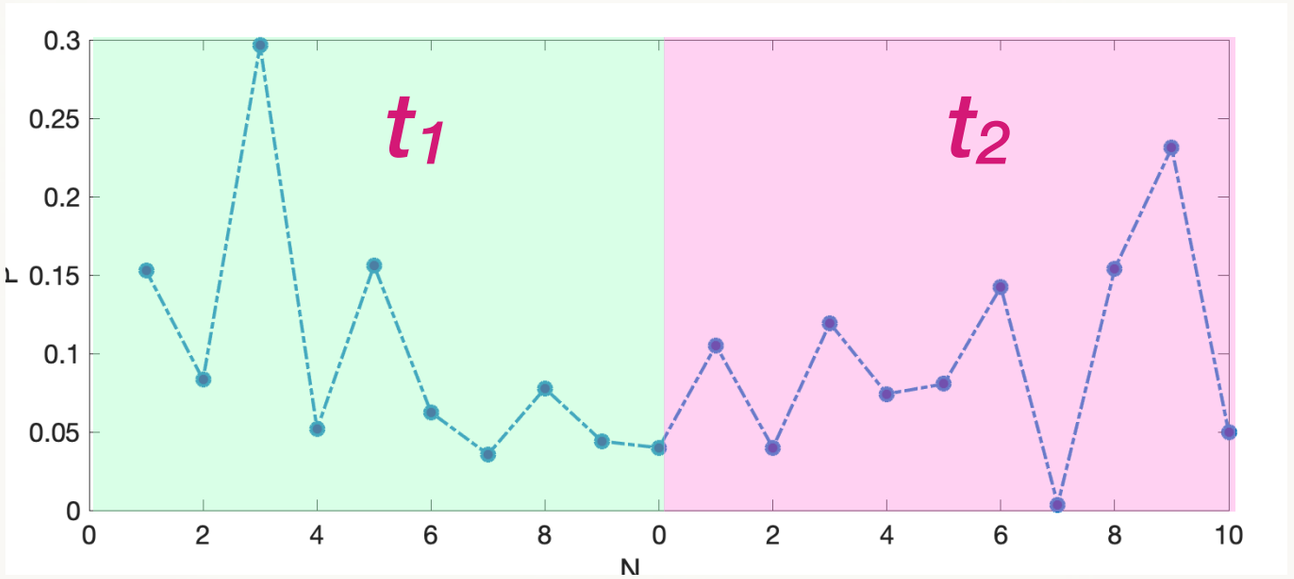
$\overline{\pi}_m(t)$ Measured probability distribution

Genetic operations

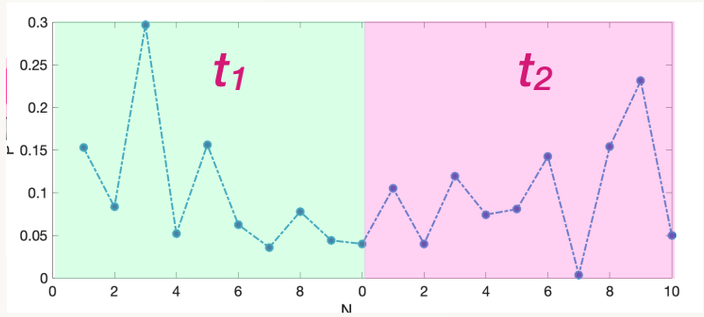
2. Evaluate the scores S_i, according to the **fitness** function



$\overline{\pi}_m(t)$ Measured probability distribution



Genetic operations

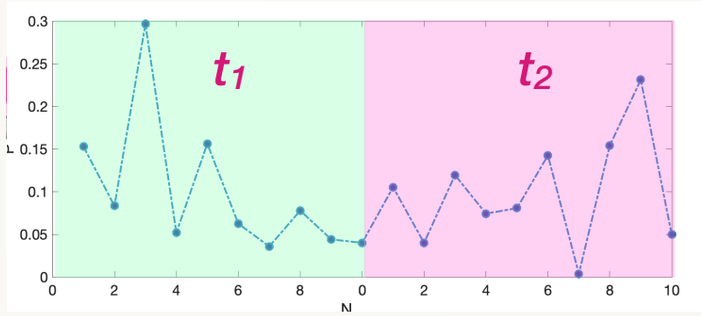


2. Evaluate the scores S_i, according to the **fitness** function

$\overline{\pi}_m(t)$ Measured probability distribution

Genetic operations

2. Evaluate the scores S_i, according to the **fitness** function

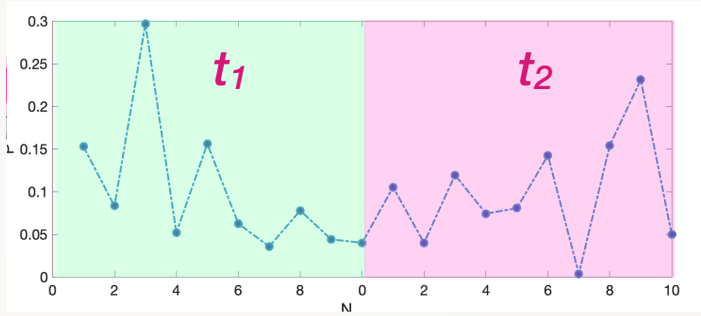


$\overline{\pi}_m(t)$ Measured probability distribution

$$\pi_m = \text{concatenate}\left[\overline{\pi}_m(t_1), \overline{\pi}_m(t_2)\right]$$

Genetic operations

2. Evaluate the scores S_i, according to the fitness function



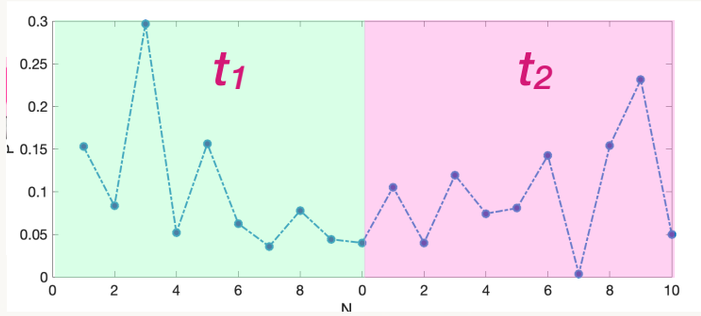
$\overline{\pi}_m(t)$ Measured probability distribution

$$\pi_m = \text{concatenate}\left[\overline{\pi}_m(t_1), \overline{\pi}_m(t_2)\right]$$

For each individual i, evolve the QW from $|\psi_0\rangle$ using the generated couplings Λ_i

Genetic operations

2. Evaluate the scores S_i, according to the **fitness** function



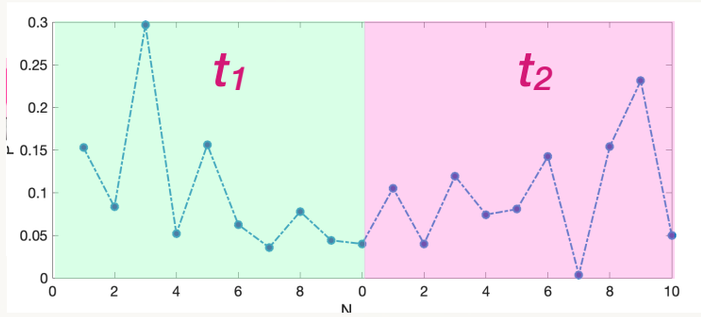
$\overline{\pi}_m(t)$ Measured probability distribution

$$\pi_m = \text{concatenate}\left[\overline{\pi}_m(t_1), \overline{\pi}_m(t_2)\right]$$

For each individual i, evolve the QW from $|\psi_0\rangle$ using the generated couplings Λ_i

$$\overline{\pi}_g^{(i)}(t_k) = \left\{ \left| \langle x | e^{-iH(\Lambda_i)t_k} | \psi_0 \rangle \rangle \right|^2 \right\}_{x=1}^{N} \qquad \pi_g^{(i)} = \text{concatenate}\left[\overline{\pi}_g^{(i)}(t_1), \overline{\pi}_g^{(i)}(t_2)\right]$$

Genetic operations



2. Evaluate the scores S_i, according to the fitness function

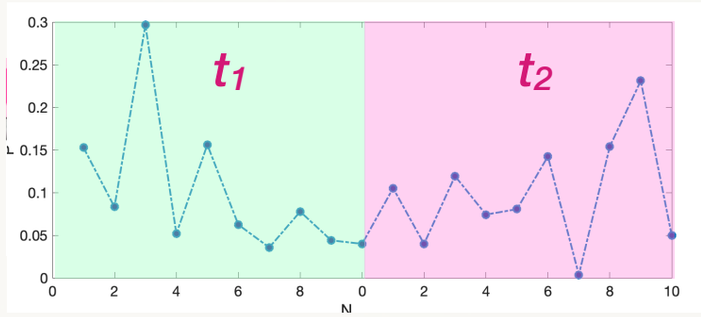
$\overline{\pi}_m(t)$ Measured probability distribution $\quad \pi_m = \text{concatenate}\left[\overline{\pi}_m(t_1), \overline{\pi}_m(t_2)\right]$

For each individual i, evolve the QW from $|\psi_0\rangle$ using the generated couplings Λ_i

$$\overline{\pi}_g^{(i)}(t_k) = \left\{ \left| \langle x | e^{-iH(\Lambda_i)t_k} | \psi_0 \rangle \rangle \right|^2 \right\}_{x=1}^{N} \qquad \pi_g^{(i)} = \text{concatenate}\left[\overline{\pi}_g^{(i)}(t_1), \overline{\pi}_g^{(i)}(t_2)\right]$$

The fitness of each individual is calculated as the "distance" between probability distributions.
Ex: Kullback-Leibler divergence (KLD); Kolmogorov distance (KD); etc..

Genetic operations



2. Evaluate the scores S_i, according to the fitness function

$\overline{\pi}_m(t)$ Measured probability distribution

$$\pi_m = \text{concatenate}\left[\overline{\pi}_m(t_1), \overline{\pi}_m(t_2)\right]$$

For each individual i, evolve the QW from $|\psi_0\rangle$ using the generated couplings Λ_i

$$\overline{\pi}_g^{(i)}(t_k) = \left\{ \left| \langle x | e^{-iH(\Lambda_i)t_k} | \psi_0 \rangle \rangle \right|^2 \right\}_{x=1}^{N} \qquad \pi_g^{(i)} = \text{concatenate}\left[\overline{\pi}_g^{(i)}(t_1), \overline{\pi}_g^{(i)}(t_2)\right]$$

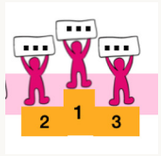
The fitness of each individual is calculated as the "distance" between probability distributions.
Ex: Kullback-Leibler divergence (KLD); Kolmogorov distance (KD); etc..

$$\mathcal{F}^{(i)} = \sum_x \pi_g(\Lambda_i) \log \frac{\pi_m}{\pi_g^{(i)}(\Lambda_i)} \qquad \mathcal{F} = \frac{1}{2} \sum_x \left| \pi_g(\Lambda_i) - \pi_m) \right|$$

The more fit an individual, the smaller its fitness score.

