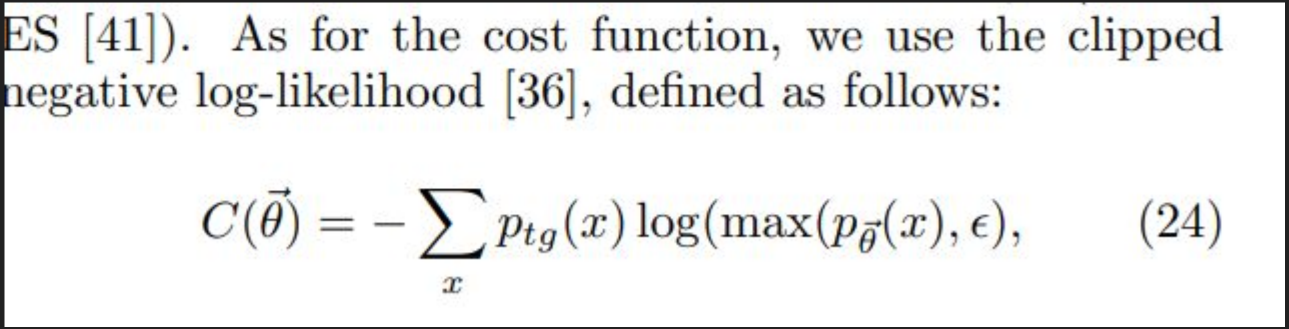
- Explain CVA

The credit value adjustment is the difference between the risk-free price of a netting set and the the price which takes the possibility of the default of the counterparty into account. A netting set is a portfolio of deals with one counterparty for which you have a netting agreement. That means you are allowed to set positive against negative market values. A netting agreement will reduce your exposure and therefore the counterparty credit risk.

- Explain Asset price dynamics

1. The probability distribution of asset prices P(s|t) at time t. Classically, for a given future time t = τ , sampling from P(s|t = τ) is typically achieved by performing stochastic simulation of how the asset price s fluctuates as a function of time up to τ , and the set of final price values is the set of samples from P(s|t = τ). One of the most common stochastic processes used for modeling price fluctuation is geometric Brownian motion.

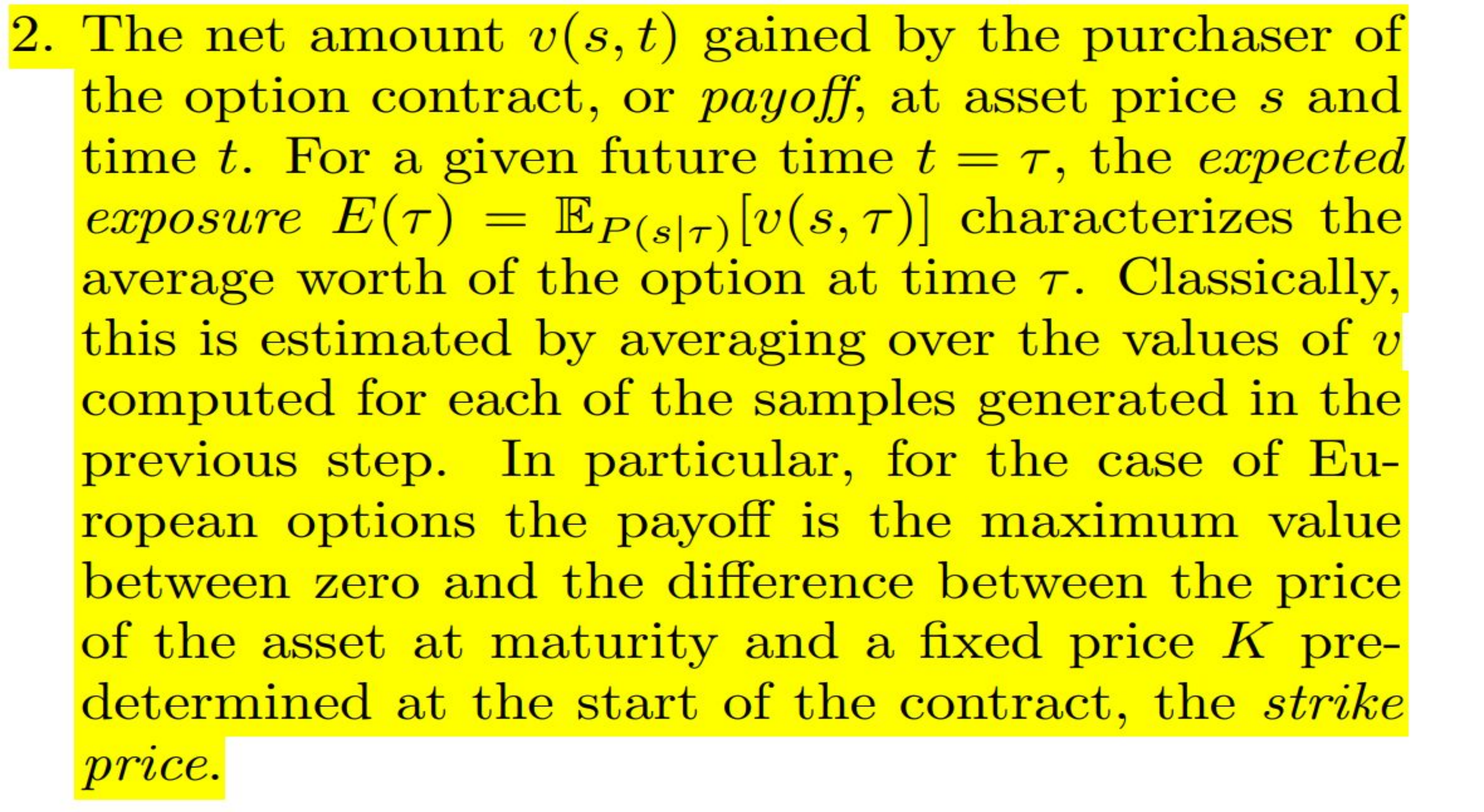
- The cost function of Asset price dynamics(equation 24)



