Online-learning tracking of Qubit control parameters under jump-diffusion type drift

Oswin Krause *Department of Computer Science, University of Copenhagen, 2100 Copenhagen, Denmark* oswin.krause@di.ku.dk
Fabrizio Berritta *Engineering Quantum Systems Group, Massachusetts Institute of Technology, 2139 Camebridge, USA* fb7@mit.edu
Ferdinand Kuemmeth *Institute of Experimental and Applied Physics, University of Regensburg, 93040 Regensburg, Germany, Germany*
Jacob Hastrup and **Morten Kjaergaard** *Center for Quantum Devices, Niels Bohr Institute, University of Copenhagen, 2100 Copenhagen, Denmark*

Abstract

Controlling a quantum computer requires the estimation of a number of parameters for each qubit. Since the parameters fluctuate quickly over time, there is a need for fast and efficient online estimation algorithms that can adapt the measurement parameters to gain as much information as possible. We present our ongoing work of developing an online Bayesian estimation protocol for estimation of the decay-time of a transmon Qubit and extend it to handle jump-diffusion models. We apply the algorithms both to simulated data with known ground truth as well as real-world data.
Keywords: Online-learning, Bayesian Tracking, Quantum Devices, High-frequency Prediction, FPGA

1 Introduction

Controlling a quantum computer requires the estimation of a large number of parameters for each qubit, for example qubit rotation frequencies, but also life-time parameters like qubit decoherence times. Due to quantum fluctuations in the environment of a qubit, these parameters change over time and are correlated on time-scales from sub-milliseconds to hours and days, see Müller et al. [2015], Carroll et al. [2022]. These fluctuations are believed to arise due to the coupling of the qubit to other nearby two-level systems (e.g., an electron repeatedly moving between two locations), see Murray [2021] and are believed to be an intrinsic property of the material. As a result, changes to the parameters are not continuous, but rather the combined effects of many discrete jump events.

As the jump events cannot be observed directly, algorithms for parameter estimation can also serve as a *sensor* that provides insight into the material quality. Better materials have less noisy environments and thus should lead to more stable device parameters. Current algorithms only provide limited insights into the process. For example Klimov et al. [2018] track the fluctuations of qubit lifetimes over several hours, but obtaining a single parameter estimate required several seconds. This is far too long to provide insight into the low time-scales of interest for qubit operations.

Existing algorithms are slow because obtaining the measurements themselves often requires an estimate for the parameter of interest - to obtain an estimate for a qubit rotation frequency, we need to rotate qubits. However, since qubit measurements result in binary outcomes, the wrong choice of parameters can lead to uninformative results, when the probability to observe one state is almost 1 due to the choice of parameters. Thus, classical approaches that perform measurements with parameters evaluated on a grid necessarily evaluate a large number of uninformative settings.

As a result, AI driven online-learning algorithms are needed that not only continuously update their estimate but also choose informative measurement parameters. However, the real-time demands of the task require the algorithms to run in the order of a few microseconds, and thus are implemented via FPGA, which limits algorithm complexity. There have been a number of approaches proposed for these algorithms, many of which follow the principle of Bayesian estimation, for example see Gebhart et al. [2023]. Still, these algorithms usually assume that the parameter is constant and thus need to be restarted to allow tracking.

In the following, we will present our recently developed algorithm for estimating the decay-time T_1 of the excited state of a transmon qubit (see, Berritta et al. [2025]) that improved estimation frequency of qubit parameters by 2 orders of magnitude and allowed the discovery of Lorentzian type noise on high-frequency time-scales. We further extend it to Bayesian tracking by including a drift model. We incorporate both a model of continuous drift, representing fast noise components, as well as a jump noise model, for the estimation of high amplitude jumps. We demonstrate the applicability of the algorithm on simulated data as well as on a real transmon qubit dataset.

