Quantum cognition nonunitary gates documentation

**Main references:**

[1] Möttönen, Vartiainen, Bergholm & Salomaa (2008) Transformation of quantum states using uniformly controlled rotations

[2] Schlimgen, Head-Marsden, Sager, Narang & Mazziotti (2021) Quantum simulation of open quantum systems using a unitary decomposition of operators

[3] Schlimgen, Head-Marsden, Sager, Narang & Mazziotti (2022) Quantum simulation of the Lindblad equation using a unitary decomposition of operators

1. **Building the propagator**

Using the steps in [2], an open quantum system’s time evolution can be written using the Kraus formalism in operator sum form:

With the Kraus maps and the system density matrix.

Any operator M can be decomposed into a Hermitian and anti-Hermitian component.

A unitary propagator matrix can be prepared as:

This propagator needs to be interpreted as a set of rotation angles angles along some vector.

1. **Efficient gate decomposition circuit**

Then, from this representation ( angles along ) we need to convert to the efficient gate decomposition via M matrix [1]:

The benefit of the “efficient gate” circuit is the one-control, one-target CNOT gates instead of the full binary pattern required before. This allows the CNOT chosen to work on current gate sets. Where

are the binary code and binary reflected Gray code encodings of integer m. E.g m = 5 🡪 and . The Gray code allows for the adjacency of the CNOT gates in Fig. 2. A table of Gray code is shown in Fig. 3.

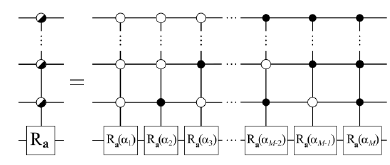


Fig. 1. From [1] for k-fold uniformly controlled rotation about the axis **.** The bottom qubit is the target qubit and black control bit is for 1, white for 0.

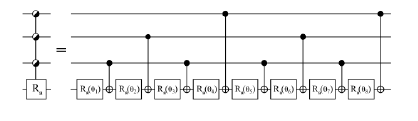


Fig. 2. From [1] for the “efficient gate decomposition” for a sample uniformly controlled rotation about the axis **.** The CNOT gates are now single control-single target and the angles are .

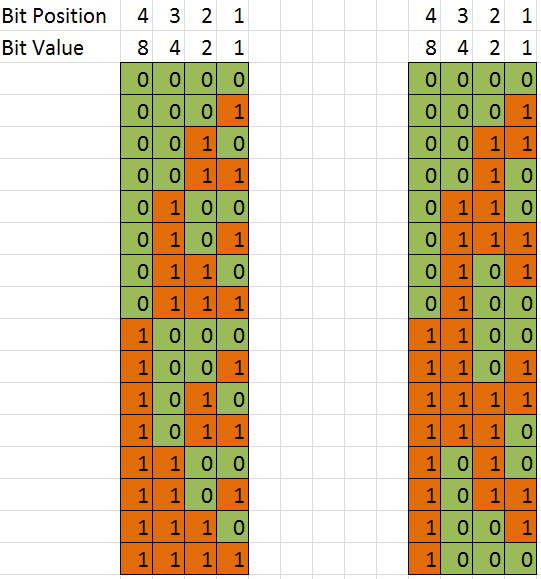


Fig. 3. Binary reflected Gray code (from [here](https://www.google.com/imgres?imgurl=https%3A%2F%2Fcomputingrmurray2.files.wordpress.com%2F2015%2F01%2F2015-01-26_145625.png&imgrefurl=https%3A%2F%2Fcomputingrmurray2.wordpress.com%2F2015%2F01%2F26%2Fbinary-reflected-gray-code%2F&tbnid=r-zdmcO3WVPpkM&vet=12ahUKEwiUjO7t84v7AhUWsXIEHTMNBL4QMygCegUIARDCAQ..i&docid=UU8ne8hkf5XAxM&w=541&h=579&q=binary%20reflected%20gray%20code&client=safari&ved=2ahUKEwiUjO7t84v7AhUWsXIEHTMNBL4QMygCegUIARDCAQ))

So far I have implemented the M matrix in jupyter so we can go from angles to angles. Ref [2] provides an example Kraus operator which I am using (in lieu of the quantum cognition Lindblad model at the moment).

1. **Euler angles for U3 gates**

Finally, we need to convert the angles along **a** vector to Euler angles in order to use Qiskit’s U3 gates. This is the last step that yields the circuit in the form of Fig. 4, ready for Qiskit implementation.

