Experimental Realization of a Quantum Autoencoder

Alex Pepper, Nora Tischler, Geoff J. Pryde

Phys. Rev. Lett. 122, 060501 – Published 11 February 2019

Recap: Autoencoders

- Unsupervised machine learning method
- Encode input data through a *latent space*
- Learn structures in data without previous knowledge

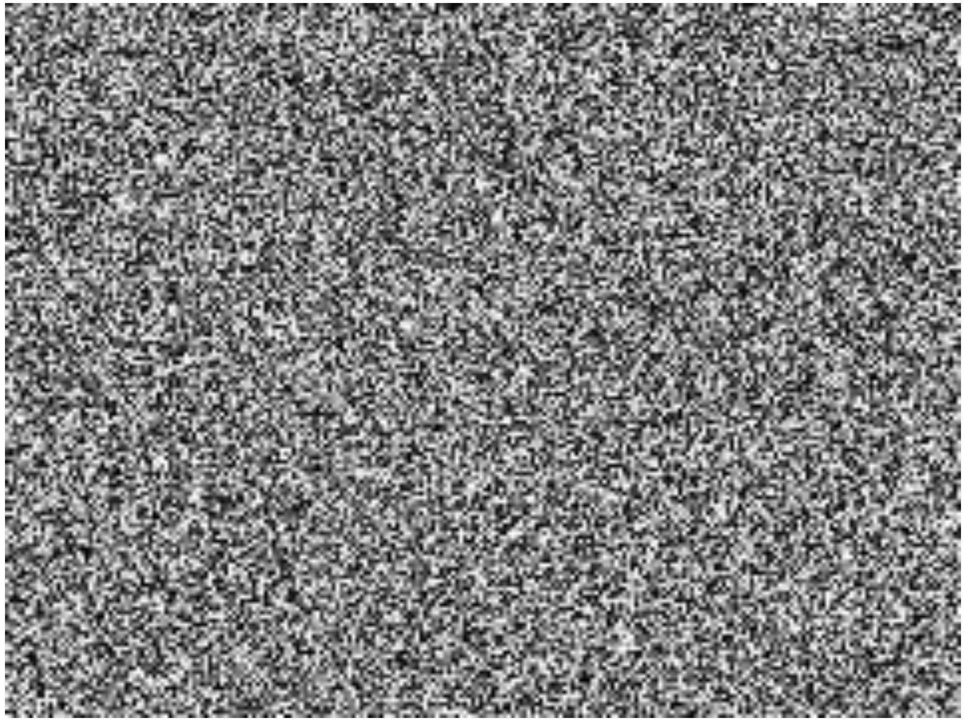
Efficient information storage: Encoding

Efficient information storage: Encoding

- Depends on data

Efficient information storage: Encoding

- Depends on data

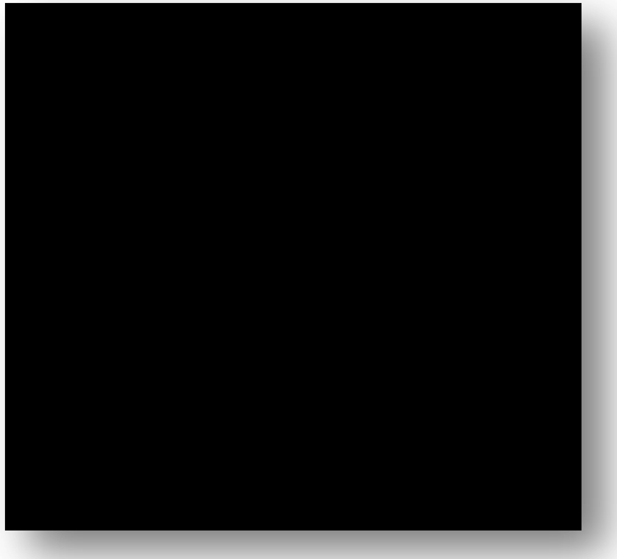


[5]

https://en.wikipedia.org/wiki/White_noise

Efficient information storage: Encoding

- Depends on data



[6]