2 Methods

For the basic algorithm, let $\lambda = 1/T_1$ be the unknown inverse decay time. After initialisation of the qubit to the excited state $s_0 = 1$, we wait for time T. During this time, the qubit might have decayed to the ground state, i.e. $s_T = 0$. In the absence of noise, the time to decay follows an exponential distribution, which means that the decay probability within time T is $P(s_T = 0) = 1 - \exp(-\lambda T)$. We then obtain a measurement m of s_T. This procedure can be corrupted by state preparation and measurement errors, which are modelled by error probabilities α and β leading to the observational model

$$P(m = 1|T, \lambda) = \beta + (1 - \beta - \alpha)e^{-\lambda T} \ . \tag{1}$$

Using a Bayesian approach starting from a prior $p_0(\lambda_0) = \Gamma(\lambda_0; \kappa, \theta)$, where θ is the inverse scale parameter, the algorithm progresses in iterations where in the kth iteration a waiting time T_k is computed and a measurement m_k is performed. After this, the algorithm computes a posterior approximation $p_k(\lambda_k)$. Computing the exact posterior $p_k(\lambda_k|m_k, T_k) \propto P(m_k|T_k, \lambda_k)p_{k-1}(\lambda_k)$ is not feasible within the device limitations, and therefore it is assumed that the posterior distribution resembles a $\Gamma(\lambda_k; \kappa, \theta_k)$-distribution, which is fit via moment matching. Based on minimum variance considerations, the measurement time $T_k = 0.5/E[\lambda_{k-1}]$, $\lambda_{k-1} \sim \Gamma(\kappa_{k-1}, \theta_{k-1})$ is chosen.

To extend this model, we include a drift term $p(\lambda_k|\lambda_{k-1}, T_k)$, where the dependence on T_k includes the effects of the drift while a measurement is performed. The resulting probabilistic model follows the Bayesian tracking formulation and

$$p_k(\lambda_k|m_k) \propto P(m=1|T_k, \lambda_k) \underbrace{\int p(\lambda_k|\lambda_{k-1}, T_k)p_{k-1}(\lambda_{k-1})\, d\lambda_{k-1}}_{p_k^{\text{pred}}(\lambda_k)}, \quad (2)$$

To model both jumps and continuous drift, we model $p(\lambda_k|\lambda_{k-1}, T_k)$ as a mixture distribution, where we assume that λ_k is drawn from p_0 when a jump happens and follows a continuous drift model otherwise. The probability of jump follows an exponential distribution $p(\text{jump}|T) = 1 - \exp(-\gamma T)$, where γ is a rate parameter. For the continuous drift, we use the SDE proposed in Bibby et al. [2005] which creates a stochastic process with autocorrelation time ξ and stable distribution $\Gamma(\kappa_0, \theta_0)$. We use a one-step Euler integration to the SDE, compute the mean and variance of the resulting random variable and perform moment matching to obtain another Γ-distribution. Since the jumps adds the prior as another mixture component in every iteration, this model is difficult to implement on an FPGA. To make inference tractable, we only consider jumps at every Mth iteration, and we only allow a maximum of two mixture components in the posterior $p_k(\lambda_k) = (1-w_k)p_k^{\text{cont}}(\lambda_k) + w_k p_k^{\text{jump}}(\lambda_k)$. When a third component is added, we distribute the probability mass w_k of p_k^{jump} between $p_k^{\text{cont}}(\lambda_k)$ and the new component p_0 and perform moment matching on each pair. We found that only moving 10% on the mass to p_k^{cont} led to better results.

3 Experiments & Results

We perform two sets of experiments, the first is a simulated dataset with known baseline and the second is the application of the estimator to the dataset gathered in Berritta et al. [2025]. To be comparable, we also use $\hat{T}_1 = E[1/\lambda_k]$ as prediction and $\Gamma(3, 450)$ as prior in all experiments.