Genetic operations

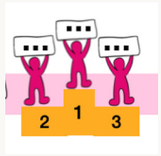
3. Hall of fame



0	1	1	…	1	1
0	0	1	…	0	1
1	0	1	…	1	0
0	1	0	…	0	1

Genetic operations

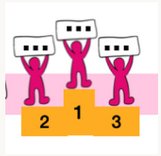
3. Hall of fame

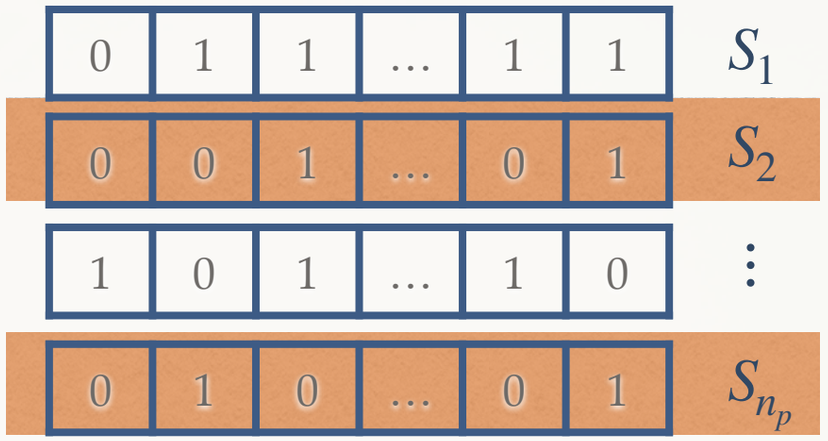


0	1	1	…	1	1	S_1
0	0	1	…	0	1	S_2
1	0	1	…	1	0	⋮
0	1	0	…	0	1	S_{n_p}

Genetic operations

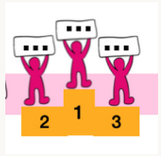
3. Hall of fame

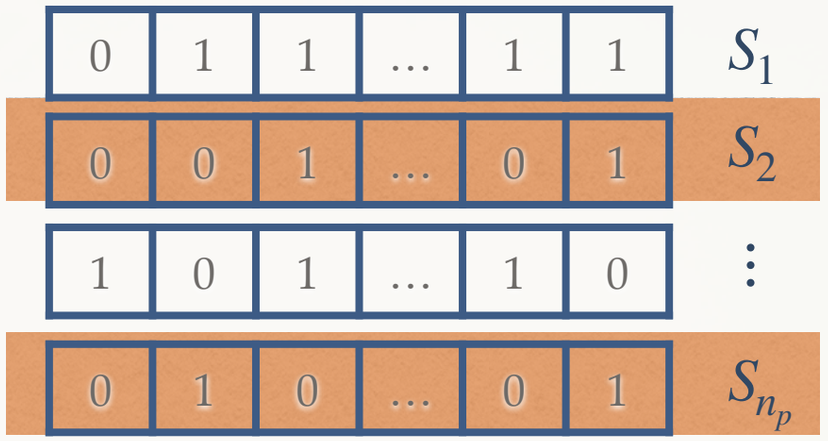




Genetic operations

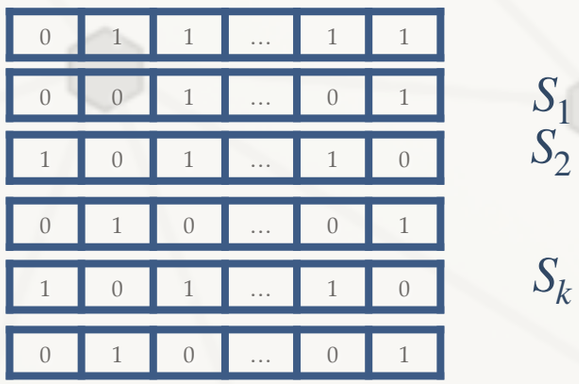
3. Hall of fame

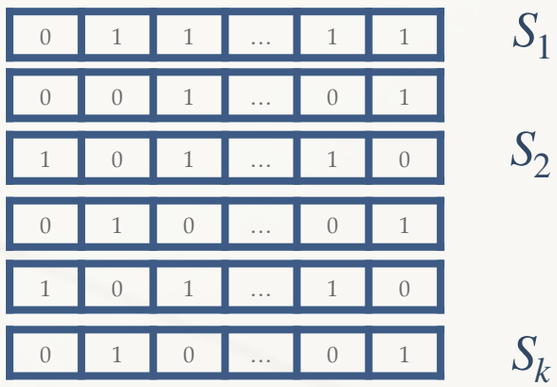




4. Tournament selection

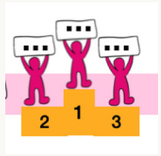
During each tournament, k individuals at random are selected among the whole population. The fittest one among the k (e.g. that with the smallest KLD) is chosen as a parent.

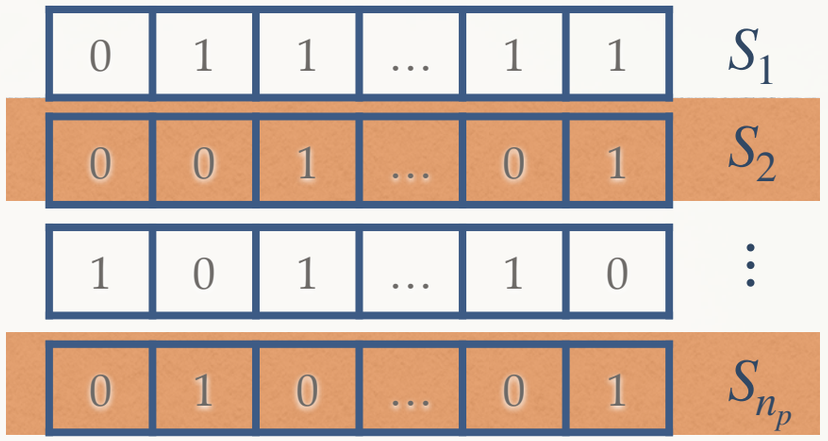




Genetic operations

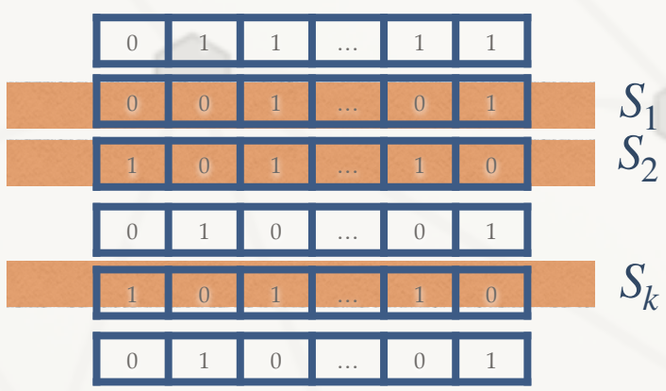
3. Hall of fame

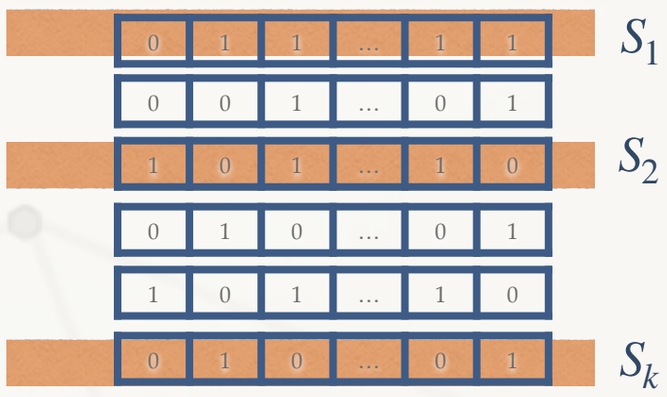




4. Tournament selection

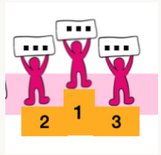
During each tournament, k individuals at random are selected among the whole population. The fittest one among the k (e.g. that with the smallest KLD) is chosen as a parent.

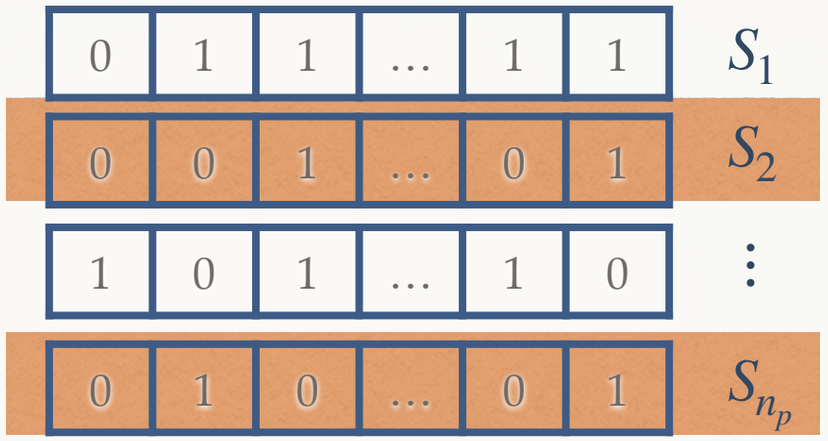




Genetic operations

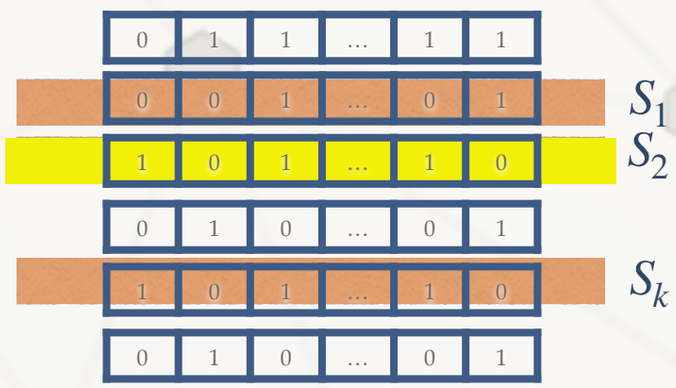
3. Hall of fame





4. Tournament selection

During each tournament, k individuals at random are selected among the whole population. The fittest one among the k (e.g. that with the smallest KLD) is chosen as a parent.



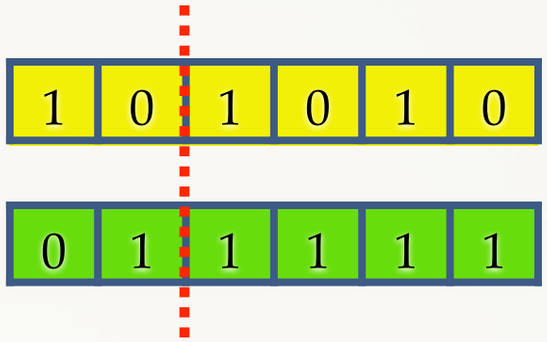
Genetic operations

1	0	1	0	1	0

0	1	1	1	1	1

5. Crossover to create two children, with p_c

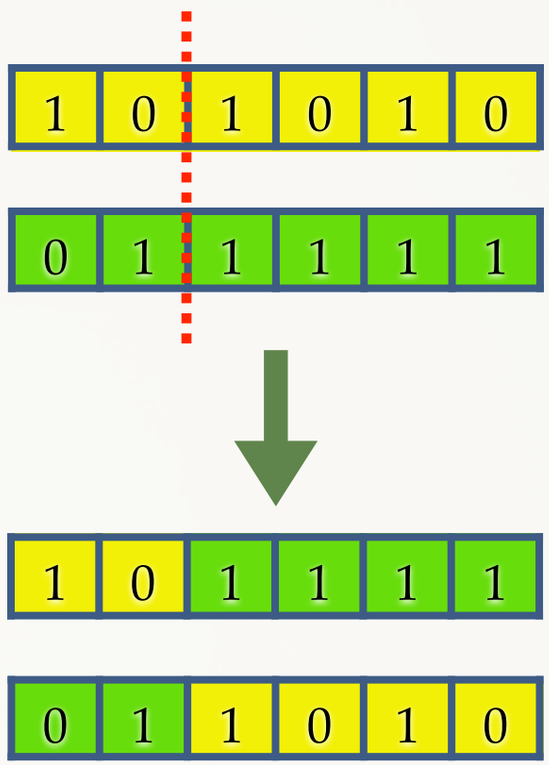
Genetic operations



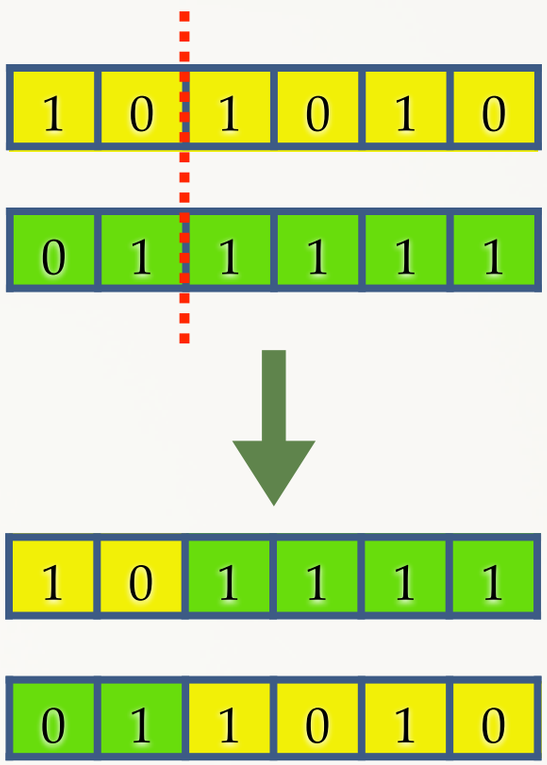
5. Crossover to create two children, with p_c

