Overview

Programming quantum computers will be bug-prone as programming classical computers.

Strong static type systems eliminate the possibility of several classes of bugs in classical computing. We need similar assurance of correctness in the quantum paradigm.

We propose Quantum Hoare Types that let programmers reason about and enforce certain semantic properties of their quantum programs.

Our approach has potential to be a unified system for programming, specifying, and reasoning about quantum programs.

Quantum Computing

Hadamard Gate

The Coin Flip of Quantum!

When the input is 0, it outputs 0 or 1 with a 50/50 chance.
When the input is 1, it also outputs 0 or 1 with a 50/50 chance.

Example: Quantum Teleportation

Alice is trying to send a message encoded in a qubit $q$ to Bob. There are three broad steps in the protocol:
1. Alice & Bob create entanglement between their two qubits.
2. Alice sends the message.
3. Bob reconstructs the message.

Quantum Hoare Types

A Hoare type [Nanevski et al. 2008] encodes preconditions and postconditions in the same spirit as Hoare triples.

$\{P\} \mathit{x} : A \{Q\}$

A classic example:

```
alloc : \forall o. \Pi x : o. \{\text{emp} \} y : \text{nat} \{ y \rightarrow x \}
```

To extend Hoare Type Theory to support quantum circuits, we augment its syntax with that of Qwire [Paykin et al. 2017] which is a core quantum circuit language that can cooperate with an arbitrary classical host language.

Qwire’s circuit language has a wire type $W$ that can be a unit, a bit, a qubit or a tuple of wires:

$$W := 1 \mid \text{bit} \mid qubit \mid W1 \otimes W2$$

The host language (HTT) is augmented with circuit types to support interaction between the two languages and to treat circuits as data:

$$A := \ldots \mid \text{Unit} \mid \text{Bool} \mid A \times A \mid \text{Circuit}(W1, W2)$$

Further Work

We look to explore how our types will evolve when we start incorporating higher-order constructs in our quantum language such as those demonstrated in languages like Quantum Lambda Calculus [Selinger and Valiron 2006] and Quipper [Green et al. 2013].

Another venue for exploration is to incorporate more precise types that can distinguish between qubits in pure classical state vs. those in superposition vs. those in entanglement.