For the first experiment, we simulated a jump-diffusion process and compared the Bayesian estimation algorithm without drift-model and without restarting (baseline), to the model with only using continuous drift and the one using both continuous and jump diffusion. We chose $\alpha = \beta = 0.01$. We fitted the model parameters on the simulation using grid-search and for the jump-model we merged the mixture components every $M = 50$ iterations. We then ran

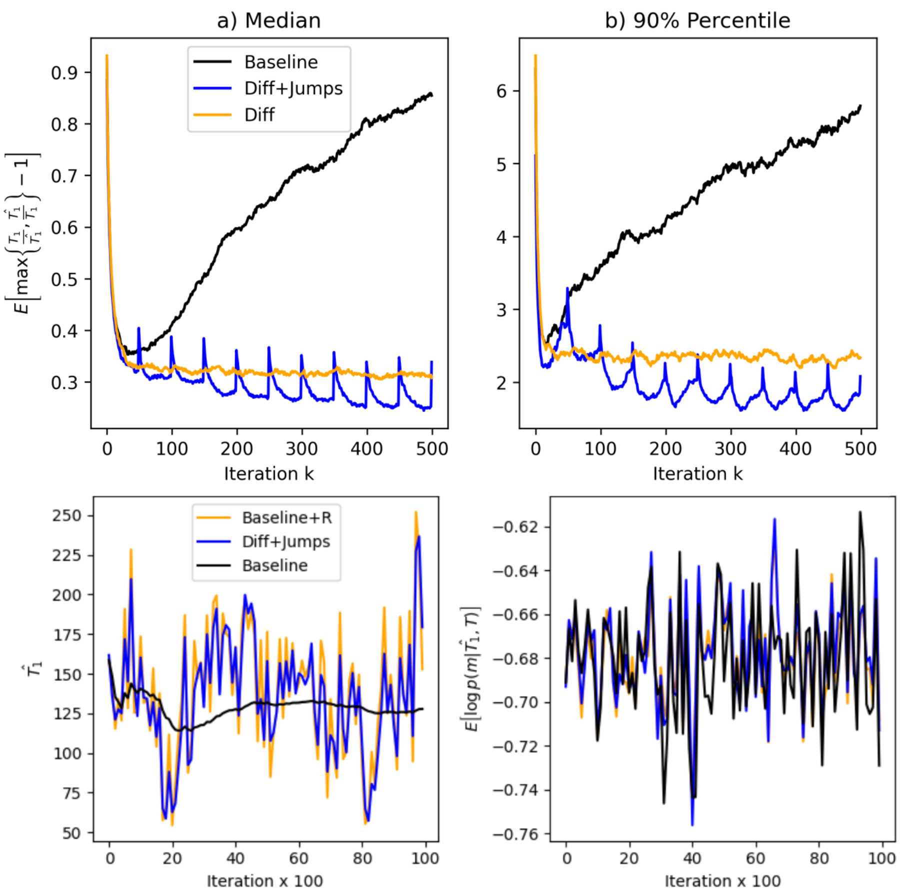

Figure 1: fig:Preliminary results. a) b) show simulated results, while c,d shows results on real Qubit data. See text for details

10000 simulations for each algorithm for 500 iterations. In Figure 1(a,b), we show the median and 90% percentile relative error between the true simulated T_1 and the prediction $\hat{T}_1$ over the iterations. As one can see, the baseline without restarting performs well for the first 50 iterations, after which a restart is optimal. The continuous drift model is able to follow the trend with stable error rate, but does not improve significantly on the error due to the required fast drift to catch jumps. Lastly, for the full model we see a pattern of spikes after every 50 iterations, due to merging, but can improve over the other models afterwards.

In the second experiment, we use the data gathered by the baseline algorithm, implemented on a Quantum Machine OPX FPGA. It was run for 100 iterations with error rates $\alpha = \beta = 0.12$ before restarting. We merge the dataset of 100 consecutive runs of the algorithm and compute trajectories of $\hat{T}_1$ using our algorithms based on the concatenated sequences and compute the data log-likelihood using $\hat{T}_1$ as estimate. In Figure 1(c), we show example trajectories of the Baseline with and without restarting, compared to our jump-diffusion model for one example trajectory. The algorithm without restarting slows down, while the other trajectories closely align. However, the log-likelihood estimate in Figure 1(d) does not show any difference between the algorithms. We hypothesize that this is likely due to low informative of samples due to high error rates and restarts leading to uninformative T_k.