Genetic operations

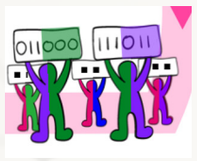
5. Crossover to create two children, with p_c



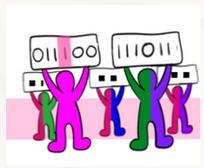
Genetic operations

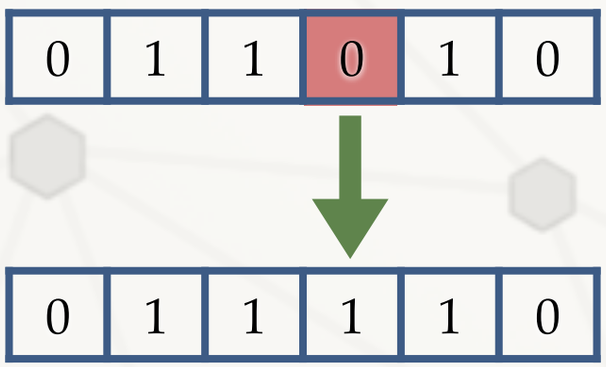


5. Crossover to create two children, with p_c



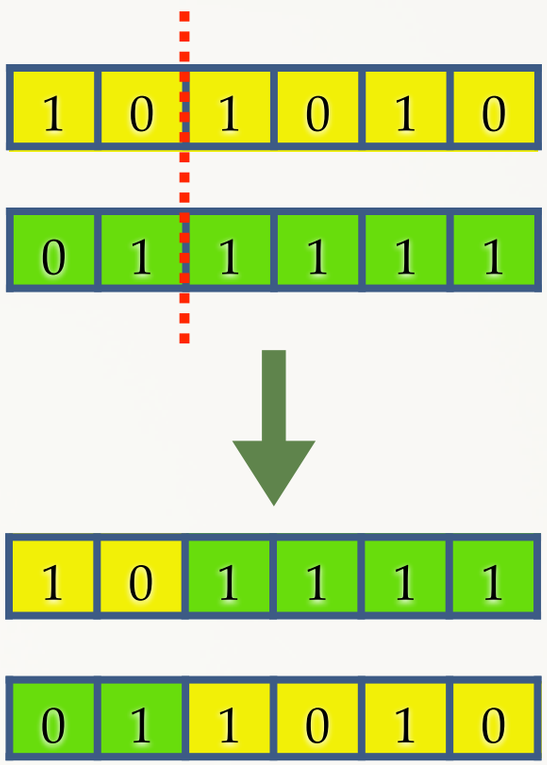
6. Mutation, with probability p_m



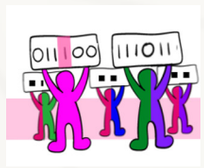


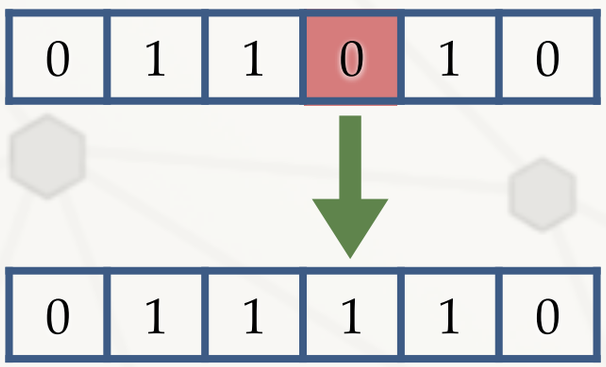
Genetic operations

5. Crossover to create two children, with p_c

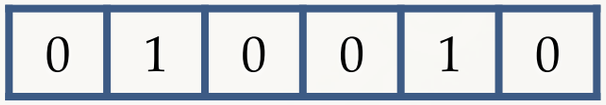


6. Mutation, with probability p_m



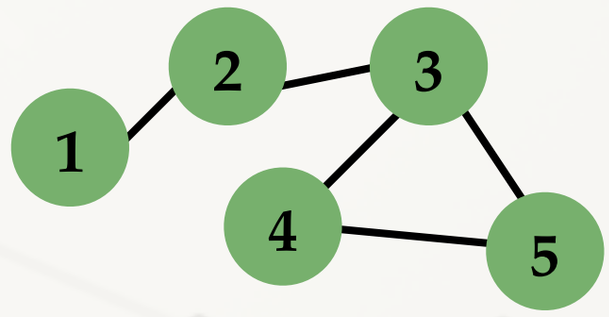


7. Repeat until target is found

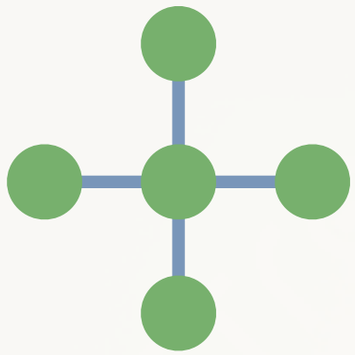


$$\mathcal{F}^i = 0$$

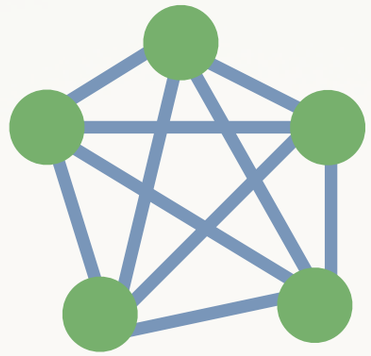
$$\Lambda_i = \Lambda^{TRUE}$$



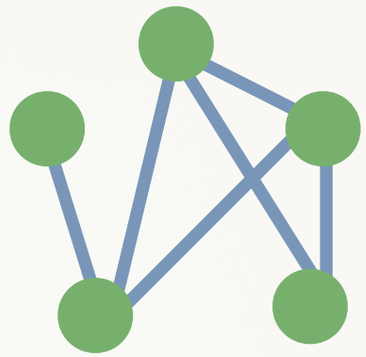
Results



star



complete



arbitrary

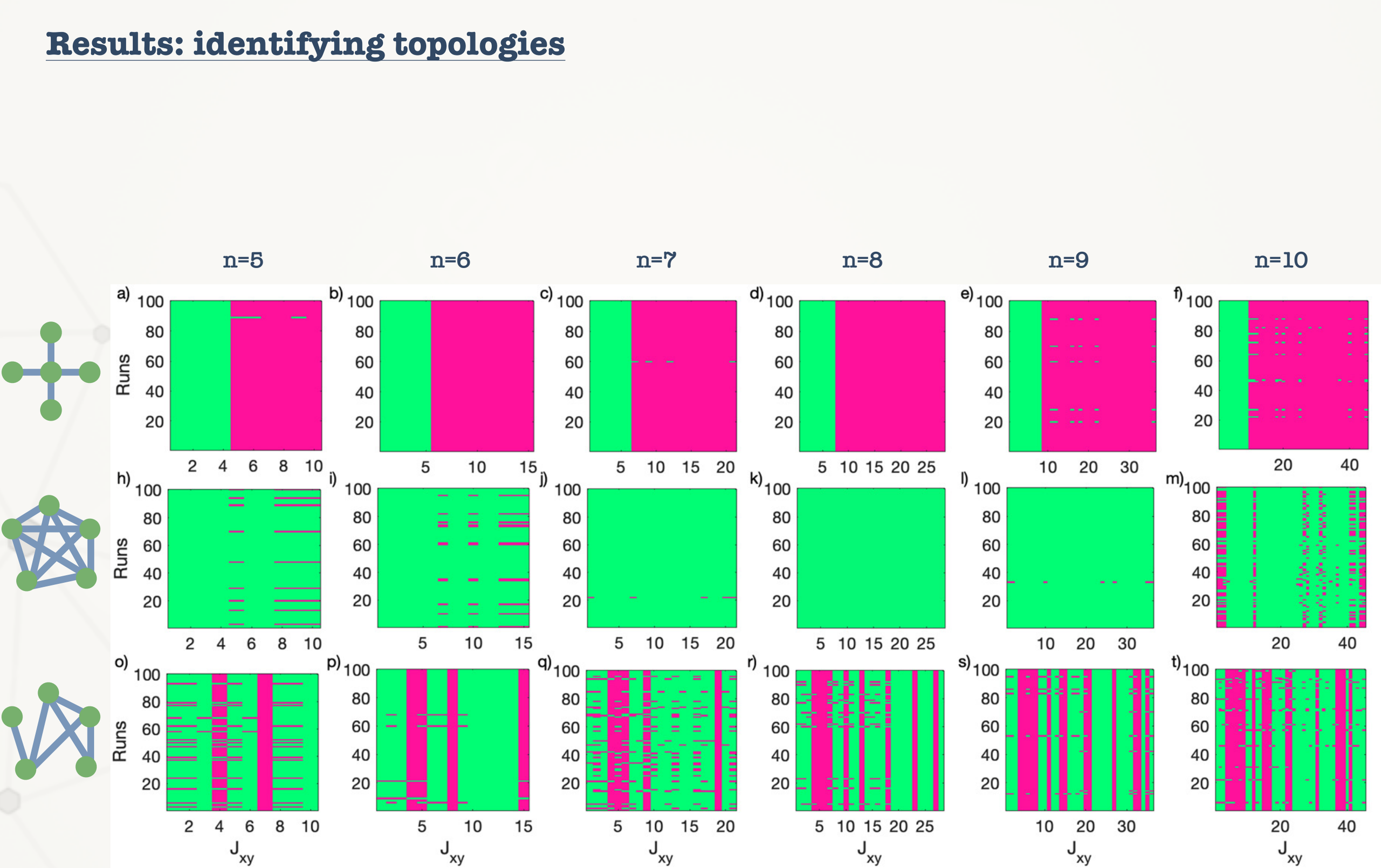
nodes $n = 5,\ldots,10$

Binary strings Λ_i of lengths n_c from 10 to 45

0	1	1	…	0	1	0

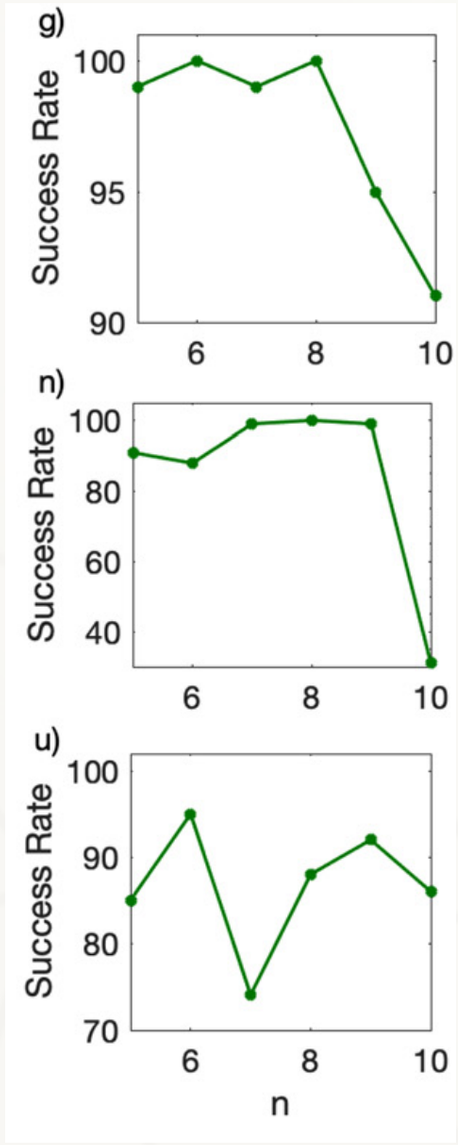
Search space 2^{n_c}

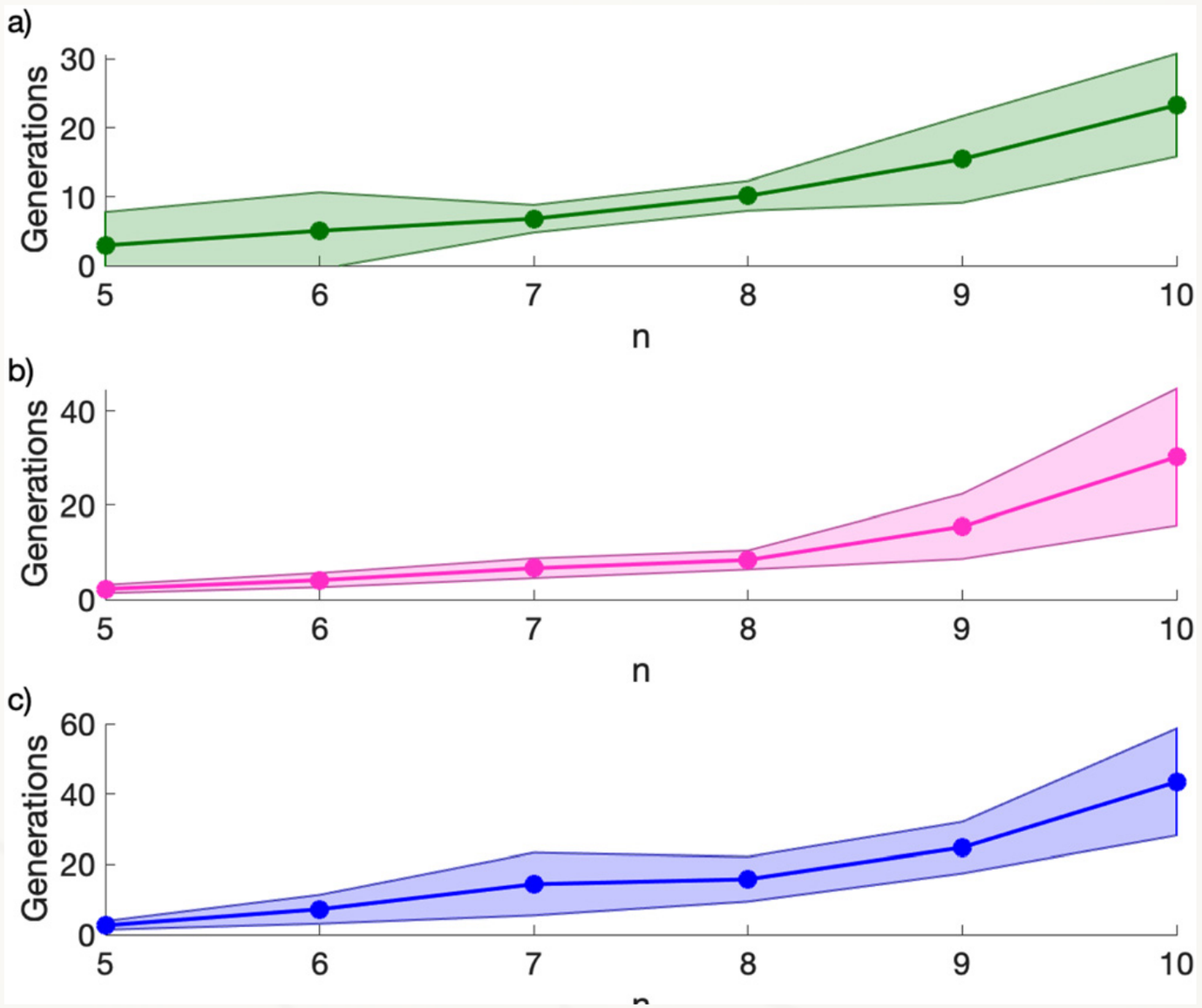
Results: identifying topologies



$J_{xy} = 1$ $J_{xy} = 0$

Results: performances





Results: the effect of noise

Real life scenario: Poissonian noise on the probability distributions.
We set a threshold value T below which two probabilities are considered equal.

Results: the effect of noise

Real life scenario: Poissonian noise on the probability distributions.
We set a threshold value T below which two probabilities are considered equal.

$n = 5$ sites
star, complete graph
$n_g = 5$

Results: the effect of noise

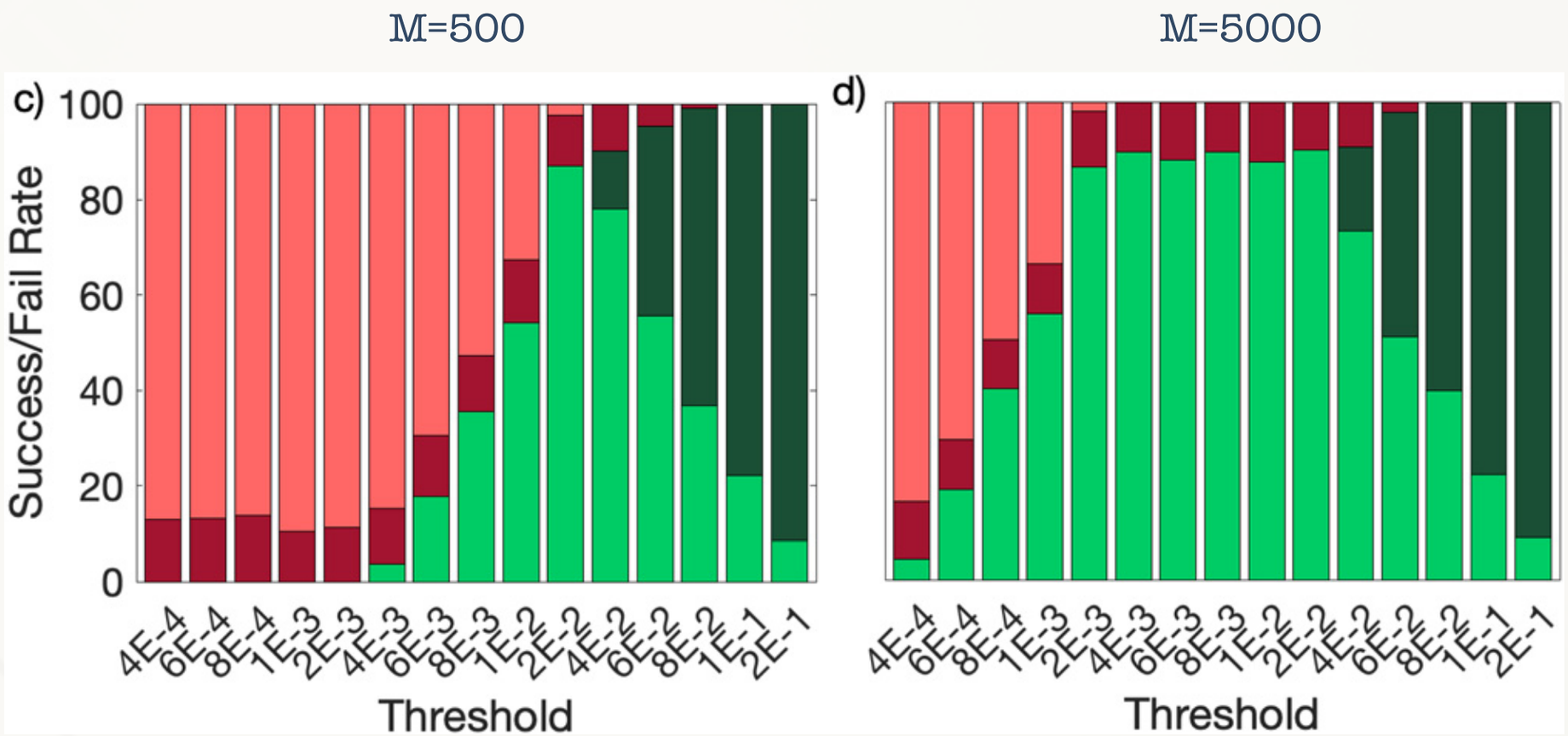
Real life scenario: Poissonian noise on the probability distributions.
We set a threshold value T below which two probabilities are considered equal.

