Sample title

Dingchao Gao

Institute of Software Chinese Academy of Sciences

October 21, 2022

SIS,

problem [3]

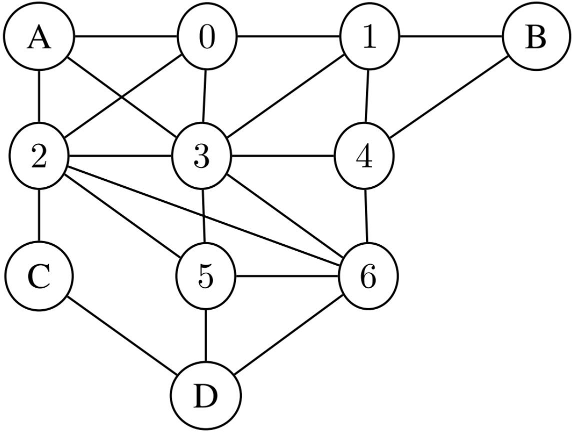


Figure: Zed city as an undirected graph

boolean function

python implementation of f

```
def f(v0, ..., v6 : BitVec(2)) ->
    BitVec(1):
  c0 = (v0 != ''00)
  c1 = (v1 != ''01) and (v1 != v0)
  c2 = (v2 != ''00) and (v2 != ''10)
       and (v2 != v0)
  c3 = (v3 != ''00) and (v3 != v0)
       and (v3 != v1) and (v3 != v2)
  c4 = (v4 != ''01) and (v4 != v1)
       and (v4 != v3)
  c5 = (v5 != ''11) and (v5 != v2)
       and (v5 != v3)
  c6 = (v6 != ''11) and (v6 != v2)
       and (v6 != v3) and (v6 != v4)
        and (v6 != v5)
  return c0 and c1 and c2 and c3
       and c4 and c5 and c6
```

hand-optimized python implementation of f

```
def f(v0, ..., v6 : BitVec(2)) ->
    BitVec(1):
  c1 = (v1[0] == v1[1]) and (v3 !=
       v1)
  c023 = ((v0 ^ v2 ^ v3) == ''00)
  c4 = (v4 != v1) and (v4 != v3)
  c5 = (v5 != v2) and (v5 != v3)
  c6 = ((v2 ^ v3 ^ v5 ^ v6) == ''00)
       and (v6 != v4)
  return c1 and c023 and c4 and c5
       and c6
```

flow

- Angel:prepare a uniform quantum state given as input a Boolean function
- Tweedledum:synthesizing, manipulating, and optimizing quantum circuits
- Caterpillar:automatically translate the combinational parts of a quantum algorithm into quantum gates

initial state[1]

- $$|\varphi_j\rangle = \frac{f}{\sqrt{|f|}} = \frac{1}{\sqrt{|f|}} \sum_{x \in \text{on}(J)} |x\rangle \tag{1}$$

- $$\mathrm{QSP}_f |0\rangle^{\otimes n} = \left(\mathrm{QSP}_{f_{\bar{x}_i}} \oplus \mathrm{QSP}_{f_{x_i}}\right) \left(G\left(p_f\left(\bar{x}_i\right)\right) \otimes I_{2^n-1}\right) |0\rangle \tag{2}$$

$$G(p)|0\rangle = \sqrt{p}|0\rangle + \sqrt{1-p}|1\rangle \tag{3}$$

$$G\left(p\left(x_i\right)\right) = R_y\left(2\cos^{-1}\left(\sqrt{p\left(x_i\right)}\right)\right) \tag{4}$$

[1] Fereshte Mozafari et al. "Automatic Uniform Quantum State Preparation Using Decision Diagrams". In: *2020 IEEE 50th International Symposium on Multiple-Valued Logic (ISMVL)* (2020). DOI: 10.1109/ismvl49045.2020.00-10.

