Impacts of parameter drift and induced decoherence in entangled quantum link generation on RL-based policy performance

Radu Ionut Ciobanu

Introduction

- Multiple protocols exist to generate entangled links.
- The main trade-off between them is between the fidelity of the link and the chance of successfully generating it.
- Choosing which protocol to use to generate a link is not a trivial task.
- These links also decay over time in memory.
- It is usually assumed that the rate at which these links decay is constant.
- In reality it changes over time in a process called parameter drift.
- Another source of instability is a phenomenon called induced decoherence: the more links stored in memory, the faster they decay.
- We are interested in investigating the effectiveness of reinforcement learning at solving this problem.
- We are also interested in how robust the models trained at when facing the instabilities described.

To investigate this, the following research sub-questions are posed:

1. **How to model entangled link generation protocol selection as a reinforcement learning problem?**
2. **How to model decay rate drift as a reinforcement learning problem?**
3. **How to model induced decoherence as a reinforcement learning environment?**
4. **How do the agents trained on the models constructed in questions 1-3 compare to one another in performance and training time?**

Methods

Four different RL environments were developed. All of these share the following parameters:

The list of protocols which represents the action space, the threshold fidelity below which the links become useless, and the number of links that have to be generated.

The environments are:

1. **Discrete state space**

The first environment models the state space as the expected lifetime of every link in memory. This environment was only used during hyperparameter tuning and initial testing, and as a technical benchmark for the others.

2. **Continuous state space, static decay rate**

This environment is used as a control for the experiment. It has a constant decay rate for the stored links.

3. **Continuous state space, time dependent decay rate (parameter drift)**

This environment has an additional parameter: an arbitrary time evolution function for the link decay rate.

4. **Continuous state space, memory dependent decay rate (induced decoherence)**

This environment is similar to the last, however the function now depends on the number of stored links in memory instead of time.

Deep Q-Learning models were trained on each environment, and then every model was evaluated on each environment.

Two different static decay rate models were used as control cases, one with a low decay rate and one with a high decay rate. The dynamic decay rate environments used a linear interpolation function between the two control values.