$n = 5$ sites
star, complete graph
$n_g = 5$

Results: the effect of noise

Real life scenario: Poissonian noise on the probability distributions.
We set a threshold value T below which two probabilities are considered equal.

$n = 5$ sites
star, complete graph
$n_g = 5$



True negatives	F(c)>T, c not found
False negatives	F(c)>T, c found
True positives	F(c)<T, c found
False positives	F(c)<T, c not found

Larger threshold

Task 2: coupling estimation

$$|\langle v|\psi(t)\rangle|^2$$

$$|\psi_0\rangle$$

QW evolution??

$H = ?$

$$H = \begin{pmatrix} 0 & J_{12} & 0 & 0 & 0 \\ J_{12} & 0 & J_{23} & 0 & 0 \\ 0 & J_{23} & 0 & J_{34} & J_{35} \\ 0 & 0 & J_{34} & 0 & J_{45} \\ 0 & 0 & J_{35} & J_{45} & 0 \end{pmatrix}$$

How do we do this?

Determine the topology, with fixed J values

$J_{yk} = 0,1$

Search problem
Genetic algorithm

Coupling estimation, at fixed topology

$J_{yk} \in [0,1]$

Multiparameter estimation
Neural network

Task 2: coupling estimation

$$|\langle v|\psi(t)\rangle|^2$$

$$|\psi_0\rangle$$

QW evolution??

$H = ?$

$$H = \begin{pmatrix} 0 & J_{12} & 0 & 0 & 0 \\ J_{12} & 0 & J_{23} & 0 & 0 \\ 0 & J_{23} & 0 & J_{34} & J_{35} \\ 0 & 0 & J_{34} & 0 & J_{45} \\ 0 & 0 & J_{35} & J_{45} & 0 \end{pmatrix}$$

How do we do this?

Determine the topology, with fixed J values

$$J_{yk} = 0,1$$

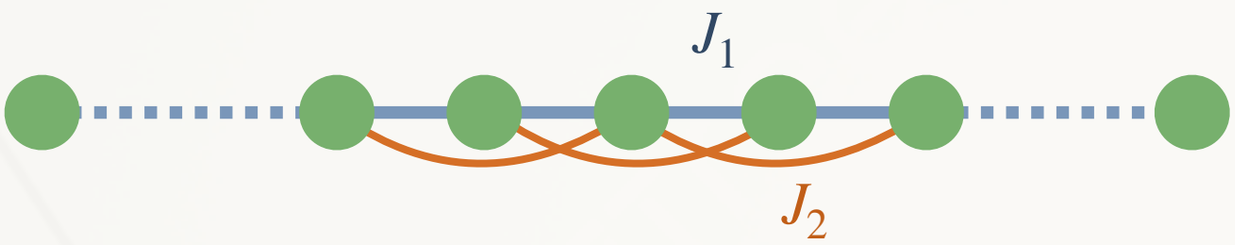
Search problem
Genetic algorithm

Coupling estimation, at fixed topology

$$J_{yk} \in [0,1]$$

Multiparameter estimation
Neural network

Task 2: Multiparameter estimation

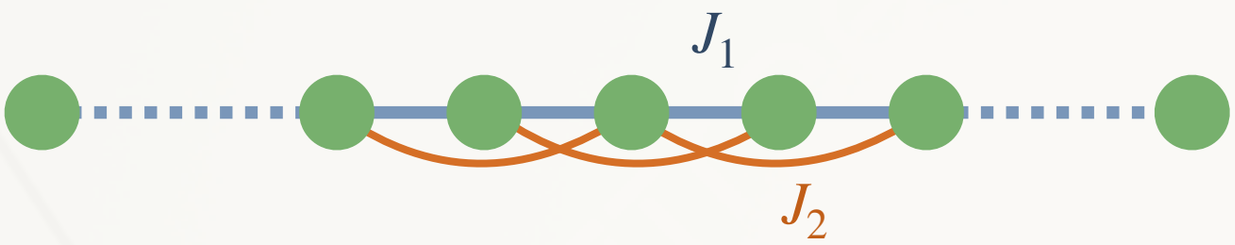


$$H(J_1, J_2) = \begin{pmatrix} 0 & J_1 & J_2 & 0 & 0 \\ J_1 & 0 & J_1 & J_2 & 0 \\ J_2 & J_1 & 0 & J_1 & J_2 \\ 0 & J_2 & J_1 & 0 & J_1 \\ 0 & 0 & J_2 & J_1 & 0 \end{pmatrix}$$