Efficient information storage: Encoding

- Depends on data
- Compressible to one bit: 1 or 0
- Don't save each pixel! Find structure

[6]

Efficient information storage: Encoding

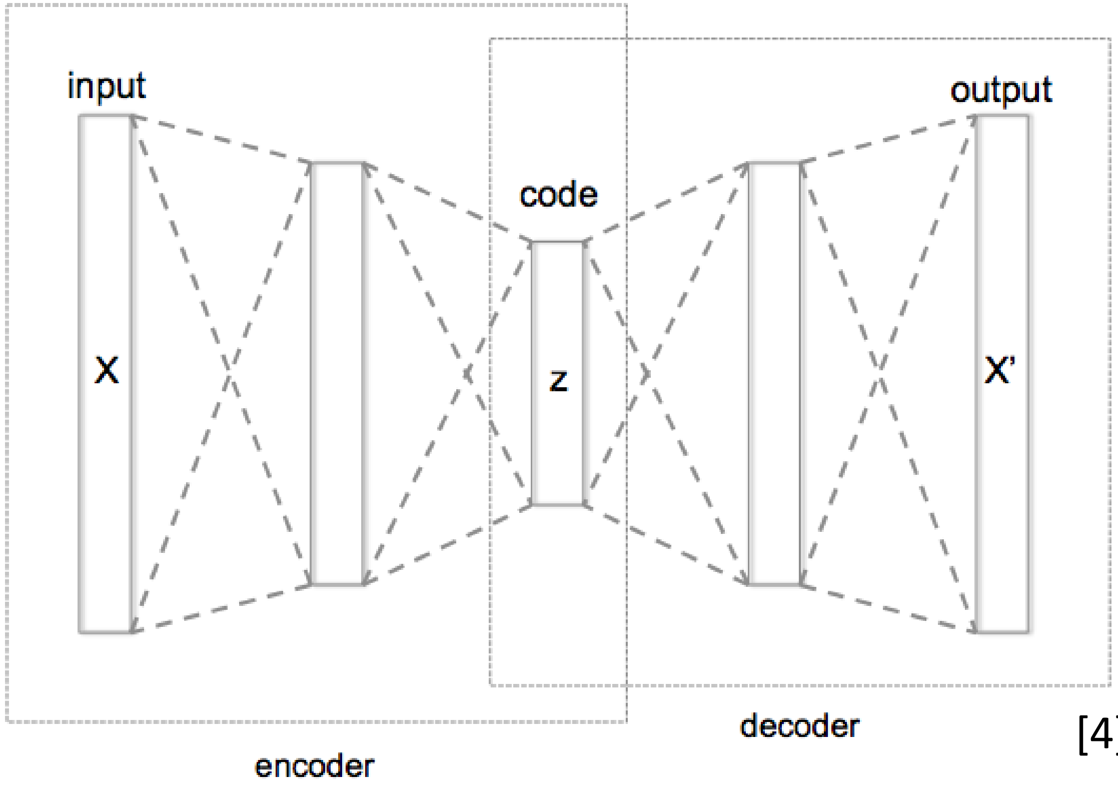
- Depends on data
- Compressible to one bit: 1 or 0
- Don't save each pixel! Find structure

- Problem: Need extensive knowledge about data
- How to do this without prior knowledge?

Autoencoders

- Encoder
  - Input X in inefficient format
  - Decreasing layer size
  - Squeeze through "bottleneck"

- Decoder
  - Increasing layer size
  - Reconstruct input as closely as possible



[4]

https://en.wikipedia.org/wiki/Autoencoder

Demonstration

- Goal: Encoding of portraits of highschool-students

- Image dimensions: 144 x 192 x 3 = 82.944

- Latent space: 80 Dimensions

→ Cut off encoder and feed manual input to decoder

Why do we care?

