# self-reflection report on ICCAD-2023-contest submission

```
We are the team QwQ.
```

## solution for the final submission

- framework: Qiskit
- method: (naive) VQE
- optim: COBYLA
- ansatz: CHC (only single excitations)
- init: uniform random of magnitude 1e-5
- qsam_optim: fold rot gates, emit SX-RZ-SX-RZ gates

## thoughts & trails

◯ less gates, less noise

> in a noisy quantum environment, if the evolution CAN NOT be corrected step by step, nothing can stand still in the face of error accumulation
> due to the noise, deeper circuit or more trainable parameters CAN NOT guarantee higher precision
> therefore, shallower circuit & less operations is just that we can bear with

- we choose the `CHC-s` ansatz, it is tiny in trainables and shallow in depth
  - it has the least gate count (even only 6 trainable parameters before circuit decomposition) in the well-known ansatz zoo
  - `CHC-s` achieves nearly the same result compared to `CHC-sd`
  - we do not consider the standard `UCC-sd` though has the best precision, because it is too deep and will **soon crash in a noisy runtime**
- we perform some QSAM optimization at pulse level, which reduces useless operation count
  - fold sequential rot gates to one, i.e. `R(x) + R(y) => R(x+y)`
  - emit the frequently found sequent: `SX + RZ(pi) + SX + RZ(pi) => -jI => I`

◯ unbalanced is better than balanced

> for the local optimzer, decide a better init params for the ansatz

- `zero` init just works, but we found that a bit random disturbance is better than all `zeros` as the init point, in final precision
- that is, we use `uniform` init with a very small variance
- other inits like `uniform` or `normal` with a large variance often leads to failure

◯ train ansatz with noise (might not work)

> what if we optimize the ansatz with noise, will it be like adversarial training?

- this idea lookingly works an Cairo & Montreal noise, but does not imporve much (1e-1~1e-2)
- this idea does NOT work on Kolkata noise, totally

◯ error mitigation (not fully-implemented)

> in the noisy runtime, we do not tend to learn the noisy like some methods
> because learning the noise recursively introduces more noises, and dramatically slows down the simulation
> we prefer the ZNE error mitigation method, that's to repeat the circuit to amplify the noise, and extrapolate the noiseless case

- however, we do not have enough time to run & record the statistics... the personal reason is given in the last of this text

◯ better topology, less SWAPs (not implemented)

> the SWAP gate is heavy, try remapping the qubits and acorrdingly permuting the paulis strings to reduce SWAP in ansatz

- did not have time to dive into the hardware (

Other stories:

- Thaks for the contest, letting me explorer the noisy quantum runtime, which I hardly thought about before
- The contest lasts for 1 month, however, my vacation took away 14 days, so our final submission is all made in a hurry (~7 days)

    - the Qiskit framework is especially slow (do not know why) than PyChemiQ and Mindquantum, which I usually work with
    - still some ideas not fully-implmented, e.g. error mitigation (ZNE)
- Hope me good luck next time (

by Armit
2023/10/10