$J_1, J_2 \in [0,1]$

$$\pi_x(t; J_1, J_2) = \left| \langle x | e^{-iH(J_1,J_2)\,t} | \psi_0 \rangle \right|^2 \rightarrow$$ **Estimate J_1 and J_2**

Task 2: Multiparameter estimation



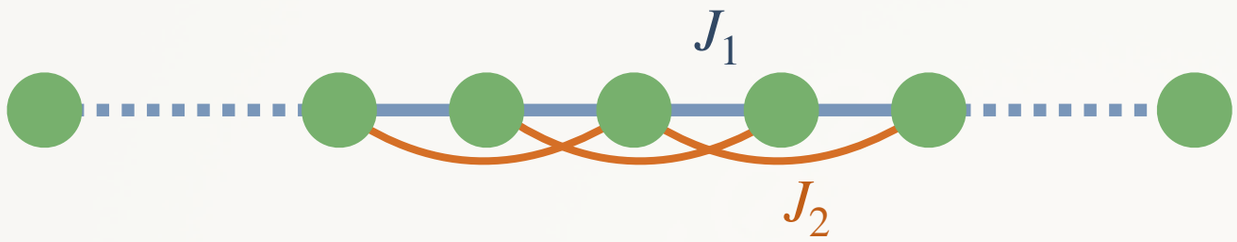
$$H(J_1, J_2) = \begin{pmatrix} 0 & J_1 & J_2 & 0 & 0 \\ J_1 & 0 & J_1 & J_2 & 0 \\ J_2 & J_1 & 0 & J_1 & J_2 \\ 0 & J_2 & J_1 & 0 & J_1 \\ 0 & 0 & J_2 & J_1 & 0 \end{pmatrix}$$

$J_1, J_2 \in [0,1]$

$$\pi_x(t; J_1, J_2) = \left| \langle x | e^{-iH(J_1,J_2)\,t} | \psi_0 \rangle \right|^2 \rightarrow$$ **Estimate J_1 and J_2**

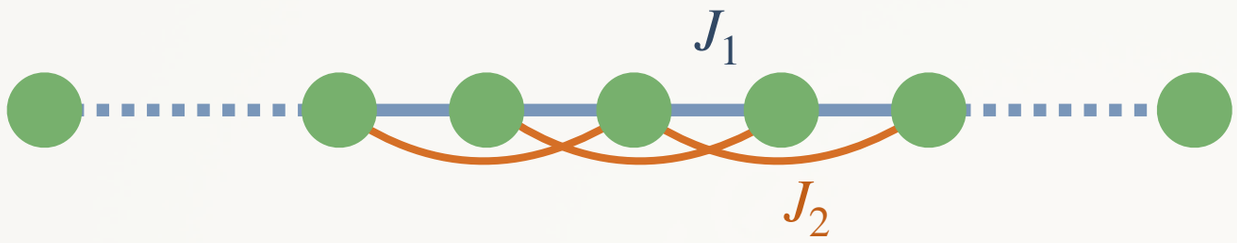
Our aim: to infer the **Hamiltonian parameters** of a CTQW on a line graph with n-neighbor interactions, having access only to the probability distribution over the nodes at a known time t and to the initial state of the system $|\psi_0\rangle$.

ML approach

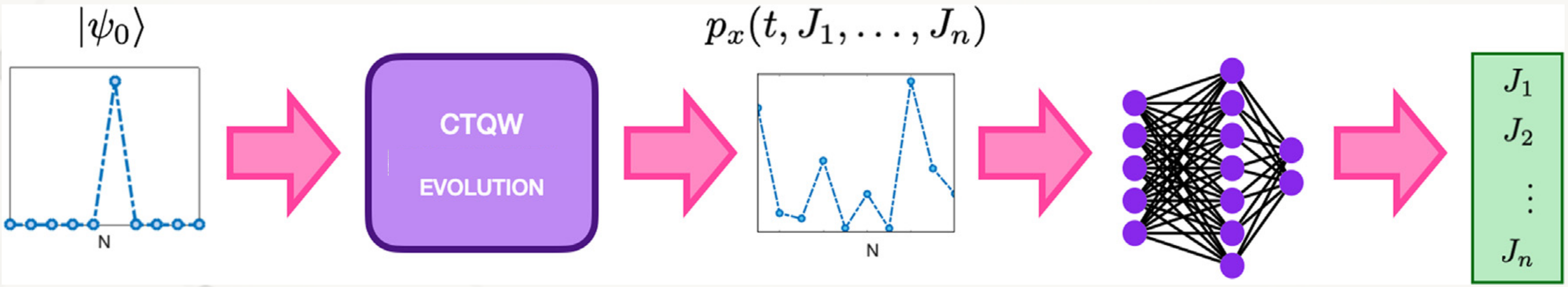


$$\pi_x(t; J_1, J_2) = \left| \langle x | e^{-iH(J_1,J_2)\,t} | \psi_0 \rangle \right|^2 \rightarrow \textbf{Estimate } J_1 \textbf{ and } J_2$$

ML approach



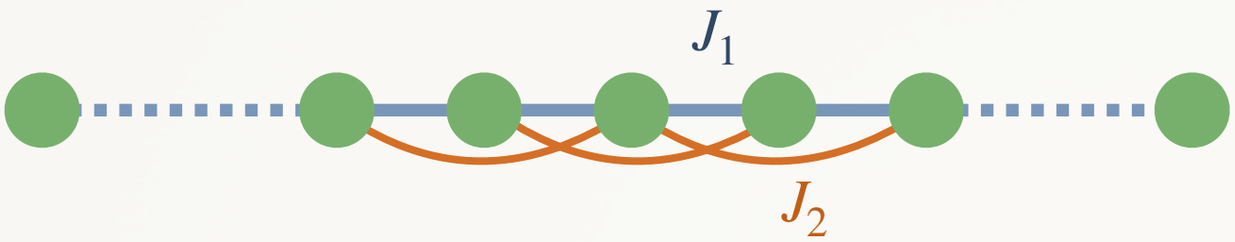
$$\pi_x(t; J_1, J_2) = \left| \langle x | e^{-iH(J_1,J_2)\,t} | \psi_0 \rangle \right|^2 \rightarrow \textbf{Estimate } J_1 \textbf{ and } J_2$$



t=1.5

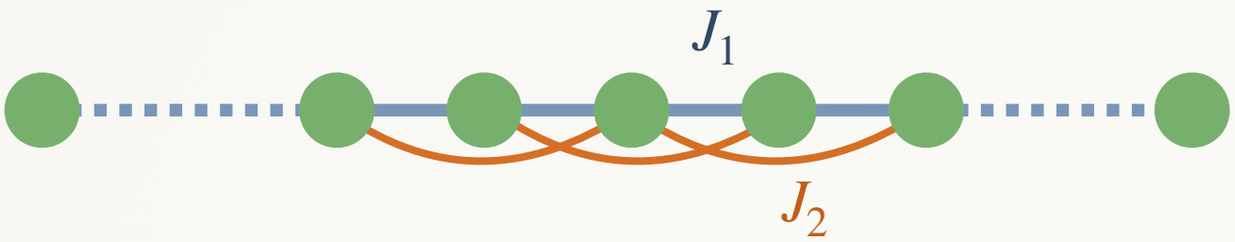
The estimation is carried out using a deep neural network model, which we train with data simulating experimental measured counts.

Task 2: Multiparameter estimation



We perform a position measurement of the QW

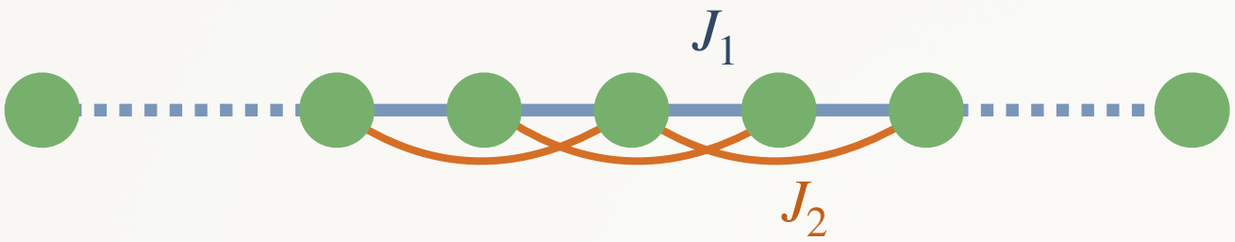
Task 2: Multiparameter estimation



We perform a position measurement of the QW

Quantum metrology: local estimation theory

Task 2: Multiparameter estimation

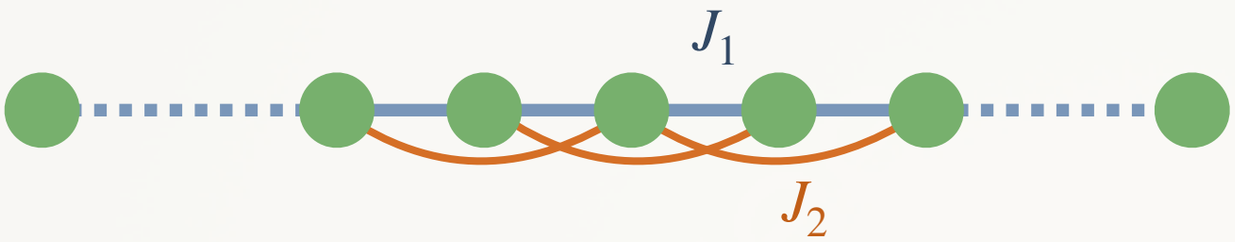


We perform a position measurement of the QW

Quantum metrology: local estimation theory

Maximum information extractable from a measurement: **Fisher information**

Task 2: Multiparameter estimation



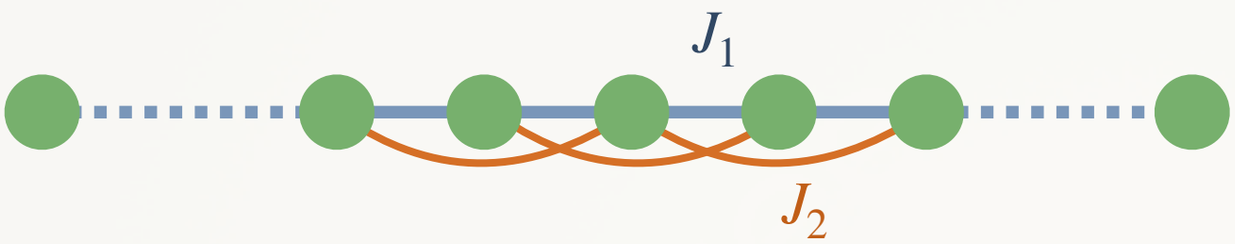
We perform a position measurement of the QW

Quantum metrology: local estimation theory

Maximum information extractable from a measurement: **Fisher information**

Fisher matrix: $$F_{ik} = \sum_x \frac{\partial_{J_i}\pi_x(t;J_1,J_2)\,\partial_{J_k}\pi_x(t;J_1,J_2)}{\pi_x(t;J_1,J_2)}$$