- Application in Quantum Physics
  - Most prominently: Quantum Computers

- Resources are even more valuable than in classical computers

- Reduce complicated Hilbert-space without assumptions

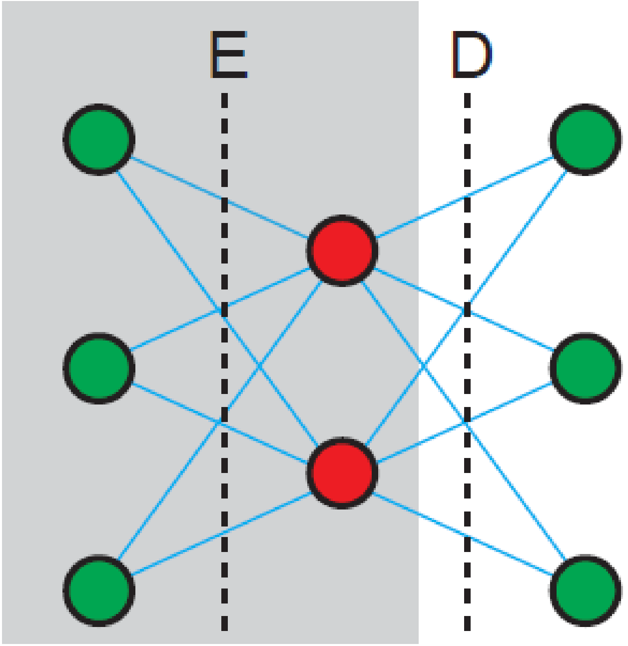
- Classic approach: handcrafted for single use case

