



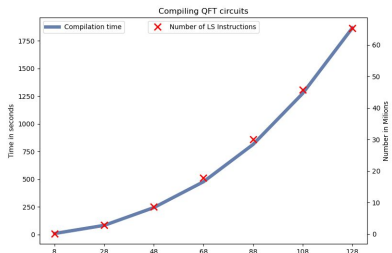
PRESENTER:  
**George Watkins**

**BACKGROUND:** A major challenge with the current generation of Quantum Computers is the occurrence of errors while performing computations. A proposed solution for their mitigation are quantum error correcting codes (QECC). This project focuses on the methodology of Lattice Surgery of Surface Codes which achieves fault tolerance by “surgically” merging and splitting a lattice of many quantum bits (qubits). In particular, we made a compiler that automatically translates quantum programs to instructions that specify Lattice Surgery merges and splits to add QECC.

**METHODS**

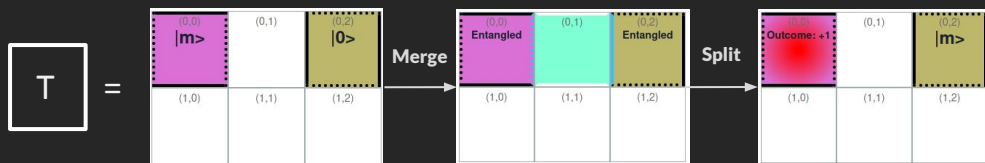
1. Accept a QASM circuit as input
2. High precision Clifford+T approx.
3. Generate Lattice Surgery Instructions
4. Routing on custom layouts
5. View the result in browser

**RESULTS**



# A state of the art, open source, user friendly, integrated way to add Error Correction to Quantum Programs

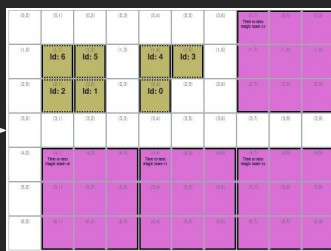
Lattice Surgery is a language for quantum error correction we compile and visualize:



Custom Layouts that can be hand or machine optimized for each computational task, as ASCII text

```

rrrrrrrr444
rQQRQQR444
rQQRQAR444
rrrrrrrrrrr
r111222333
r111222333
r111222333
    
```



We call LS Instructions an intermediate representation to separate the pre processing from the routing and scheduling

```

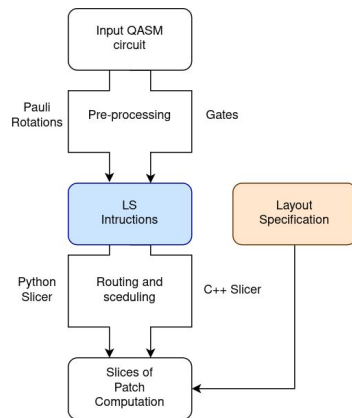
HGate 2
SGate 1
HGate 2
Init 4 |+>
RequestMagicState 9
MultiBodyMeasure 1:Z,4:Z
MeasureSinglePatch 4 Z
MultiBodyMeasure 2:X,4:X
SGate 2
Init 5 |+>
MultiBodyMeasure 1:Z,5:X
MultiBodyMeasure 2:X,5:X
MeasureSinglePatch 5 Z
    
```

The combination of the two formats above allows us to produce instructions that, with the addition of an **Error Decoder**, are ready to be sent to a physical quantum computer

Visit us at:  
[latticesurgery.com](http://latticesurgery.com)  
[github.com/latticesurgery-com](https://github.com/latticesurgery-com)  
(Give us a star!)

**Compiler Pipeline**

A pluggable pipeline in two major stages, optimized for execution speed and with a linear runtime option



**George Watkins**, [gwwatkin@sfu.ca](mailto:gwwatkin@sfu.ca)  
(Math and Computing Science)

**Alex Nguyen**, [alex.nguyen.5@sfu.ca](mailto:alex.nguyen.5@sfu.ca)  
(Computing Science)