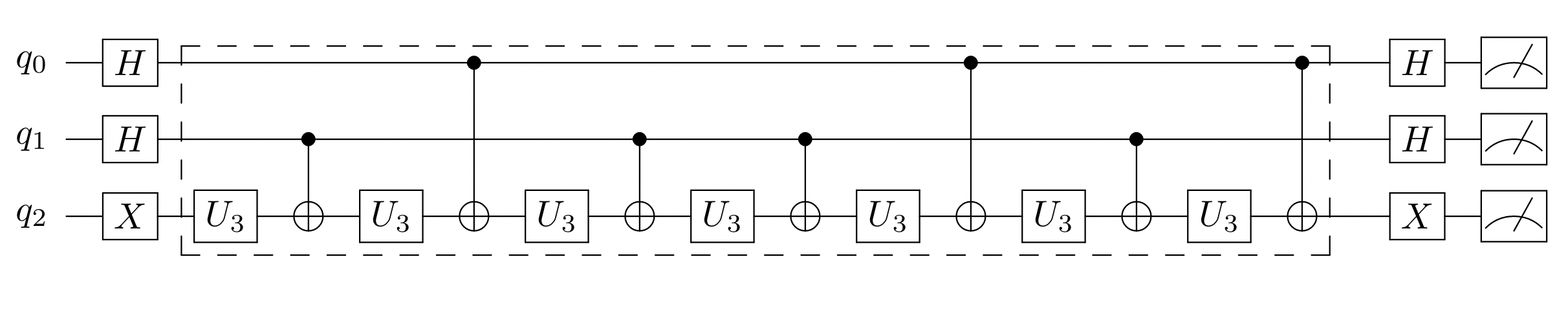


Fig. 4. Desired final circuit configuration from [3] for the nonunitary operators

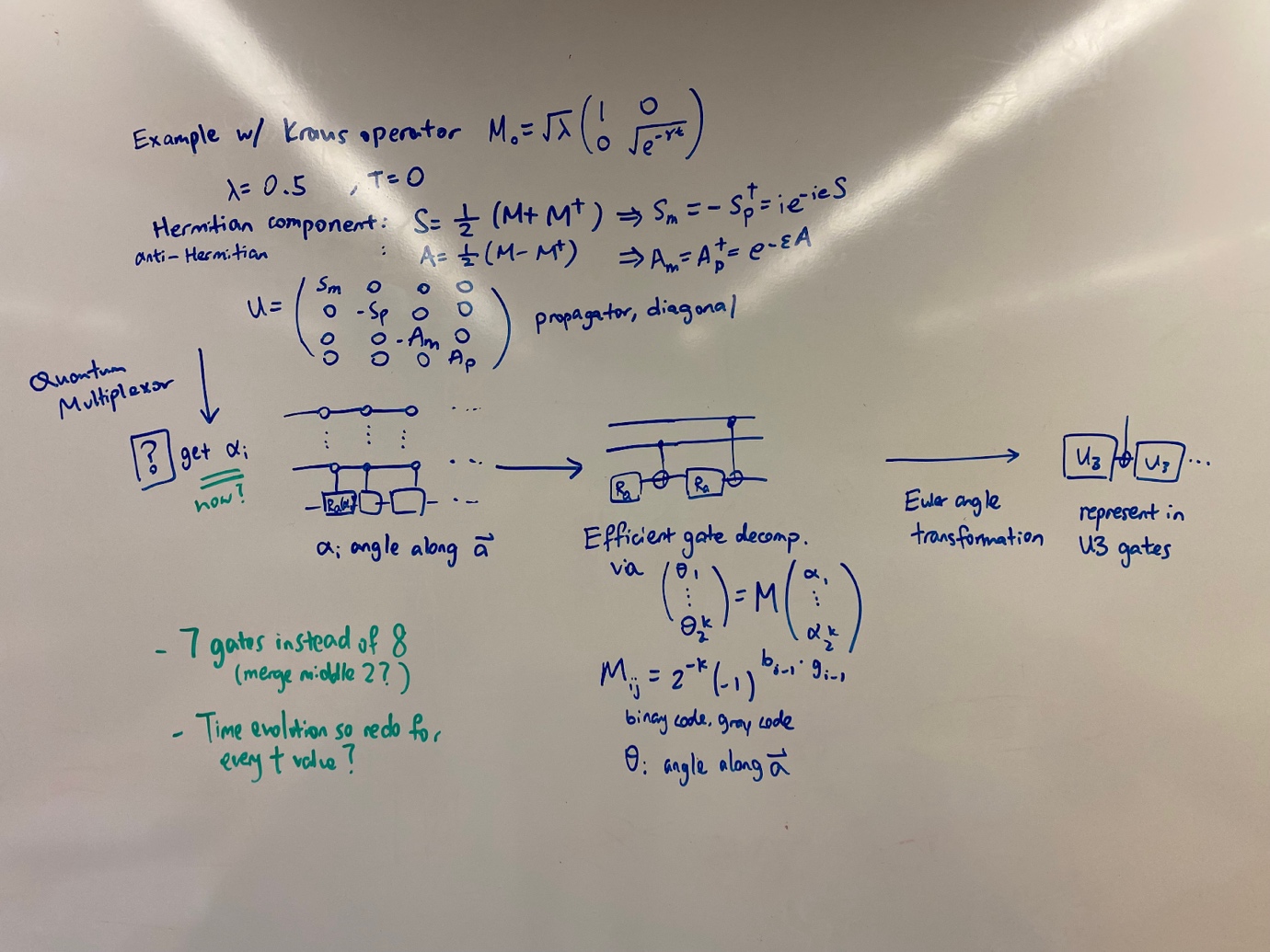


Fig. 5. My own flowchart for the program