Task 2: Multiparameter estimation



We perform a position measurement of the QW

Quantum metrology: local estimation theory

Maximum information extractable from a measurement: **Fisher information**

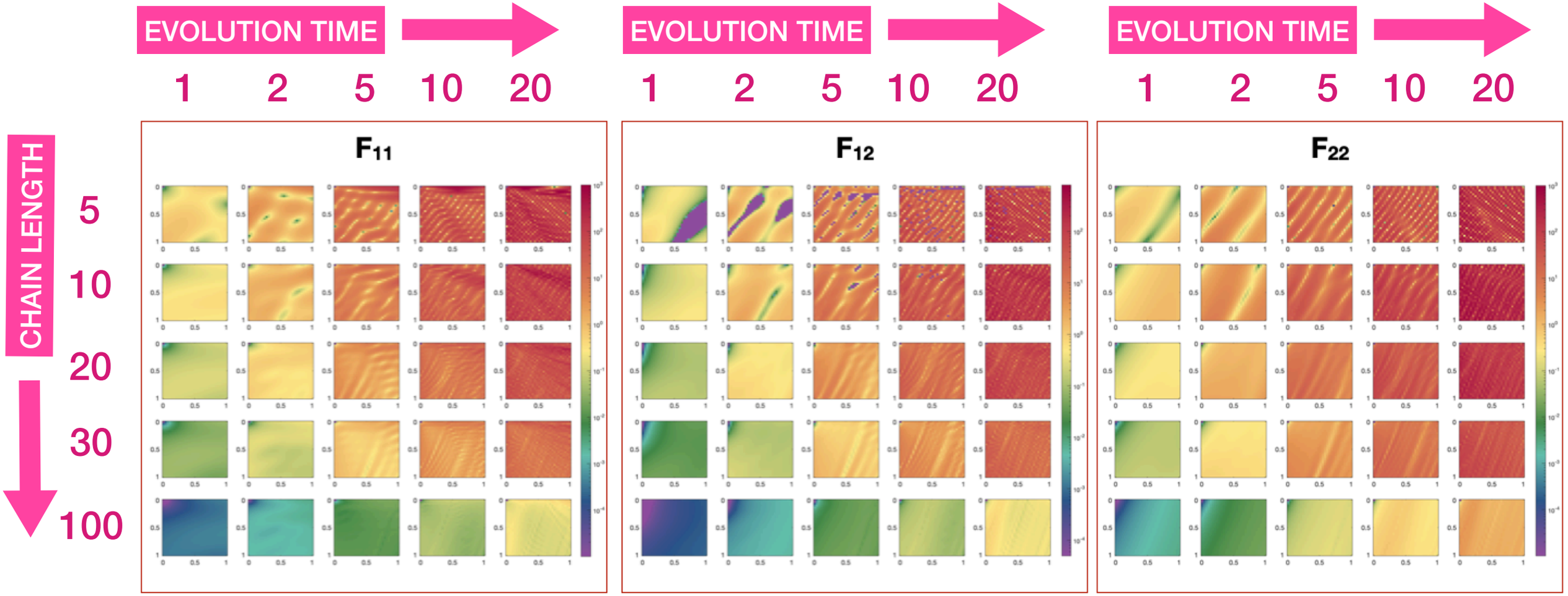
Fisher matrix: $$F_{ik} = \sum_x \frac{\partial_{J_i}\pi_x(t;J_1,J_2)\,\partial_{J_k}\pi_x(t;J_1,J_2)}{\pi_x(t;J_1,J_2)}$$

Cramér-Rao bound: lower bound on the variance of the estimated parameter $\Delta^2[J_k]$

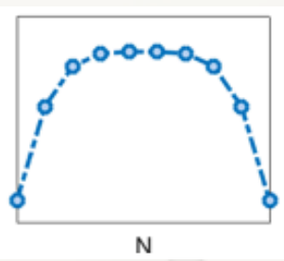
$$M\Delta^2[J_k] \geq (F^{-1})_{kk}$$

it will serve as a reference to quantify the performance of our estimation

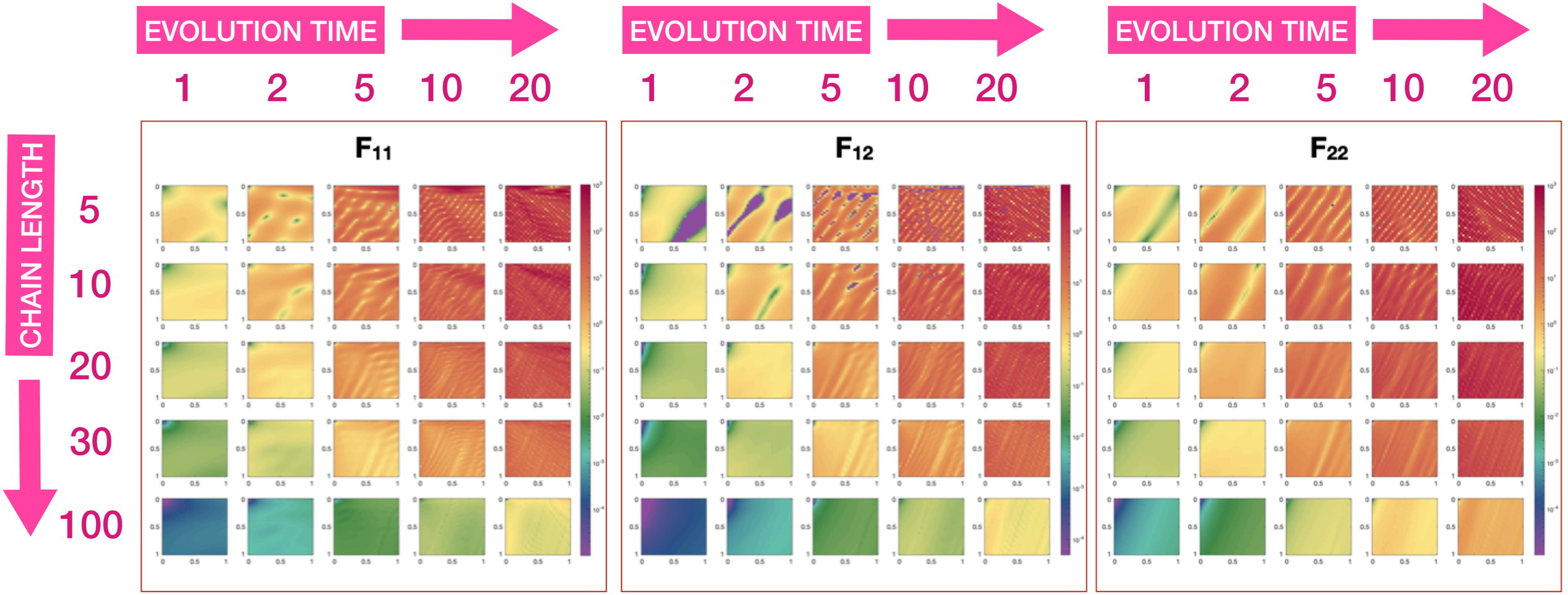
Fisher information



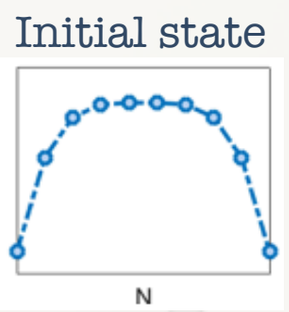
Initial state



Fisher information

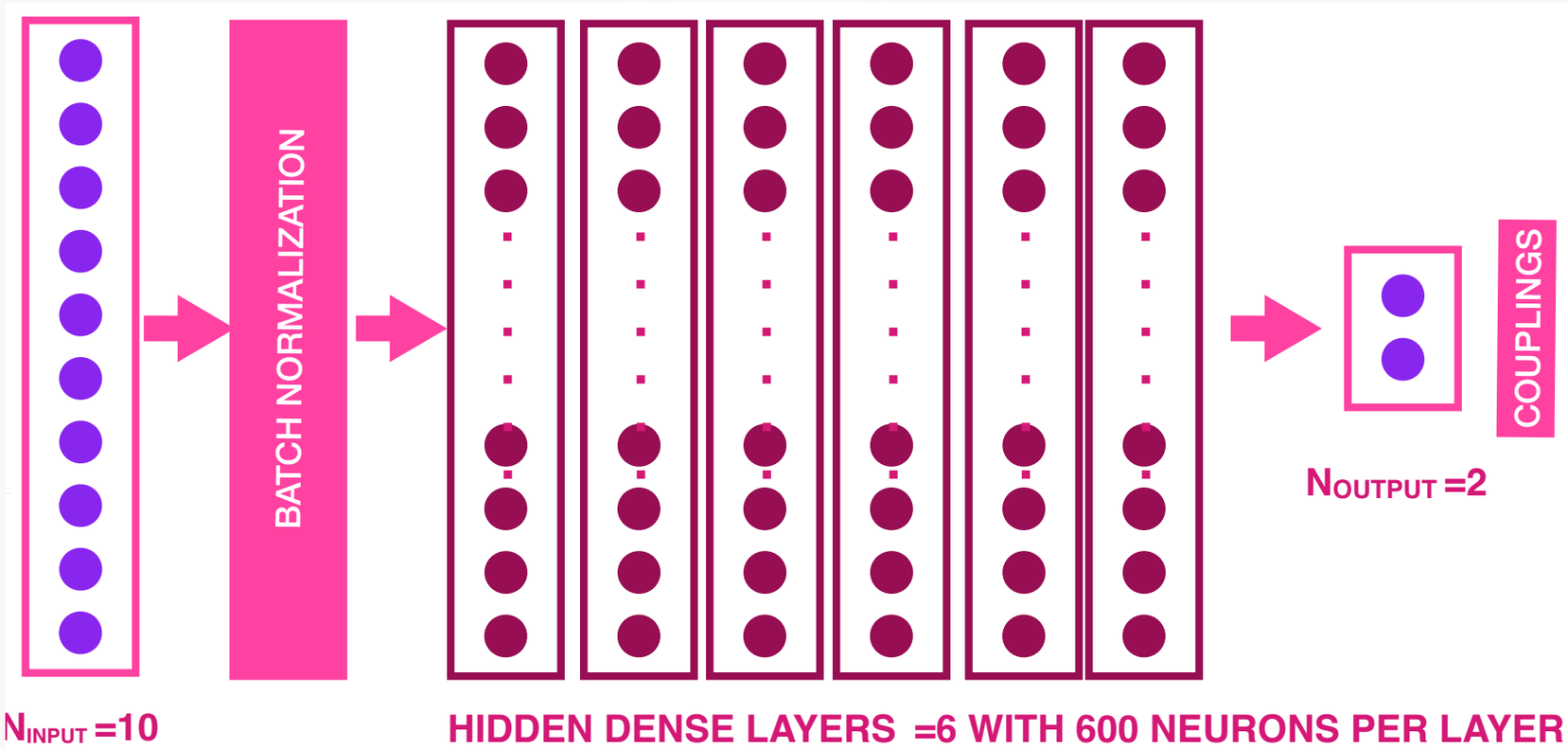


Key points:
Long times, short chains: extremely featured
Short times, long chains: bad estimation



Estimation by a neural network

$\pi_x(t; J_1, J_2)$



Create training set
Generate $N_{samp} = 2^{14}$ random values for $\{J_1, J_2\}$ unit distrib. in [-0.2,1.2]
&
generate the $\pi_x(t = 1.5; J_1, J_2)$

Bootstrap dataset
Generate simulated counts with $M = 2 \times 10^5$
&
simulate fluctuations with 500 MC experiments extracting new $\{\pi'_x\}$ from a Poisson distribution with mean $\{\pi_x\}$