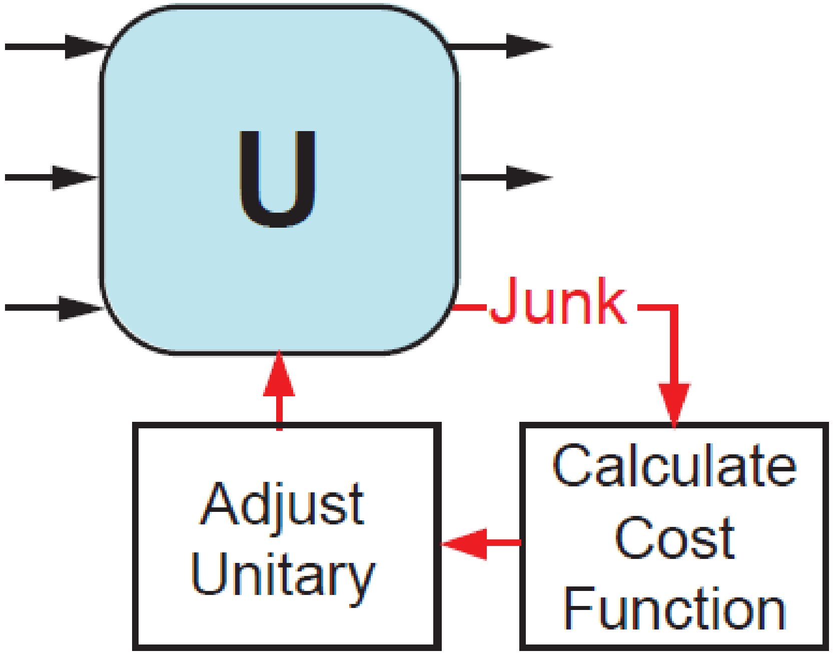
Proposal: Quantum-Autoencoder

- Compress qudits to qunits (d > n)
  - Here: d = 3, n = 2

- Encode qutrits in single photons

Proposal: Quantum-Autoencoder



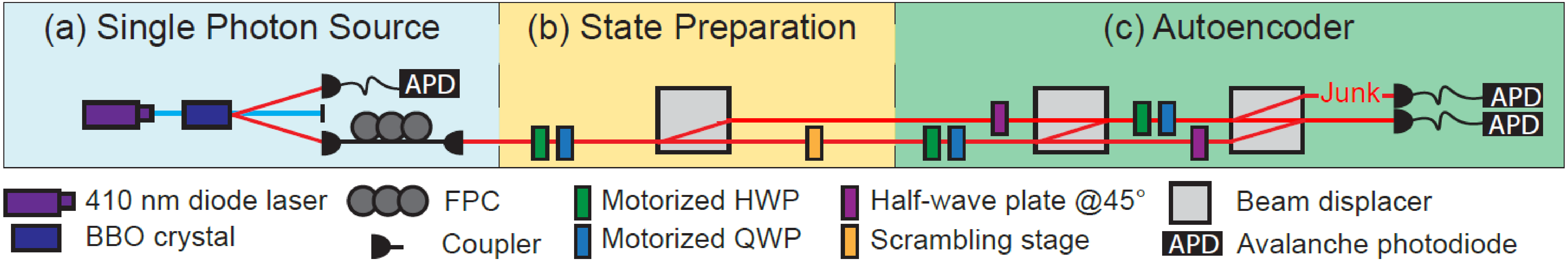


[1]

Unitary U: characterized by set of parameters

Minimize occupation probability of *junk mode*

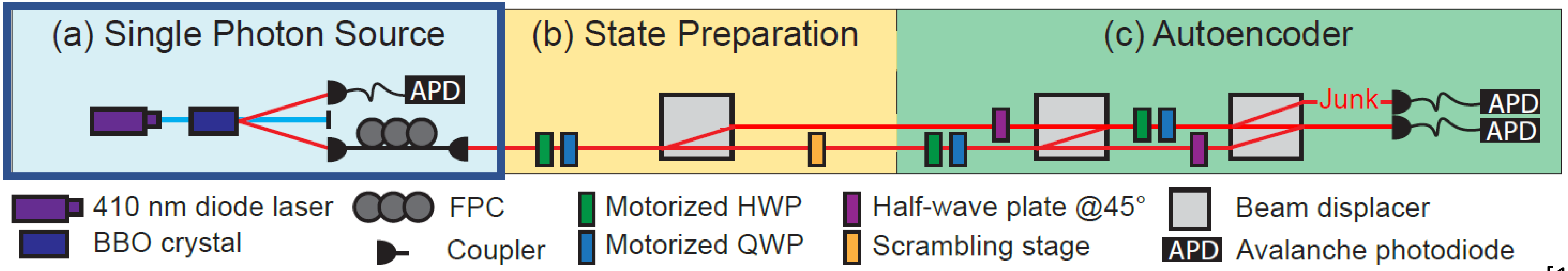
Experimental realization



[1]

Single Photon Source

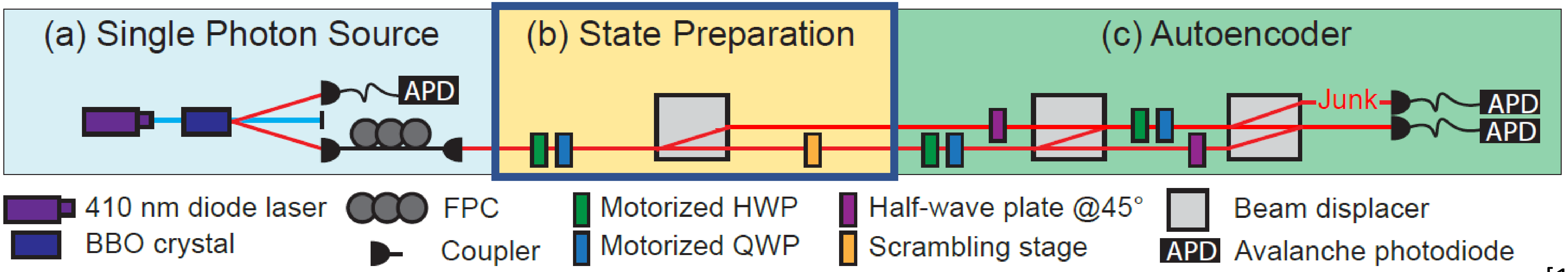
- Create photon pairs (SPDC)
  - Detect one photon
  - Couple other photon to fiber & pass through FPC