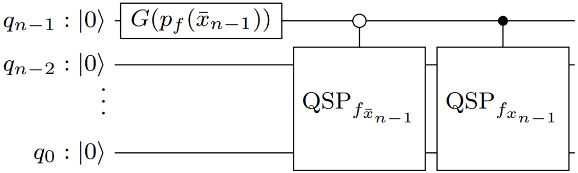


Figure: the general idea of QSP in the quantum circuit model for $i = n - 1$.

initial state example

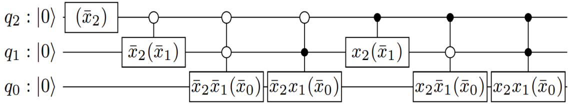
- for $f(x) = x_0x_1 \vee x_1x_2 \vee x_2x_0$
- 

Figure: the abstract quantum gates of $QSP_{<x_0x_1x_2>}$

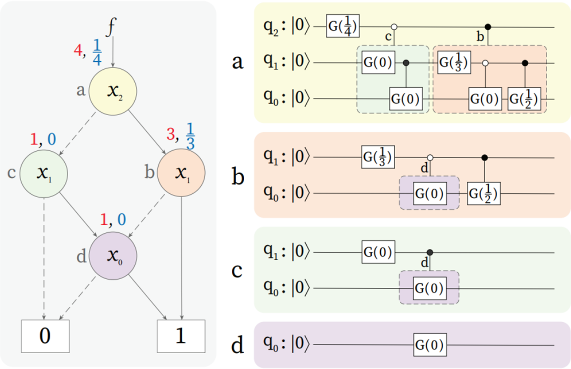


Figure: BDD for boolean function $< x_0 x_1 x_2 >$ and the procedure of extracting gates for eachnode from bottom to top

compiling oracle

- $$x_i = x_{j(i)} \oplus x_{k(i)} \quad \text{or} \quad x_i = x_{j(i)}^{p(i)} \wedge x_{k(i)}^{q(i)} \tag{5}$$

XAG[2]

- change a for b
- figure

[2] Giulia Meuli et al. "The Role of Multiplicative Complexity in Compiling Low T-count Oracle Circuits". In: (2019). DOI: 10.1109/iccad45719.2019.8942093.

XAG example

- for $f(x) = x_0x_1 \vee x_1x_2 \vee x_2x_0$
- we

$$x_4 = x_1 \oplus x_2, \qquad x_5 = x_2 \oplus x_3 \tag{6}$$

$$x_6 = x_4 \wedge x_5, \qquad x_7 = x_2 \oplus x_6 \tag{7}$$

XAG example

- for $f(x) = x_0x_1 \vee x_1x_2 \vee x_2x_0$
- we

$$x_4 = x_1 \oplus x_2, \qquad x_5 = x_2 \oplus x_3 \tag{8}$$

$$x_6 = x_4 \wedge x_5, \qquad x_7 = x_2 \oplus x_6 \tag{9}$$

XAG example

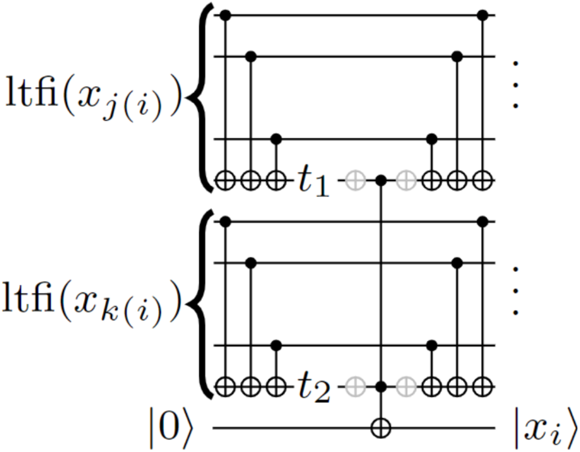


Figure: Quantum circuit construction for AND step in XAG

result

	Hand-optimized		Non-optimized	
	Qubits	cost	Qubits	cost
IBM's solution	32	5004		
Whit3z solution	32	2474		
XAG-based flow	31	2202	56	4347
XAG-based flow with pebbling	21	4497	30	7737

Table: quality of results for boolean function (hand-optimized and non-optimized), where $cost = q_1 + 10q_2$