ES [41]). As for the cost function, we use the clipped negative log-likelihood [36], defined as follows:

$$C(\vec{\theta}) = -\sum_{x} p_{tg}(x) \log(\max(p_{\vec{\theta}}(x), \epsilon), \qquad (24)$$

- The payoff function(Section II point 2)



2. The net amount $v(s,t)$ gained by the purchaser of the option contract, or *payoff*, at asset price s and time t. For a given future time $t = \tau$, the *expected exposure* $E(\tau) = \mathbb{E}_{P(s|\tau)}[v(s,\tau)]$ characterizes the average worth of the option at time τ. Classically, this is estimated by averaging over the values of v computed for each of the samples generated in the previous step. In particular, for the case of European options the payoff is the maximum value between zero and the difference between the price of the asset at maturity and a fixed price K predetermined at the start of the contract, the *strike price*.

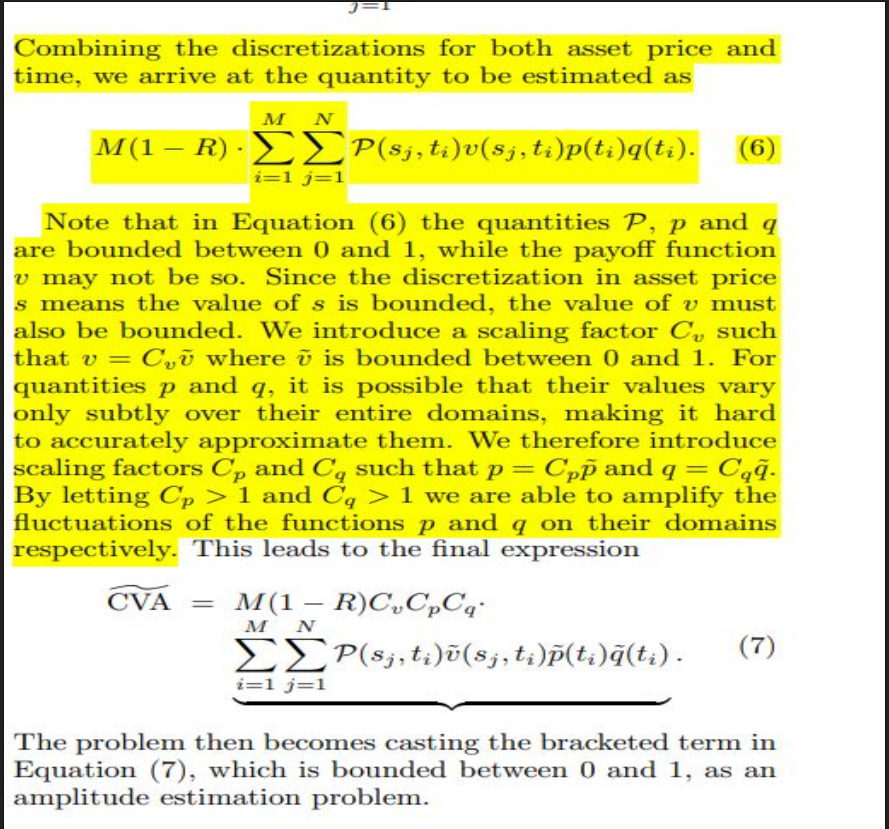
- The probability of default(section II point 3)