[1]

State Preparation

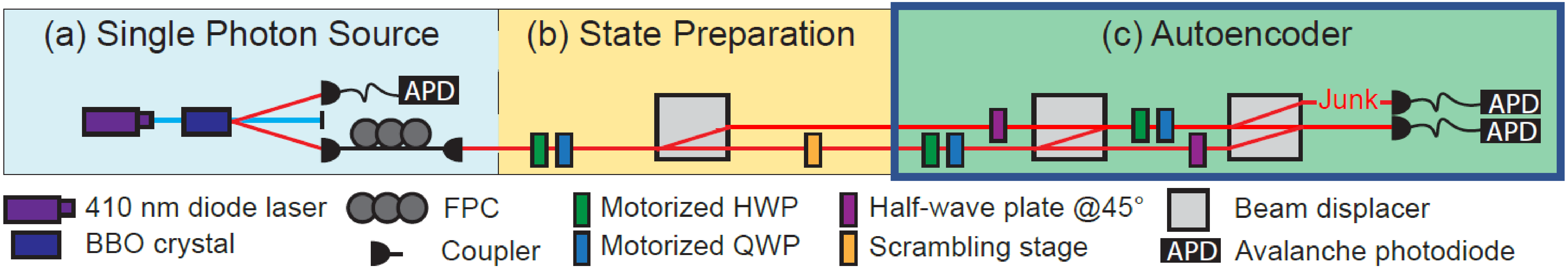
- Encode qubit in polarization state via two (adjustable) waveplates
- Pass through polarizing beam displacer
- Scramble lower spatial mode (fixed)



[1]

Autoencoder

- 3 x 3 unitary transformation - 4 free parameters
  - 2 x 2 unitaries (wave plates)
  - Mode permutation (beam displacers)



Quantify compression performance

- Use occupation probability of junk mode: P_j
  - Success probability of encoding process is $1 - P_j$
  - Fidelity between input state and output is also $1 - P_j$

  → Minimize P_j during training

- Define cost function as average junk mode occupation prob.

Training process

- Initiate settings randomly $\quad$ x = (x_1, x_2, x_3, x_4)
- At each iteration estimate gradient $\nabla C|_{x_{cur}}$
  - Probe n-th plate: $x_{cur} \xrightarrow{\text{rotate by } s_a} x_{pn}$
  - $\left.\frac{\partial C}{\partial x_n}\right|_{x_{cur}} = \left[C(x_{pn}) - C(x_{cur})\right]/s_a$
  - Return to previous position and repeat for next plate
  - Rotate all plates simultaneously
  - $x_{cur} \rightarrow x_{cur} - s_a \nabla C|_{x_{cur}}$

