Algebraic Reasoning of Quantum Programs via Non-Idempotent Kleene Algebra

Yuxiang Peng, Mingsheng Ying, Xiaodi Wu
06/16/2022
Classical While-Program Equivalences

• A classical compiler rule: *loop unrolling.*

\[ \text{UNROLLING1} \equiv \]
\[ \text{while } q > 0 \text{ do} \]
\[ \phantom{\text{while }} P \]
\[ \text{done.} \]

\[ \text{UNROLLING2} \equiv \]
\[ \text{while } q > 0 \text{ do} \]
\[ \phantom{\text{while }} P; \]
\[ \phantom{\text{while }} \text{if } q > 0 \text{ then } P \]
\[ \text{done.} \]

• Equivalent classical programs.
Quantum While-Programs Equivalences

• What if quantum programs?

Unrolling 1 \equiv
\text{while } M[q] = 0 \text{ do } P; \\
\text{done.}

Unrolling 2 \equiv
\text{while } M[q] = 0 \text{ do } \\
P; \\
\text{if } M[q] = 0 \text{ then } P; \\
\text{done.}

Features:
• Measurements change states.
• Intrinsic non-deterministic nature.

They are equivalent if $M$ is projective. ($M_i M_j = \delta_{ij} M_i$)
KAT-like Algebraic Reasoning

• Kleene Algebra with Tests: “Regular expressions” $\iff$ programs:

$\text{Unrolling 1} \equiv$

while $M[q] = 0$ do
  $P$
  done.

$(m_0 p)^* m_1$

$\quad \iff$

$\text{Unrolling 2} \equiv$

while $M[q] = 0$ do
  $P$;
  if $M[q] = 0$ then $P$
  done.

$(m_0 p (m_0 p + m_1 \cdot 1))^* m_1$

• What are the axioms? Are they sound and complete?
Algebraic Reasoning via NKA

- Non-idempotent Kleene Algebra (NKA)
  \[ (m_0p(m_0p + m_1 \cdot 1))^*m_1 \]
  \[ = (m_0pm_0p + m_0pm_1)^*m_1 \]
  \[ = \ldots \]
  \[ = (m_0p)^*m_1 \]

- **Main theorem**: algebraic derivation induces equivalence.

**Theorem.** For quantum programs \( P, Q, \{S_i\}_{i=1}^k, \{T_i\}_{i=1}^k \), where \( \llbracket S_i \rrbracket = \llbracket T_i \rrbracket \) for all \( i \). If \( \vdash_{\text{NKA}} \left( \bigwedge_{i=1}^k \text{Enc}(S_i) = \text{Enc}(T_i) \right) \rightarrow \text{Enc}(P) = \text{Enc}(Q) \), then \( \llbracket P \rrbracket = \llbracket Q \rrbracket \). Here \( \text{Enc} \) is the encoding to algebraic expressions.
Structure of Concepts

- Quantum while-programs studied by Ying et al.
- Quantum path model embedded in sound
- NKA complete
- NKA with Tests
- Main theorem
- Normal form theorem
- Propositional quantum Hoare logic
- Compiler rules

Algebraic proofs
Non-idempotent Kleene Algebra

- NKA removes idempotency from KA.
  - Many rules of KA are still in NKA.

- Facts about NKA:
  - Sound and complete models
    - Rational power series over $\mathbb{N} = \mathbb{N} \cup \{\infty\}$ [Bloom & Ésik, 2009].
    - Weighted automata = RPS [Schützenberger, 1961].
  - Complexity
    - Deciding equation is PSPACE-complete.
    - Deciding inequality is undecidable [Eilenberg, 1974].

| Axioms of NKA |
|---------------|
| **Semiring Laws** | **Star Laws** |
| $p + (q + r) = (p + q) + r;$ | $1 + pp^* \leq p^*$; |
| $p + q = q + p;$ | $q + pr \leq r \rightarrow p^* q \leq r;$ |
| $p + 0 = p;$ | $q + rp \leq r \rightarrow qp^* \leq r;$ |
| $p(qr) = (pq)r;$ | 
| $1p = p1 = p;$ | 
| $0p = p0 = 0;$ | 
| $p(q + r) = pq + pr;$ | 
| $(p + q)r = pr + qr;$ | 

| **Partial Order Laws** |
|------------------------|
| $p \leq p;$ |
| $p \leq q \land q \leq p \rightarrow p = q;$ |
| $p \leq q \land q \leq r \rightarrow p \leq r;$ |
| $p \leq q \land r \leq s \rightarrow p + r \leq q + s;$ |
| $p \leq q \land r \leq s \rightarrow pr \leq qs;$ |

Derivable rules in NKA [Ésik & Kuich, 2004]