Results and Conclusion

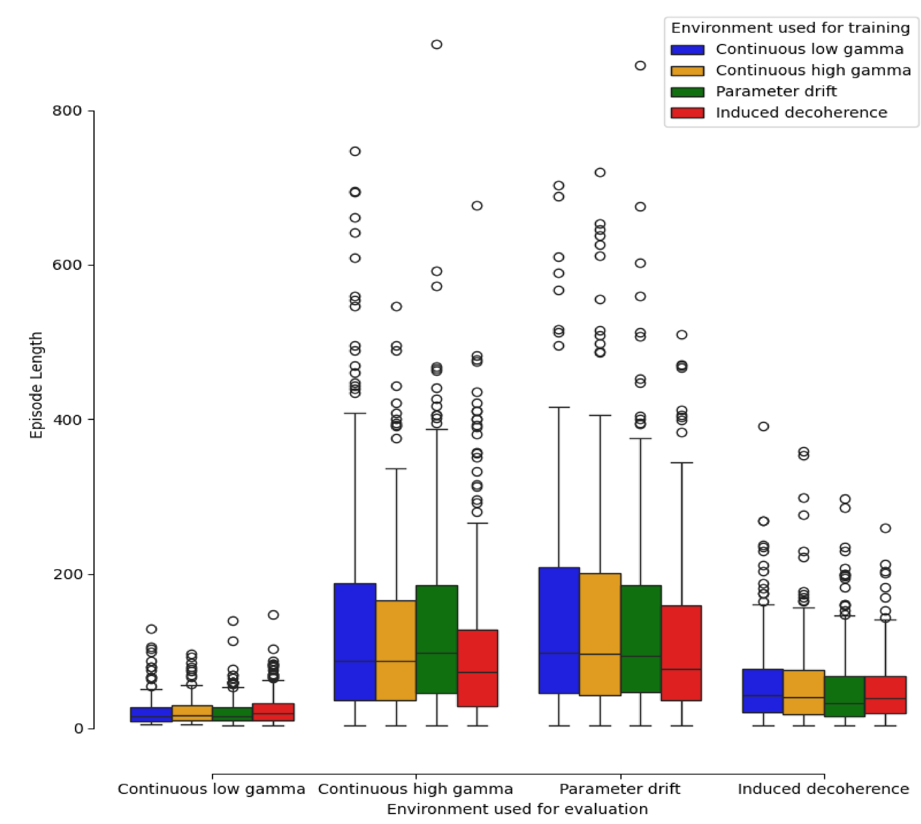

Figure 1: Box plots of the performance of every model in every environment

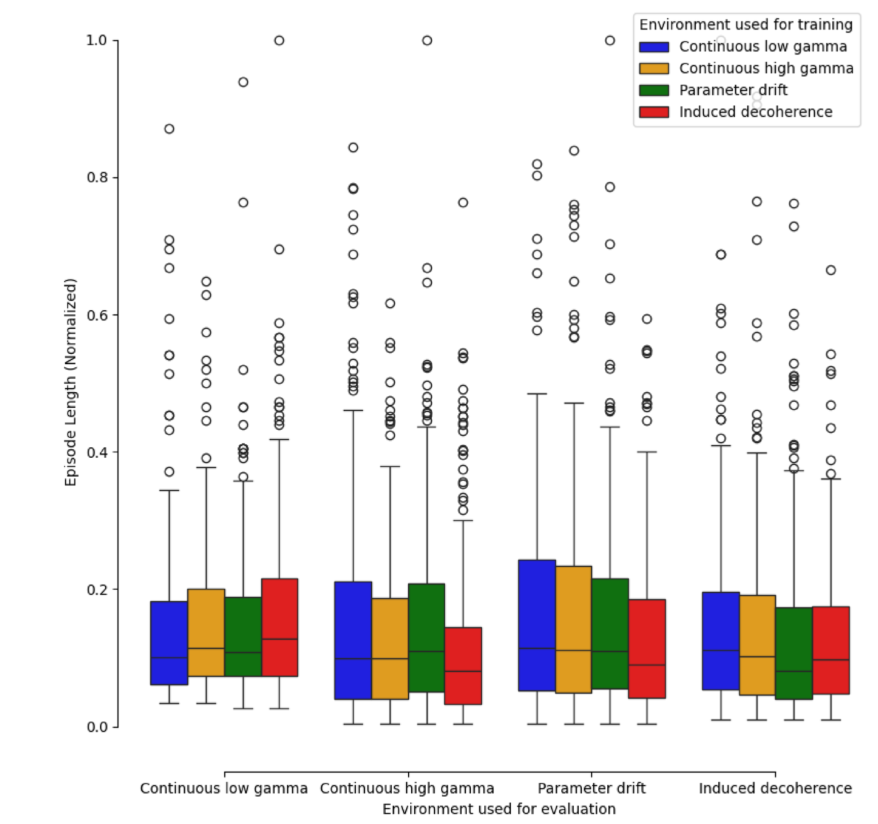

Figure 2: Normalized data of Figure 1. Note the similar performance of each model.

Little relative difference was noticed between the four models. We feel that this is a sign that our approach is robust and capable of handling the unstable environments described.

However we feel that the following limitations must be acknowledged:

- The experiment uses only "easy" environments (low number of required links). Perhaps a difference does exist in performance but it is drowned out by the inherent randomness of the system.
- The mathematical models used for parameter drift and induced decoherence are simple prototypes. An in-depth theoretical investigation of these phenomena is beyond the scope of our work.

We believe our work shows that there is potential in a RL-based solution to the protocol selection problem. We present a solid proof of concept, however larger scale experimentation is required to confirm our results.