Training process

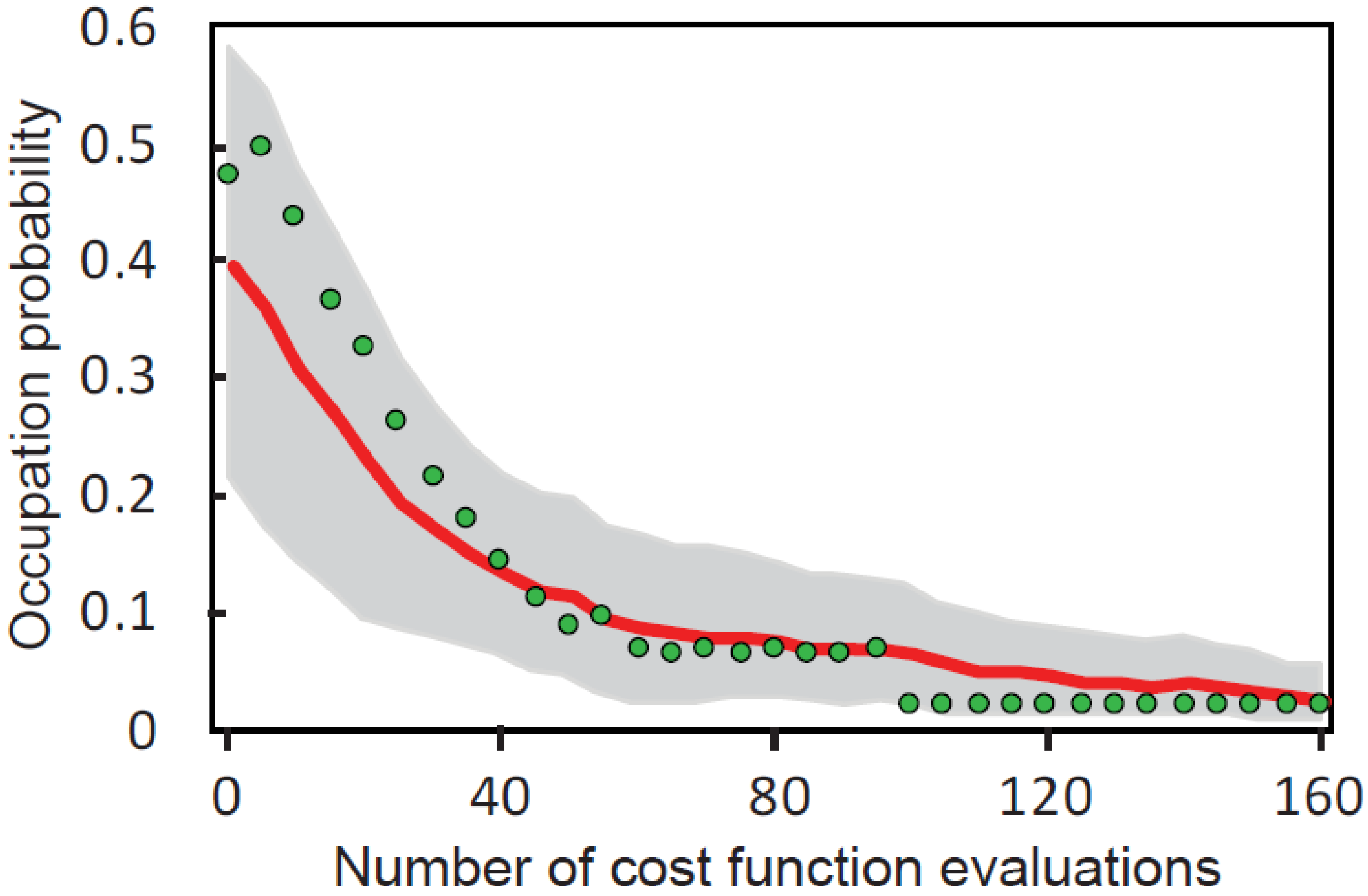
- Start with coarse adjustment value (12°), switch to smaller value (5°) when cost function smaller than 0.1

Training process

- Start with coarse adjustment value (12°), switch to smaller value (5°) when cost function smaller than 0.1
- If cost function not below 0.1 after 50 steps, rotate each plate 25°
    - Empirical method to avoid local minima