The probability of default q(t) at time t. One method for modeling q(t) is to consider it as a Poisson process where the Poisson parameter is time dependent. Its exact time dependence can be bootstrapped efficiently, and calibrated from market quantities such as CDS spreads.

- Discount factor(Section II point 3)

The discount factor p(t). It expresses the time value of money, and it is used to determine the present value of an asset in the future. The formula for the discount factor will depend on the number of periods considered for interest rate payments, where a typical choice is as continuous compound interest, which corresponds to discount factor p(t) = e^(−rt) for an interest rate r. The interest rate can also be time dependent.

- Explanation of measure of the projection operator (equation 7 and equation 19)



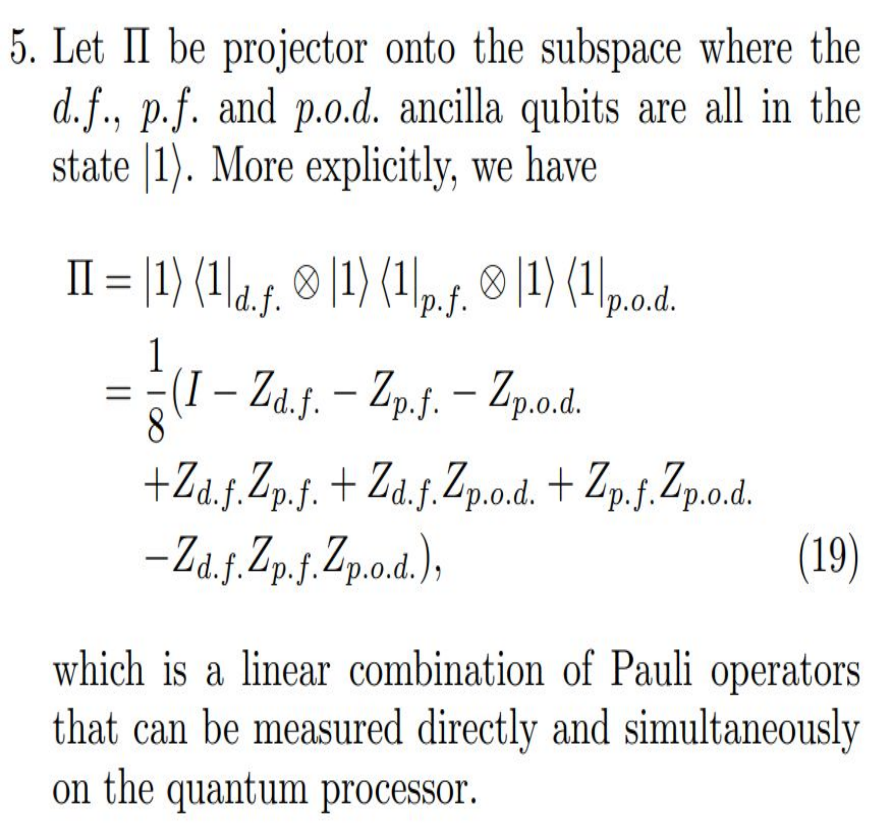
Combining the discretizations for both asset price and time, we arrive at the quantity to be estimated as

$$M(1-R)\cdot\sum_{i=1}^{M}\sum_{j=1}^{N}\mathcal{P}(s_j,t_i)v(s_j,t_i)p(t_i)q(t_i). \qquad (6)$$