Split the set
Split the dataset into training (0.8) and validation (0.2) set

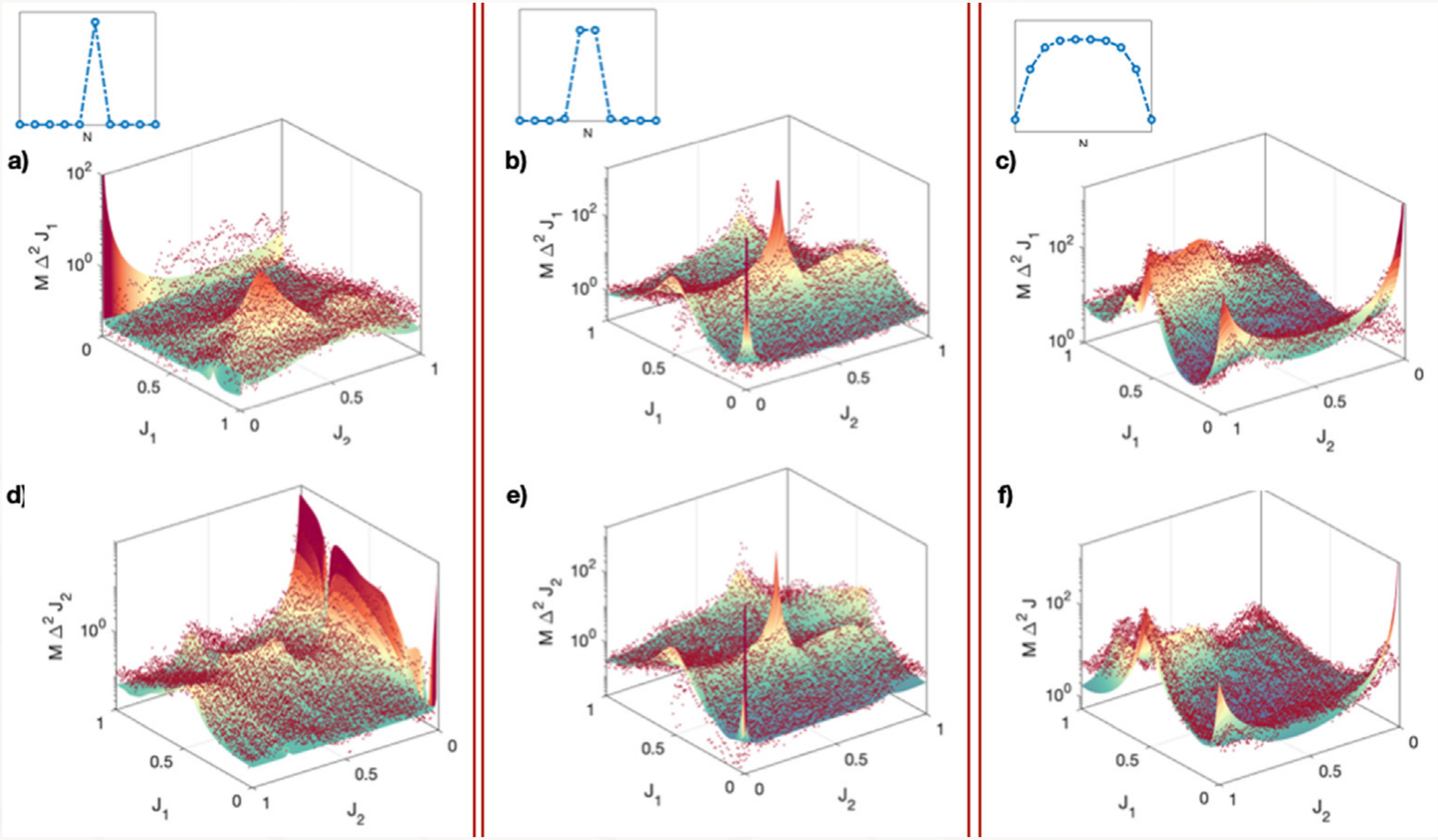
Training
Run the training for 200 epochs with batch size of 1000 using Adam optimizer with learning rate=10^{-3} and using MSE metric

Test the NN
Generate $N_{test} = 10^4$ values for $\{J_1, J_2\}$ in [0,1], evaluate the corresp. probabilities. Use the same M + generate MC Poissonian data

Evaluate precision of the estimation
Variance over the MC experiments and compare it with the CRB