- (fixed-point)  (sliding)
  - $a^* = 1 + aa^*$  
  - $(ab)^* a = a(ba)^*$
- (positivity)  (unrolling)
  - $0 \leq a$  
  - \(a^* = (aa)^*(1 + a)\)
- (denesting)
  - \((a + b)^* = a^*(ba)^* = (a*b)^*a^*\)
Encoding Quantum While-Programs

• Encode as “regular expressions”.

\[
\begin{align*}
\text{Enc}(\text{skip}) &= 1; \\
\text{Enc}(q := |0\rangle) &= E(\langle q := |0\rangle); \\
\text{Enc}(\text{abort}) &= 0; \\
\text{Enc}(q := U[q]) &= E(\langle q := U[q]\rangle); \\
\text{Enc}(P_1; P_2) &= \text{Enc}(P_1) \cdot \text{Enc}(P_2); \\
\text{Enc}(\text{case} M[q] \rightarrow P_i \text{ end}) &= \sum_i E(M_i) \cdot \text{Enc}(P_i); \\
\text{Enc}(\text{while} M[q] = 1 \text{ do } P \text{ end}) &= (E(M_1) \cdot \text{Enc}(P))^* E(M_0)
\end{align*}
\]

\[E: \text{elementary operations } \Rightarrow \text{ symbols}\]

• Kleene star: \(\mathcal{E}^* = \mathcal{E}^0 + \mathcal{E}^1 + \mathcal{E}^2 + \cdots\)
  • “*” is partially defined for quantum channels.
  • \(\mathcal{E}_I^* = \mathcal{E}_I + \mathcal{E}_I + \mathcal{E}_I + \cdots\): divergent sum
• Aim for a total Kleene star function.
Quantum Path Model

- Quantum processes take *sum of all paths*.
  - $\mathcal{M}_0\left(\sum_n|0\rangle\langle 0|\right) = \sum_n|0\rangle\langle 0|$, $\mathcal{M}_0\left(\sum_n|1\rangle\langle 1|\right) = 0$.
  - Need to distinguish *different infinities*.

- Quantum path model
  - $\mathcal{P}\mathcal{O}_\infty$: generalization of *quantum states*
    - Equivalence classes of quantum state multisets.
    - Embeds quantum states.
  - $\mathcal{P}$: generalization of *quantum channels*
    - *Linear* and *monotone* transformations of $\mathcal{P}\mathcal{O}_\infty$.
    - Embeds quantum channels.
Quantum Interpretation

• QI interprets expressions into QPM.
  • \text{int} = (\mathcal{H}, \text{eval}).
  • \text{eval}: \text{symbols} \Rightarrow \text{quantum channels}.

\begin{align*}
Q_{\text{int}}(0) &= O_{\mathcal{H}}, & Q_{\text{int}}(e + f) &= Q_{\text{int}}(e) + Q_{\text{int}}(f), \\
Q_{\text{int}}(1) &= I_{\mathcal{H}}, & Q_{\text{int}}(e \cdot f) &= Q_{\text{int}}(e) \cdot Q_{\text{int}}(f), \\
Q_{\text{int}}(a) &= \langle\text{eval}(a)\rangle^\dagger, & Q_{\text{int}}(e^*) &= Q_{\text{int}}(e^*)^*.
\end{align*}

• Axioms of NKA are \textbf{sound} and \textbf{complete} w.r.t. quantum interpretation.

\textbf{Theorem.} For expressions \(e, f\) over a finite alphabet, there is
\[ \vdash_{\text{NKA}} e = f \iff \forall \text{int}: Q_{\text{int}}(e) = Q_{\text{int}}(f) \]

\textbf{Insight:} NKA captures all equations for quantum.

• Soundness leads to the \textit{main theorem}.

• QI inverts encoding:
  • \(Q_{\text{int}}(\text{Enc}(P)) = \langle\lceil P \rceil\rangle^\dagger\).

\textbf{algebra expression} \hspace{1cm} \textbf{interpretation} \hspace{1cm} \textbf{encoding} \hspace{1cm} \textbf{program semantics}
Verifying Compiler Rule

• Revisit loop unrolling

\( \vdash_{\text{NKA}} m_1 m_1 = m_1 \land m_1 m_0 = 0 \rightarrow (m_0 p)^* m_1 = (m_0 p (m_0 p + m_1 \cdot 1))^* m_1. \)

• Main theorem \( \implies \) if \( M_i \circ M_j = \delta_{ij} M_i. \)

• More examples in the paper
  • Quantum specific rule: loop boundary cancellation
  • Real world application: quantum signal processing

Derivable equations in NKA:

\begin{align*}
(a + b)^* &= a^* (ba^*)^* \\
a^* &= 1 + aa^* \\
a^* &= (aa)^* (1 + a)
\end{align*}
Quantum Böhm–Jacopini Theorem

• A normal form theorem:

**Theorem.** For quantum program $P$, there is a quantum program with one **while loop** that is equivalent to $P; p_c := |0\rangle$. Here $C$ is an auxiliary classical space.