Note that in Equation (6) the quantities $\mathcal{P}$, p and q are bounded between 0 and 1, while the payoff function v may not be so. Since the discretization in asset price s means the value of s is bounded, the value of v must also be bounded. We introduce a scaling factor C_v such that $v = C_v\tilde{v}$ where $\tilde{v}$ is bounded between 0 and 1. For quantities p and q, it is possible that their values vary only subtly over their entire domains, making it hard to accurately approximate them. We therefore introduce scaling factors C_p and C_q such that $p = C_p\tilde{p}$ and $q = C_q\tilde{q}$. By letting $C_p > 1$ and $C_q > 1$ we are able to amplify the fluctuations of the functions p and q on their domains respectively. This leads to the final expression

$$\begin{aligned}\widetilde{\text{CVA}} &= M(1-R)C_vC_pC_q\cdot\\ &\underbrace{\sum_{i=1}^{M}\sum_{j=1}^{N}\mathcal{P}(s_j,t_i)\tilde{v}(s_j,t_i)\tilde{p}(t_i)\tilde{q}(t_i)}\,. \qquad (7)\end{aligned}$$

The problem then becomes casting the bracketed term in Equation (7), which is bounded between 0 and 1, as an amplitude estimation problem.



5. Let Π be projector onto the subspace where the *d.f.*, *p.f.* and *p.o.d.* ancilla qubits are all in the state $|1\rangle$. More explicitly, we have

$$\begin{aligned}\Pi &= |1\rangle\langle 1|_{d.f.}\otimes|1\rangle\langle 1|_{p.f.}\otimes|1\rangle\langle 1|_{p.o.d.}\\ &= \frac{1}{8}(I - Z_{d.f.} - Z_{p.f.} - Z_{p.o.d.}\\ &\quad + Z_{d.f.}Z_{p.f.} + Z_{d.f.}Z_{p.o.d.} + Z_{p.f.}Z_{p.o.d.}\\ &\quad - Z_{d.f.}Z_{p.f.}Z_{p.o.d.}), \qquad (19)\end{aligned}$$