Results: Training process

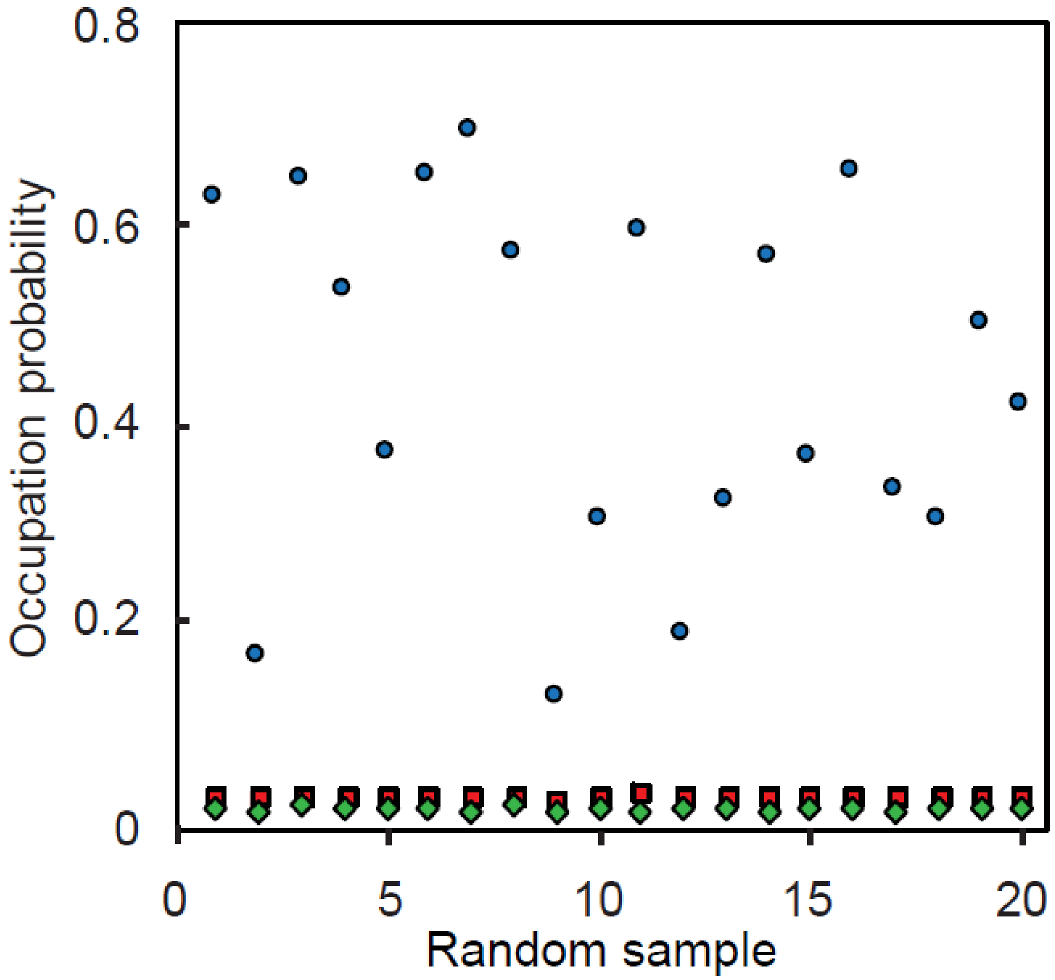
- Training starting from 20 different initializations of the unitary
- Two fixed, randomly selected training states
- Average occupation prob. 0.03 ± 0.03 after 160 cost function evaluations



[1]