induction

ABC

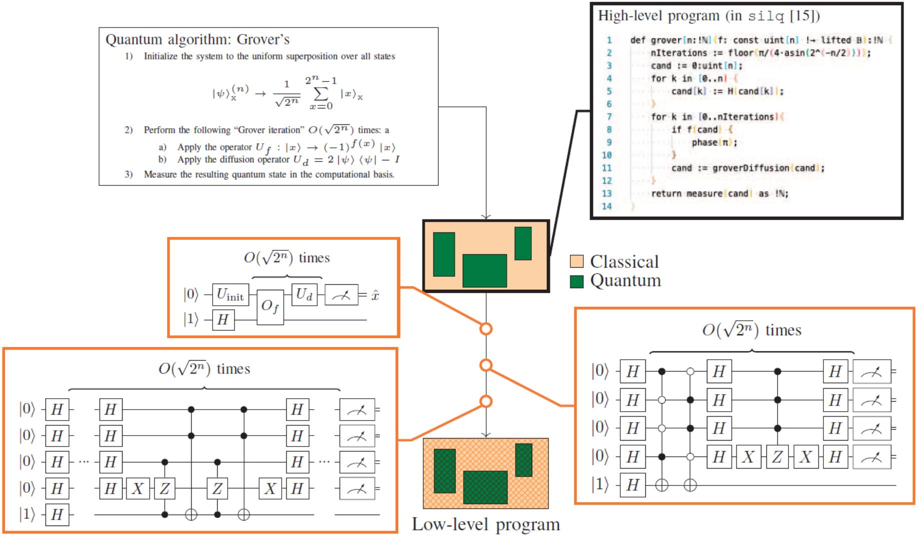


Figure: compilation flow overview

flexibility

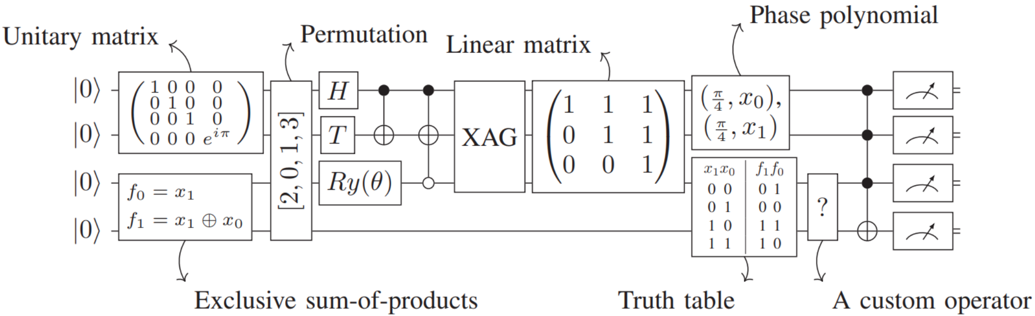


Figure: tweedledum's IR flexibility

synthesis

-

synthesis

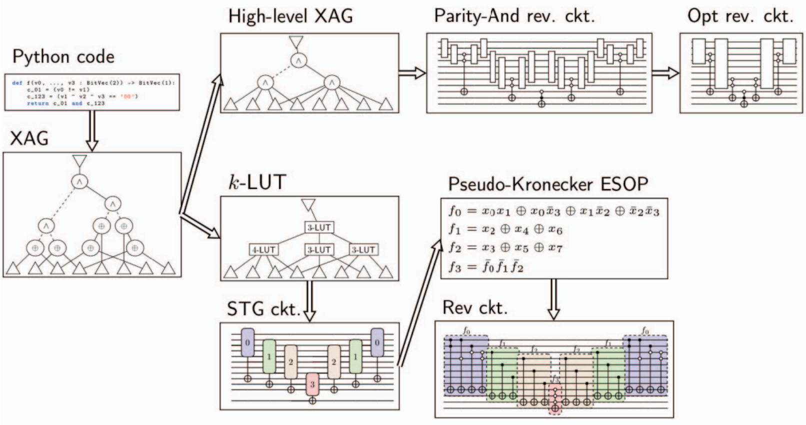


Figure: overview of possible Boolean function synthesis flows

compilation passes

-

references I

- Giulia Meuli et al. "The Role of Multiplicative Complexity in Compiling Low T-count Oracle Circuits". In: (2019). DOI: 10.1109/iccad45719.2019.8942093.
- Fereshte Mozafari et al. "Automatic Uniform Quantum State Preparation Using Decision Diagrams". In: *2020 IEEE 50th International Symposium on Multiple-Valued Logic (ISMVL)* (2020). DOI: 10.1109/ismvl49045.2020.00-10.
- Bruno Schmitt et al. "From Boolean functions to quantum circuits: A scalable quantum compilation flow in C++". In: *2021 Design, Automation Test in Europe Conference Exhibition (DATE)* (2021). DOI: 10.23919/date51398.2021.9474237.

END

Thank you