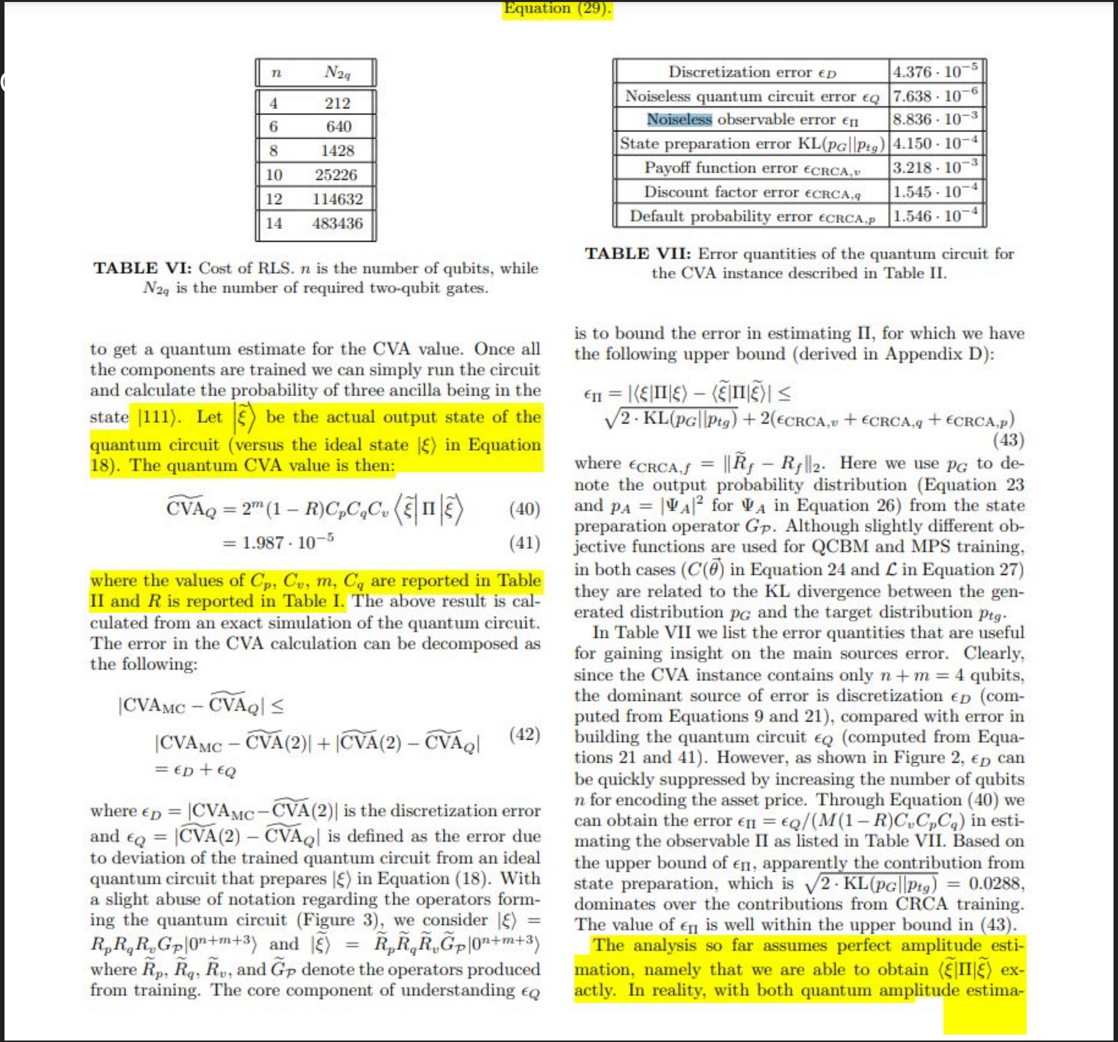
which is a linear combination of Pauli operators that can be measured directly and simultaneously on the quantum processor.

- Noiseless CVA measurement, the values of C_v, C_p, C

Scaling constant for payoff C_v =1.8201814

Scaling constant for default probability C_q= 0.0002038

Scaling constant for discount factor C_p=1



Equation (29).

n	N_{2q}
4	212
6	640
8	1428
10	25226
12	114632
14	483436

TABLE VI: Cost of RLS. n is the number of qubits, while N_{2q} is the number of required two-qubit gates.

Discretization error ϵ_D	$4.376 \cdot 10^{-5}$
Noiseless quantum circuit error ϵ_Q	$7.638 \cdot 10^{-6}$
Noiseless observable error ϵ_Π	$8.836 \cdot 10^{-3}$
State preparation error $\mathrm{KL}(p_G \| p_{tg})$	$4.150 \cdot 10^{-4}$
Payoff function error $\epsilon_{\mathrm{CRCA},v}$	$3.218 \cdot 10^{-3}$
Discount factor error $\epsilon_{\mathrm{CRCA},q}$	$1.545 \cdot 10^{-4}$
Default probability error $\epsilon_{\mathrm{CRCA},p}$	$1.546 \cdot 10^{-4}$

TABLE VII: Error quantities of the quantum circuit for the CVA instance described in Table II.

to get a quantum estimate for the CVA value. Once all the components are trained we can simply run the circuit and calculate the probability of three ancilla being in the state $|111\rangle$. Let $\left|\tilde{\xi}\right\rangle$ be the actual output state of the quantum circuit (versus the ideal state $|\xi\rangle$ in Equation 18). The quantum CVA value is then:

$$\widetilde{\mathrm{CVA}}_Q = 2^m(1-R)C_pC_qC_v \left\langle \tilde{\xi} \middle| \Pi \middle| \tilde{\xi} \right\rangle \tag{40}$$

$$= 1.987 \cdot 10^{-5} \tag{41}$$

where the values of C_p, C_v, m, C_q are reported in Table II and R is reported in Table I. The above result is calculated from an exact simulation of the quantum circuit. The error in the CVA calculation can be decomposed as the following:

$$\begin{aligned} &|\mathrm{CVA}_{\mathrm{MC}} - \widetilde{\mathrm{CVA}}_Q| \leq \\ &\quad |\mathrm{CVA}_{\mathrm{MC}} - \widetilde{\mathrm{CVA}}(2)| + |\widetilde{\mathrm{CVA}}(2) - \widetilde{\mathrm{CVA}}_Q| \\ &\quad = \epsilon_D + \epsilon_Q \end{aligned} \tag{42}$$

where $\epsilon_D = |\mathrm{CVA}_{\mathrm{MC}} - \widetilde{\mathrm{CVA}}(2)|$ is the discretization error and $\epsilon_Q = |\widetilde{\mathrm{CVA}}(2) - \widetilde{\mathrm{CVA}}_Q|$ is defined as the error due to deviation of the trained quantum circuit from an ideal quantum circuit that prepares $|\xi\rangle$ in Equation (18). With a slight abuse of notation regarding the operators forming the quantum circuit (Figure 3), we consider $|\xi\rangle = R_pR_qR_vG_\mathcal{P}|0^{n+m+3}\rangle$ and $|\tilde{\xi}\rangle = \tilde{R}_p\tilde{R}_q\tilde{R}_v\tilde{G}_\mathcal{P}|0^{n+m+3}\rangle$ where $\tilde{R}_p$, $\tilde{R}_q$, $\tilde{R}_v$, and $\tilde{G}_\mathcal{P}$ denote the operators produced from training. The core component of understanding ϵ_Q is to bound the error in estimating Π, for which we have the following upper bound (derived in Appendix D):

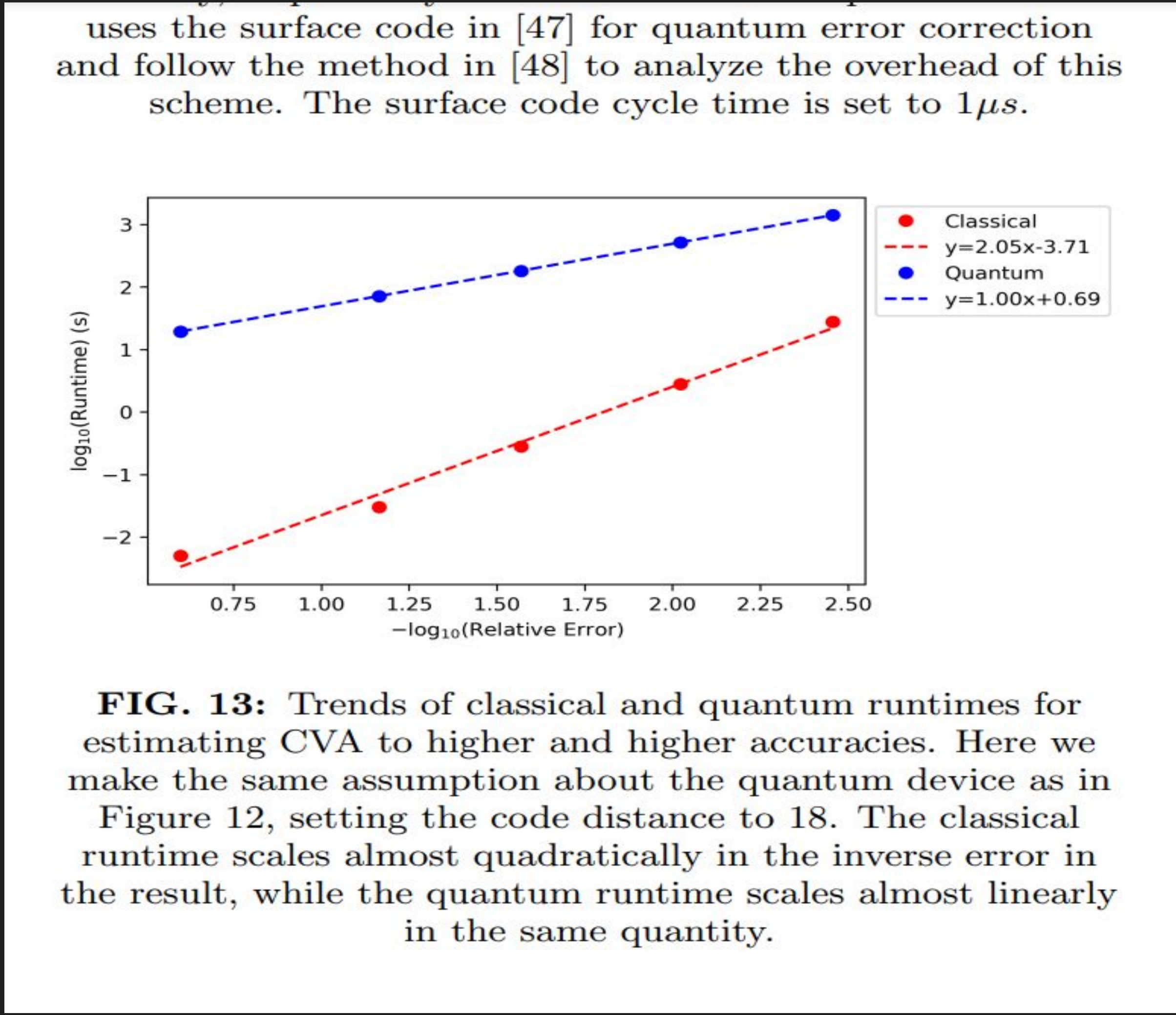
$$\begin{aligned} \epsilon_\Pi &= |\langle\xi|\Pi|\xi\rangle - \langle\tilde{\xi}|\Pi|\tilde{\xi}\rangle| \leq \\ &\sqrt{2 \cdot \mathrm{KL}(p_G||p_{tg})} + 2(\epsilon_{\mathrm{CRCA},v} + \epsilon_{\mathrm{CRCA},q} + \epsilon_{\mathrm{CRCA},p}) \end{aligned} \tag{43}$$