• Observed in [Yu, 2019]. We give an algebraic proof to it.

• Idea:
  • Reconstruct control flows.
  • Prove equivalences via NKA.
NKA with Tests (NKAT)

• Classical tests serve two functionalities:
  • Property test and branch guard.
• Quantum: separate concepts.

• NKA with Tests
  • Quantum predicates: an effect algebra.
    • EA $(\mathcal{L}, \oplus, 0, e)$: 5 axioms.
    • EA is embedded in NKA.
  • Quantum measurements: partitions $(m_i)_{i \in I}$.
    • $m_i \mathcal{L} \subseteq \mathcal{L}$ and $\sum_{i \in I} m_i e = e$. 

|                      | Property test | Branch guard |
|----------------------|---------------|--------------|
| Classical test       | ✔️            | ✔️           |
| Quantum predicate    | ✔️            | ❌           |
| Quantum measurement  | ❌            | ✔️           |
Propositional Quantum Hoare Logic

- NKAT encodes quantum Hoare triples:
  \[ \models_{par} \{A\}P\{B\} \iff \text{Enc}(P)\overline{b} \leq \overline{a} \]

- Propositional QHL (a fragment of QHL [Ying, 2011])

  \[
  \begin{align*}
  (\text{Ax.Sk}) & : \{A\} \text{ skip } \{A\} \\
  (\text{Ax.Ab}) & : \{H\} \text{ abort } \{O_H\} \\
  (\text{R.OR}) & : \frac{A \sqsubseteq A' \{A'\}P\{B'\} \quad B' \sqsubseteq B}{\{A\}P\{B\}} \\
  (\text{R.IF}) & : \frac{\sum_i M_i^o(A_i)\text{ case } M \xrightarrow{p_i} \{P_i\} \text{ end}\{B\}}{\{A\}_{P_1; P_2}\{C\}} \\
  (\text{R.SC}) & : \frac{\{A\}_{P_1}\{B\}_{P_2}\{C\}}{\{A\}_{P_1; P_2}\{C\}} \\
  (\text{R.LP}) & : \frac{A \sqsubseteq A' \{A'\}P\{B\} \quad B' \sqsubseteq B}{\{A\}P\{B\}} \\
  \end{align*}
  \]

  \[
  \begin{align*}
  (\text{Ax.Sk}) & : 1\overline{a} \leq \overline{a}, \\
  (\text{Ax.Ab}) & : 0\overline{0} \leq \overline{1}, \\
  (\text{R.OR}) & : a \leq a' \land a'b' \leq \overline{a'} \land b' \leq b \Rightarrow p\overline{b} \leq \overline{a}, \\
  (\text{R.IF}) & : \left( \bigwedge_{i \in L} p_i \overline{b} \leq \overline{a_i} \right) \Rightarrow (\sum_{i \in L} m_i p_i) \overline{b} \leq \sum_i m_i a_i, \\
  (\text{R.SC}) & : p_1 \overline{b} \leq \overline{a} \land p_2 \overline{c} \leq \overline{b} \Rightarrow p_1 p_2 \overline{c} \leq \overline{a}, \\
  (\text{R.LP}) & : pm_0 a + m_1 b \leq \overline{b} \Rightarrow (m_1 p)^* m_0 \overline{a} \leq m_0 a + m_1 b.
  \end{align*}
  \]

- Algebraic reasoning is easier than matrix analysis.
Future Directions

• Applications
  • Quantum NetKAT for quantum software-defined networks?
  • Finer characterizations of quantum measurements?

• Automation
  • Bisimulation and co-algebra for NKA?
    • Faster equivalence checking of NKA equations.
    • Algorithms deciding Horn formulae.
  • Formal systems in Coq?
Q&A

Thanks!
References

• [Ésik&Kuich, 2004] Zoltán Ésik and Werner Kuich. 2004. Inductive *-semirings. *Theoretical Computer Science* 324, 1 (2004), 3–33.

• [Bloom&Ésik, 2009] Stephen L Bloom and Zoltán Ésik. 2009. Axiomatizing rational power series over natural numbers. *Information and Computation* 207, 7 (2009), 793–811.

• [Schützenberger, 1961] M.P. Schützenberger. 1961. On the definition of a family of automata. *Information and Control* 4, 2 (1961), 245 – 270.

• [Eilenberg, 1974] Samuel Eilenberg. 1974. *Automata, languages, and machines*. Academic press.

• [Yu, 2019] Nengkun Yu. 2019. Quantum Temporal Logic. *arXiv e-prints*, Article arXiv:1908.00158 (July 2019).

• [Ying, 2011] Mingsheng Ying. 2011. Floyd–Hoare Logic for Quantum Programs. *ACM Transactions on Programming Languages and Systems* 33, 6 (2011).