

High level programming language for quantum computing

Davide Camino



UNIVERSITÀ
DI TORINO

davide.camino@edu.unito.it

Obiettivo del lavoro

Reasoning su quantum computer

- ① Base di conoscenza classica + query
- ② Embedding su quantum computer
- ③ Esecuzione e recupero risultati

Perché il quantum computing

Grazie a

- Entanglement
- Superposition

immagine

Violiamo

Strong Church-Turing Thesis

immagine

Due paradigmi di programmazione

Quantum Gate
immagine

Quantum Annealing
immagine

Quantum Gate

- Formalismo molto studiato
- Gate Reversibili
- Set di porte universali
- Algoritmi di Shore, ecc.

immagine

Quantum Annealer

immagine

- Ispirato a Simulated Annealing
- Effetto tunneling
- Adiabatic Quantum Computing (AQC)
- $\mathcal{H}(t) = s(t)\mathcal{H}_i + (1 - s(t))\mathcal{H}_f$

QA-Prolog

Il progetto

- Sviluppato da Scott Pakin
- Proof of concept
- Trasformazioni successive
- Prolog + Query $\rightarrow \mathcal{H}_f$
- Interagisce direttamente con il solver
- Raccoglie e organizza i risultati

immagine

Pipeline

immagine

Trasformazioni

- ① Prolog → Verilog (HDL)
- ② Verilog → Circuito digitale
- ③ Circuito digitale → \mathcal{H}_f simbolica
- ④ \mathcal{H}_f simbolica → \mathcal{H}_f fisica

QASM

\mathcal{H}_f simbolica $\rightarrow \mathcal{H}_f$ fisica

Cos'è

- Quantum macro assembler
- Sviluppato in Python
- Basso livello di astrazione
- Si interfaccia con Ocean

Cosa permette di fare

- Riferimento simbolico a *qubit*
- *Qubit* “pinnati” a TRUE o FALSE
- Incapsulare pattern in macro
- Creazione di librerie di macro
- Pulizia dell'output:
 - solo *qubit* “interessanti”
 - no slack variables

QASM

Esempio: Macro

```
# 2-input OR gate (Y = A or B)
!begin_macro or2
$A 0.3333
$B 0.3333
$Y -0.6667

$A $B 0.3333
$A $Y -0.6667
$B $Y -0.6667
!end_macro or2
```

Figure: or gate

```
# 1-input NOT gate (Y = not A)
!begin_macro not1
$A -0.5
$Y -0.5

$A $Y 1.0
!end_macro not1
```

Figure: not gate

```
# 2-input AND gate (Y = A and B)
!begin_macro or2
$A 0
$B 0
$Y 1

$A $B 0
$A $Y -0.7
$B $Y -0.7
!end_macro or2
```

Figure: and gate

Yosys - edif2qasm

Yosys - edif2qasm

Esempio

Quantum Computing
oooo

QA-Prolog
oooooooo●○

QAC to QAOA
ooooo

Ontologie
oooo

Esempio Completo
o

QA-Prolog

QA-Prolog

Esempio

Quantum Computing
oooo

QA-Prolog
ooooooooo

QAC to QAOA
●oooo

Ontologie
oooo

Esempio Completo
○

QAOA

Da \mathcal{H} a operatori di Paoli

Quantum Computing
oooo

QA-Prolog
oooooooo

QAC to QAOA
oo●oo

Ontologie
oooo

Esempio Completo
o

Esempio

Quantum Computing
oooo

QA-Prolog
oooooooo

QAC to QAOA
ooo●o

Ontologie
oooo

Esempio Completo
o

Esempio

Risultato 1

Quantum Computing
oooo

QA-Prolog
ooooooooo

QAC to QAOA
oooo●

Ontologie
oooo

Esempio Completo
o

Esempio

Risultato 2

Quantum Computing
oooo

QA-Prolog
ooooooooo

QAC to QAOA
ooooo

Ontologie
●ooo

Esempio Completo
○

Ontologie

Quantum Computing
oooo

QA-Prolog
oooooooo

QAC to QAOA
ooooo

Ontologie
○●○○

Esempio Completo
○

OWL

Quantum Computing
oooo

QA-Prolog
oooooooo

QAC to QAOA
ooooo

Ontologie
○○●○

Esempio Completo
○

Inferenze in OWL

Quantum Computing
oooo

QA-Prolog
ooooooooo

QAC to QAOA
ooooo

Ontologie
ooo●

Esempio Completo
○

Complessità dell'Inferenza

Da OWL-rdf a Prolog