where $\epsilon_{\mathrm{CRCA},f} = \|\tilde{R}_f - R_f\|_2$. Here we use p_G to denote the output probability distribution (Equation 23 and $p_A = |\Psi_A|^2$ for Ψ_A in Equation 26) from the state preparation operator $G_\mathcal{P}$. Although slightly different objective functions are used for QCBM and MPS training, in both cases ($C(\vec{\theta})$ in Equation 24 and $\mathcal{L}$ in Equation 27) they are related to the KL divergence between the generated distribution p_G and the target distribution p_{tg}.

In Table VII we list the error quantities that are useful for gaining insight on the main sources error. Clearly, since the CVA instance contains only $n + m = 4$ qubits, the dominant source of error is discretization ϵ_D (computed from Equations 9 and 21), compared with error in building the quantum circuit ϵ_Q (computed from Equations 21 and 41). However, as shown in Figure 2, ϵ_D can be quickly suppressed by increasing the number of qubits n for encoding the asset price. Through Equation (40) we can obtain the error $\epsilon_\Pi = \epsilon_Q/(M(1-R)C_vC_pC_q)$ in estimating the observable Π as listed in Table VII. Based on the upper bound of ϵ_Π, apparently the contribution from state preparation, which is $\sqrt{2 \cdot \mathrm{KL}(p_G||p_{tg})} = 0.0288$, dominates over the contributions from CRCA training. The value of ϵ_Π is well within the upper bound in (43).

The analysis so far assumes perfect amplitude estimation, namely that we are able to obtain $\langle\tilde{\xi}|\Pi|\tilde{\xi}\rangle$ exactly. In reality, with both quantum amplitude estima-

- The advantage in quantum simulation: Fig 13



uses the surface code in [47] for quantum error correction and follow the method in [48] to analyze the overhead of this scheme. The surface code cycle time is set to 1μs.

FIG. 13: Trends of classical and quantum runtimes for estimating CVA to higher and higher accuracies. Here we make the same assumption about the quantum device as in Figure 12, setting the code distance to 18. The classical runtime scales almost quadratically in the inverse error in the result, while the quantum runtime scales almost linearly in the same quantity.

- Limitations:

- The maximum no. of discretization steps supported is 16(n = log_2(N) = 4 qubits and the maximum no. of times steps is M = 2^m, m = 2, i.e. 4 half years)

- Noisy CVA estimation:

- using elf function we sample through several noiseless CVA evaluations to get the final measurement of CVA which should reduce the noise in the final results to get better accuracy in real hardware)