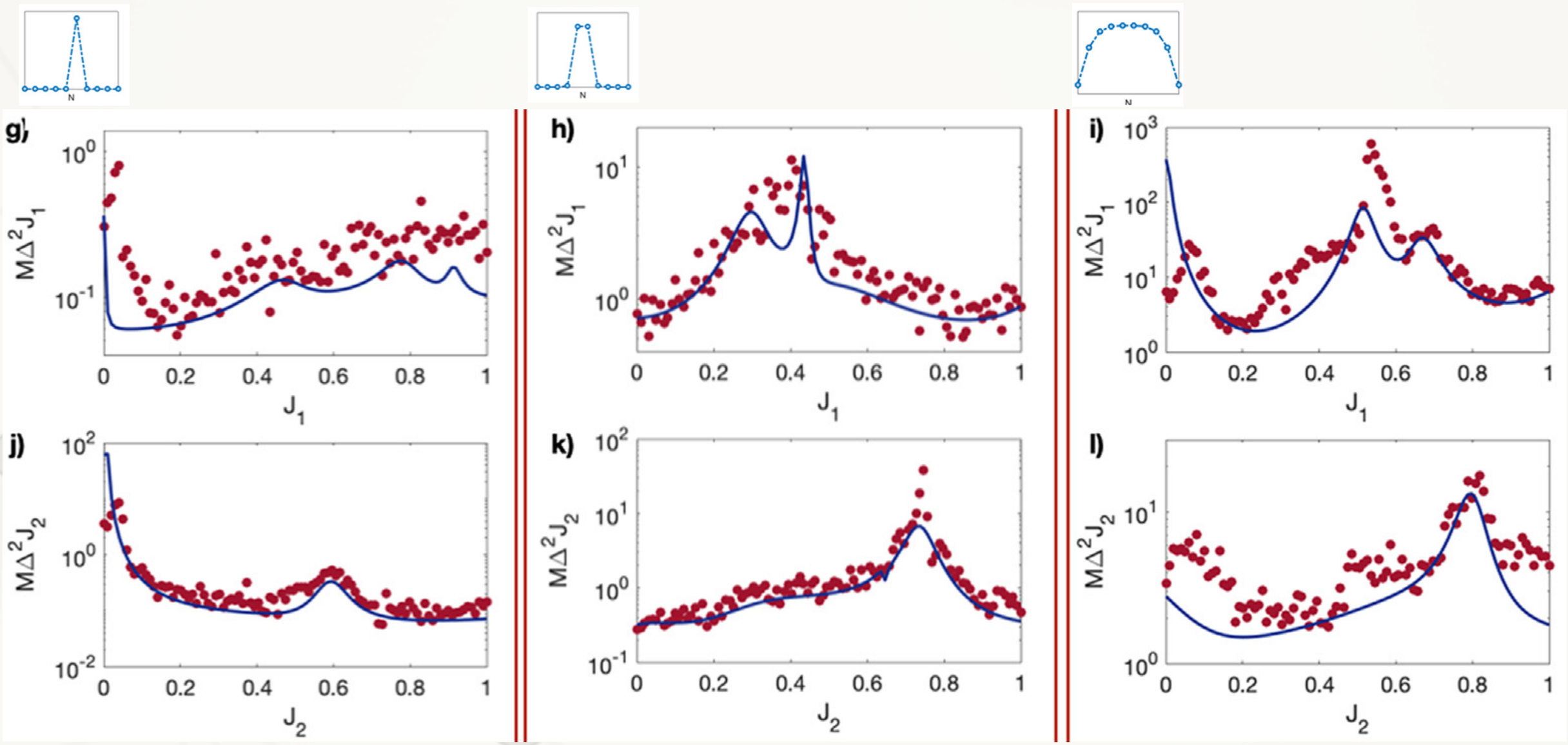
Comparison with the CRB

$$M\Delta^2[J_k] \geq (F^{-1})_{kk}$$



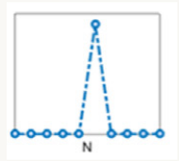
Comparison with the CRB

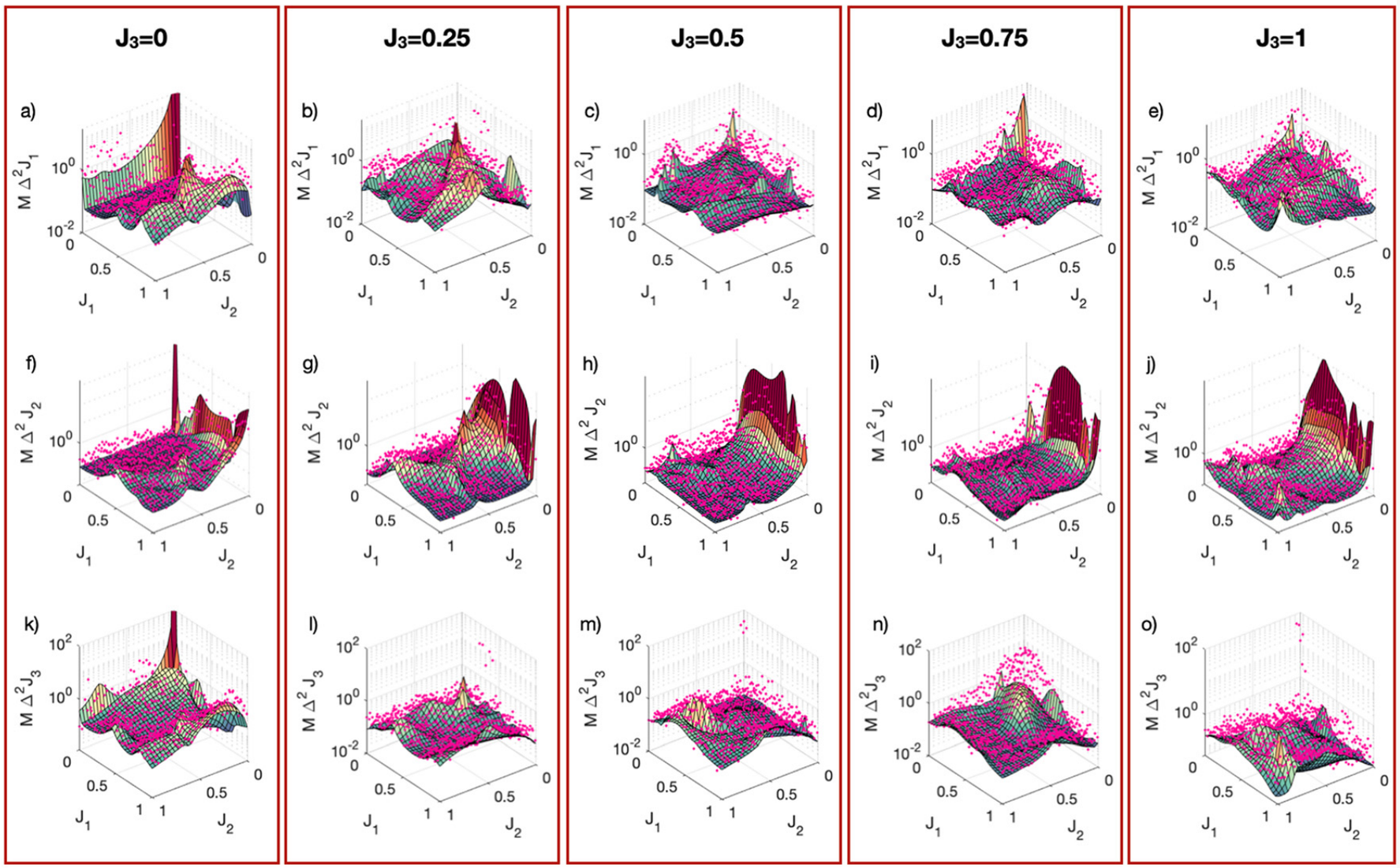
$$M\Delta^2[J_k] \geq (F^{-1})_{kk}$$

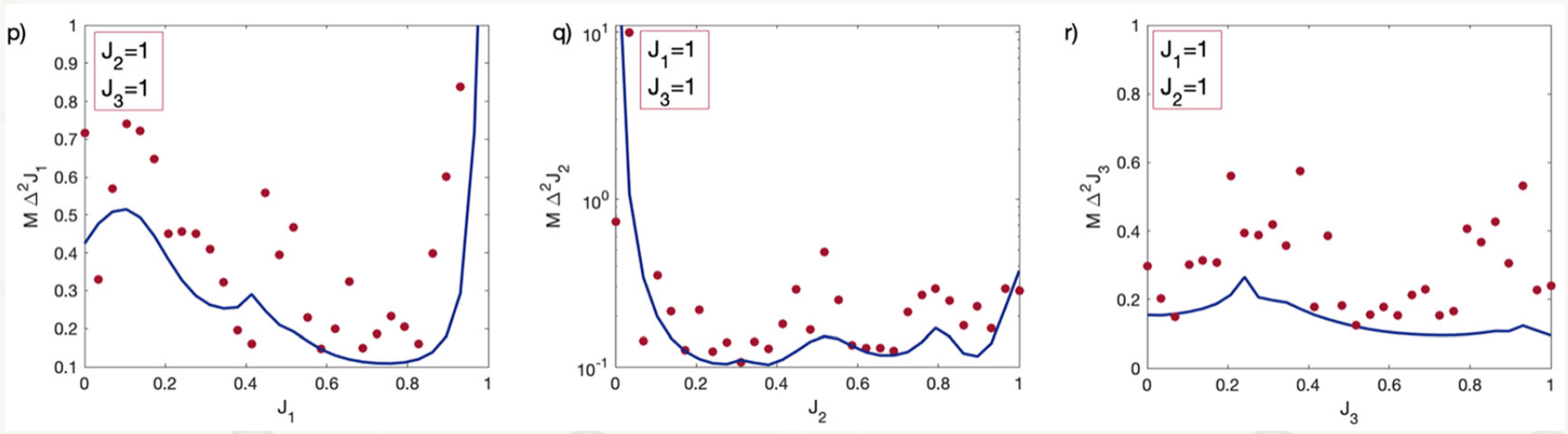


3-parameter estimation

$$M\Delta^2[J_k] \geq (F^{-1})_{kk}$$





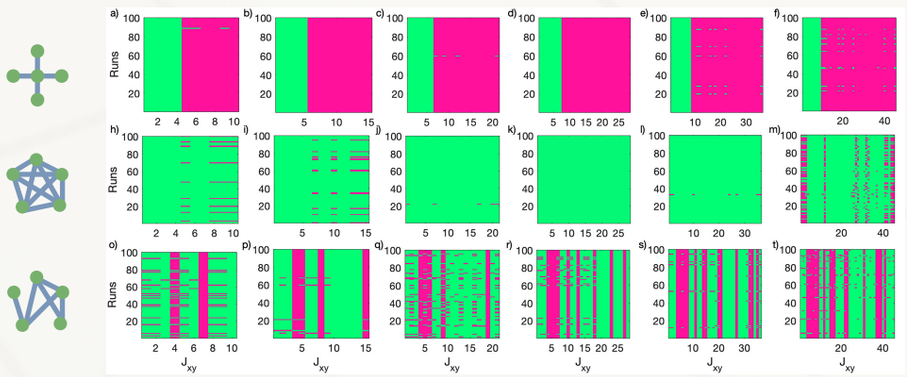


Conclusions & Final Comments

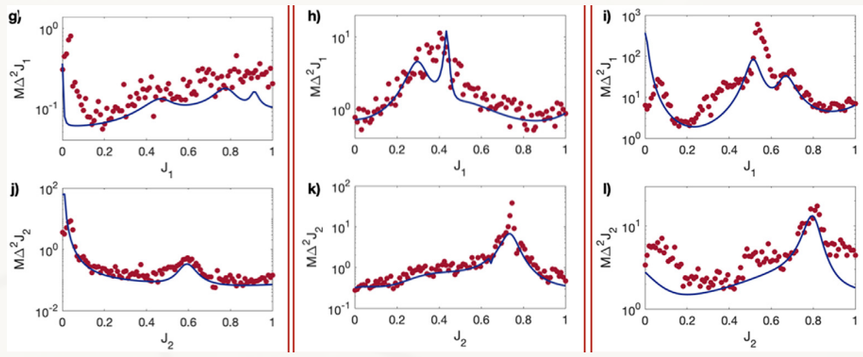
$|\langle v|\psi(t)\rangle|^2$

$|\psi_0\rangle$ → QW evolution?? → $H = ?$

- Further optimization by fine-tuning the hyperparameters for a specific network size: our results are but a lower bound to the achievable performance attainable.
- Estimation of on-site energies, or non-uniform couplings
- Our analysis involves a global measurement on the network. Localized/Partial measurements should be investigated.
- Generalize for estimation of n parameters & topology
- Important to make a connection between the probability distribution and topology



AVS Quantum Sci. 6, 014412 (2024)



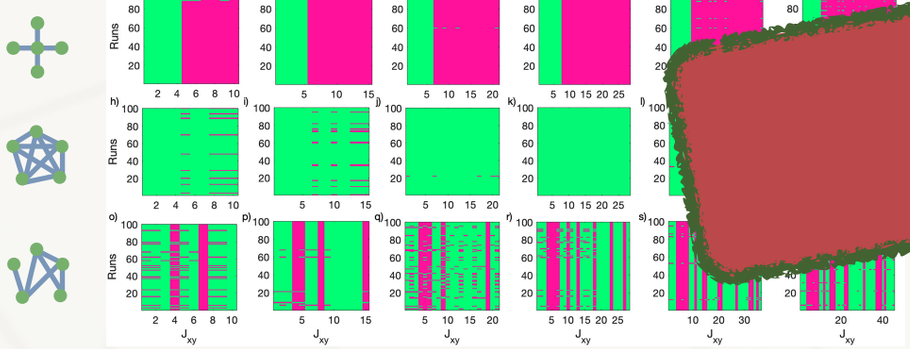
AVS Quantum Sci. 5, 014405 (2023)

Conclusions & Final Comments

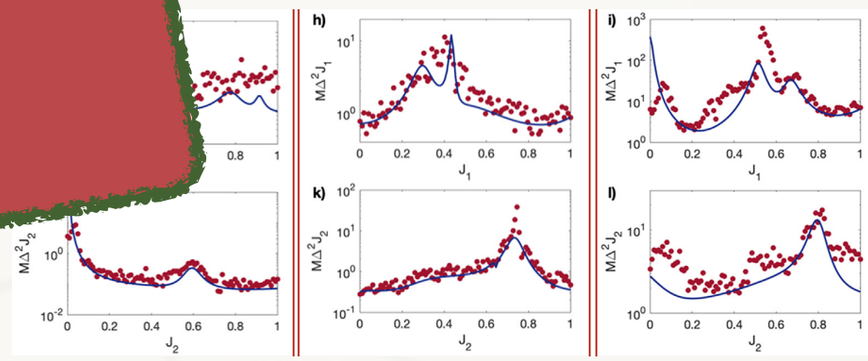
$|\langle v|\psi(t)\rangle|^2$

$|\psi_0\rangle$ → QW evolution?? → $H = ?$

- Further optimization by fine-tuning the hyperparameters for a specific network size: our results are but a lower bound to the achievable performance attainable.
- Estimation of on-site energies, or non-uniform couplings
- Our analysis involves a global measurement on the network. Localized/Partial measurements should be investigated.
- Generalize for estimation of n parameters & topology
- Important to make a connection between the probability distribution and topology



Grazie!



AVS Quantum Sci. 6, 014412 (2024)

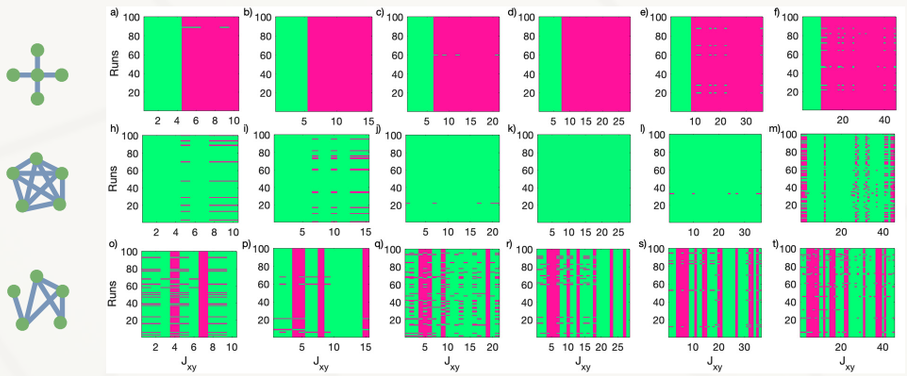
AVS Quantum Sci. 5, 014405 (2023)

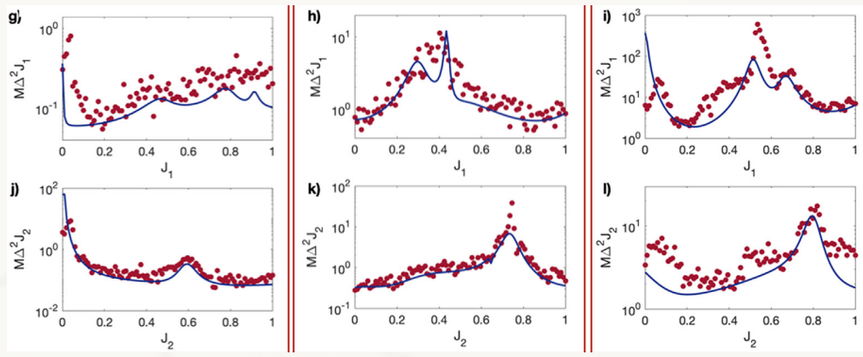
Conclusions & Final Comments

$|\langle v|\psi(t)\rangle|^2$

$|\psi_0\rangle$ → QW evolution?? → $H = ?$

- Further optimization by fine-tuning the hyperparameters for a specific network size: our results are but a lower bound to the achievable performance attainable.
- Estimation of on-site energies, or non-uniform couplings
- Our analysis involves a global measurement on the network. Localized/Partial measurements should be investigated.
- Generalize for estimation of n parameters & topology
- Important to make a connection between the quantum state/distribution and topology





The genetic algorithm: pseudocode

Algorithm 1 Genetic Algorithm

$gen \leftarrow 0$
Randomly generate n_p binary arrays $\{\Lambda_i\}$
$P_{gen} \leftarrow \{\Lambda_i\}$ ▷ Initialize population
while $gen < n_g$ **do**
  for $i = 0 \rightarrow n_p - 1$ **do**
    S_i= FITNESS$(\Lambda_i, \pi(\{t_k\}, \Lambda^{\mathrm{QW}}))$ ▷ Evaluate scores
  end for
  $best \leftarrow (\mathbf{Min}(S), \Lambda_{\mathbf{Min(S)}})$
  if $best[0] = 0$ **then**
    return $best$
  end if
  for $i = 0 \rightarrow p_e n_p - 1$ **do**
    HOF$_i \leftarrow (\Lambda_i, S_i)$ sorted by scores ▷ Hall of fame
  end for
  Insert HOF into P_{gen+1}
  for $j = 0 \rightarrow n_p(1 - p_e)/2 - 1$ **do**
    $\Lambda_1^j, \Lambda_2^j \leftarrow$ TOURNAMENT(P_{gen}, S) ▷ Select parents
    Add CROSSOVER$(\Lambda_1^j, \Lambda_2^j)$ to children ▷ Children
  end for
  for $i = 0 \rightarrow n_p(1 - p_e) - 1$ **do**
    Apply MUTATION(children$_i$) ▷ Mutation
  end for
  Insert children in P_{gen+1}
  $gen \leftarrow gen + 1$
end while

Algorithm 2 Genetic functions

function FITNESS$(\Lambda_i, \pi(t_k, \mathbf{\Lambda}^{\mathrm{QW}}))$:
  Evaluate $\pi(t_k, \Lambda_i)$
  Evaluate KLD$(\pi(t_k, \Lambda_i), \pi(t_k, \mathbf{\Lambda}^{\mathrm{QW}}))$
  return KLD
end function

function TOURNAMENT(P_{gen}, S):
  id← random integer in $[0, n_p]$
  for $j = 0 \rightarrow k - 2$ **do**
    aux← random integer in $[0, n_p)$
    if S[aux] $<$ S[id] **then**
      id← aux
    end if
  end for
  return Λ[id]
end function

function CROSSOVER(Λ_1, Λ_2):
  Generate a random integer x in $[0, 1]$
  if $x < p_c$ **then**
    $y \leftarrow$ random integer in $[0, n_c)$
    child$_1 \leftarrow$ concatenate$(\Lambda_1[0 : y], \Lambda_2[y + 1 : n_c - 1])$
    child$_2 \leftarrow$ concatenate$(\Lambda_2[0 : y], \Lambda_1[y + 1 : n_c - 1])$
  end if
  return child$_1$,child$_2$
end function

function MUTATION(child$_i$):
  for $j = 0 \rightarrow n_c - 1$ **do**
    Generate random x in [0,1]
    if $x < p_m$ **then**
      child$_i[j] \leftarrow 1-$child$_i[j]$
    end if
  end for
  return child$_i$
end function