Results: # training samples

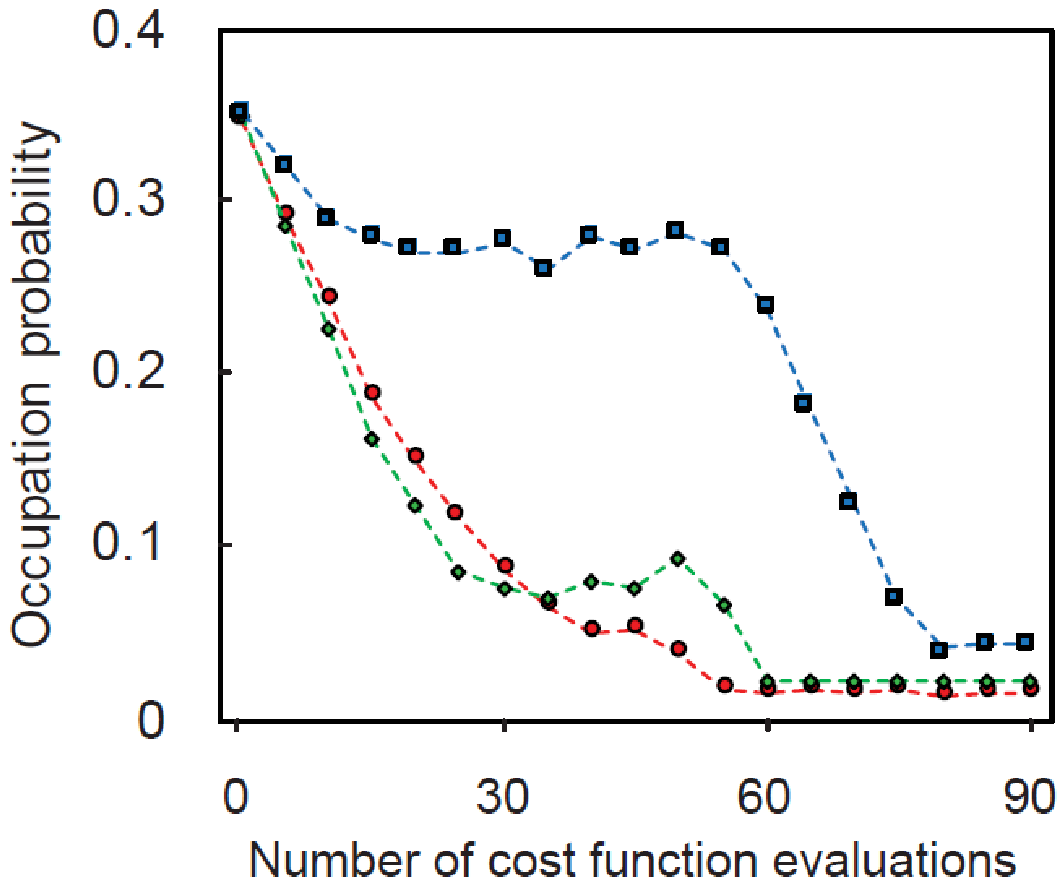
- Optimization routine run with one, two, and three training states
- Twenty random states from same subspace sent through device
- Occupation prob.:
  - 1 state: 0.4 ± 0.3
  - 2 states: 0.03 ± 0.02
  - 3 states: 0.02 ± 0.02



[1]

Results: Robustness

- Test robustness of device by rotating scrambling plate by 4° every 5 steps
- Needed training time increased
- Encoding still quite good ($p < 0.05$)



[1]

Conclusion

- Autoencoders provide flexible compression of information
- Experimental realization successful
  - Device is able to exploit underlying structure of dataset
  - Robust to drift in subspace
  - Can be extended beyond qutrits to encode arbitrarily large qudits

- Possible improvement
  - Faster implementation of unitaries
  - More sophisticated optimization algorithm (e.g. genetic algorithms)

Personal remarks

- Skeptical about number of training samples
  - Heavy dependence on chosen training states
- Still used additional knowledge to avoid local minima
  - Not task specific
- Gradient estimation is very crude
  - Could fail on more complicated problems
- Not as powerful as Neural Networks
  - No non-linearities

- General result
  - Impressing implementation
  - Very promising for future applications

Sources

- [1] Experimental Realization of a Quantum Autoencoder: The Compression of Qutrits via Machine Learning https://arxiv.org/abs/1810.01637
- [2] Autoencoder demo: https://github.com/HackerPoet/FaceEditor
- [3] Quantum optics of the beamsplitter http://www.quantum.physik.uni-potsdam.de/teaching/ss2013/qo2/script_bsplit.pdf
- [4] https://en.wikipedia.org/wiki/Autoencoder
- [5] https://en.wikipedia.org/wiki/White_noise
- [6] Drawn by myself in MS Paint