4 Conclusions

Our preliminary work showed that online-learning based algorithms can inform physics even in restricted quantum computing settings. We further show that including drift models into the estimation process can improve over restarting on simulated data, however we failed to see the same improvement on real data. We propose to use less aggressive restarting strategies, not starting from the prior, but taking the last estimate into account. Similarly, for estimation purposes, running the algorithm for longer before restarting might allow us to collect more informative data.

5 Acknowledgments

FB, JH and MK gratefully acknowledge support from the Novo Nordisk Foundation (grant number NNF22SA0081175, the NNF Quantum Computing Programme), the Villum Foundation through a Villum Young Investigator grant (grant 37467), the European Union through an ERC Starting Grant (NovADePro, 101077479), the Innovation Fund Denmark (DanQ, Grant No. 2081-00013B), and the U.S. Army Research Office (Grant No. W911NF-22-1-0042). FB and FK acknowledge funding from the European Union's Horizon 2020 research and innovation programme under grant agreements 101017733 (QuantERA II), 101204890 (HORIZON-MSCA-2024-PF-01) and EUREKA Eurostars

3 (ECHIDNA), and from the Dutch National Growth Fund (NGF) as part of the Quantum Delta NL programme and the Research Council of Norway (RCN) under INTFELLES-Project No. 333990. OK received funding via the Innovation Fund Denmark for the project DIREC (9142-00001B). The device was fabricated at Myfab Chalmers. The Chalmers team was funded by the Knut and Alice Wallenberg Foundation through the Wallenberg Center for Quantum Technology (WACQT) and the EU Flagship on Quantum Technology HORIZON-CL4-2022-QUANTUM-01-SGA project 101113946 OpenSuperQPlus100. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Army Research Office, the US Government, the European Union, or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

F. Berritta, J. Benestad, J. A. Krzywda, O. Krause, M. A. Marciniak, S. Krøjer, C. W. Warren, E. Hogedal, A. Nylander, I. Ahmad, et al. Real-time adaptive tracking of fluctuating relaxation rates in superconducting qubits. *arXiv preprint arXiv:2506.09576*, 2025.

B. M. Bibby, I. M. Skovgaard, and M. Sørensen. Diffusion-type models with given marginal distribution and autocorrelation function. *Bernoulli*, 11(2): 191–220, 2005.

M. Carroll, S. Rosenblatt, P. Jurcevic, I. Lauer, and A. Kandala. Dynamics of superconducting qubit relaxation times. *npj Quantum Information*, 8:132, Nov. 2022. ISSN 2056-6387. doi: 10.1038/s41534-022-00643-y.

V. Gebhart, R. Santagati, A. A. Gentile, E. M. Gauger, D. Craig, N. Ares, L. Banchi, F. Marquardt, L. Pezzè, and C. Bonato. Learning quantum systems. *Nature Reviews Physics*, 5(3):141–156, 2023. doi: https://doi.org/10.1038/s42254-022-00552-1.

P. V. Klimov, J. Kelly, Z. Chen, M. Neeley, A. Megrant, B. Burkett, R. Barends, K. Arya, B. Chiaro, Y. Chen, et al. Fluctuations of energy-relaxation times in superconducting qubits. *Physical review letters*, 121(9):090502, 2018.

C. E. Murray. Material matters in superconducting qubits. *Materials Science and Engineering: R: Reports*, 146:100646, Oct. 2021. ISSN 0927-796X. doi: 10.1016/j.mser.2021.100646.

C. Müller, J. Lisenfeld, A. Shnirman, and S. Poletto. Interacting two-level defects as sources of fluctuating high-frequency noise in superconducting circuits. *Physical Review B*, 92(3):035442, July 2015. ISSN 1550-235X. doi: 10.1103/physrevb.92.035442.