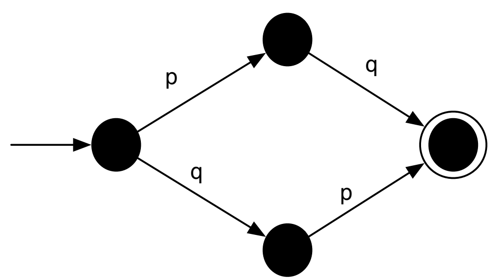
KAT-ML
An Interactive Theorem Prover for Kleene Algebra w/Tests

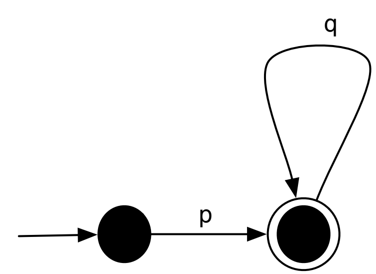
Kamal Aboul-Hosn
Dexter Kozen

Kleene Algebra (KA)

- The algebra of regular expressions



`pq + qp`



`pq*`

Axioms of KA [K91]

- K is an idempotent semiring under +,•, 0, 1

$$
\begin{aligned}
(p+q)+r &= p+(q+r) && (1) & (pq)r &= p(qr) && (2)\\
p+q &= q+p && (3) & p1 &= 1p = p && (4)\\
p+0 &= p+p = p && (5) & 0p &= p0 = 0 && (6)\\
p(q+r) &= pq+pr && (7) & (p+q)r &= pr+qr && (8)\\
1+pp^* &\leq p^* && (9) & q+pr \leq r &\to p^*q \leq r && (10)\\
1+p^*p &\leq p^* && (11) & q+rp \leq r &\to qp^* \leq r && (12)
\end{aligned}
$$

- $p^*q = \text{ least } x \text{ such that } q+px \leq x$
- $qp^* = \text{ least } x \text{ such that } q+xp \leq x$

$$x \leq y \stackrel{def}{\Longleftrightarrow} x+y=y$$

Standard Interpretation

Regular sets over Σ

$A+B = A \cup B$

$AB = \{xy \mid x \in A, y \in B\}$

$A^* = \bigcup_{n \geq 0} A^n = A^0 \cup A^1 \cup A^2 \cup \ldots$

$1 = \{\varepsilon\}$

$0 = \varnothing$

$p \in \Sigma$ interpreted as $\{p\}$

Useful Properties

$$
\begin{array}{rcll}
1 + pp^* & = & 1 + p^*p = p^* & \\
p^*p^* & = & p^{**} = p^* & \\
(pq)^*p & = & p(qp)^* & \text{sliding} \\
(p^*q)^*p^* & = & (p+q)^* & \text{denesting} \\
px = xp & \Rightarrow & p^*x = xq^* & \text{bisimulation} \\
qp = 0 & \Rightarrow & (p+q)^* = p^*q^* &
\end{array}
$$

Applications of KA

- Automata and regular expressions
- Relational algebra
- Design and analysis of algorithms
    - shortest paths
    - connectivity
    - computational geometry

Kleene Algebra w/Tests (KAT)

$$(K, B, +, \cdot, ^*, ^-, 0, 1)$$

- $(K, +, \cdot, ^*, 0, 1)$ is a Kleene algebra
- $(B, +, \cdot, ^-, 0, 1)$ is a Boolean algebra

  $B \subseteq K$
- p,q,r range over K
- A,B,C range over B

KAT-ML

- Written in Standard ML (with SML Tk)
- Works on unix-based OSes, Windows
- Goals:
  - Interactively develop proofs in KAT
  - Manage proofs and theorems in a reusable fashion
  - Formally verify proof already in the literature

Fundamental Commands

- **Publish**: Create a new theorem (without a proof!)
- **Cite**: Incorporates previous proofs into current proof

Representing Proofs

- A term abstracted over
  - Term variables `p,q,r`... and boolean variables `A,B,C`.... in the theorem
  - Proof variables P_0, P_1,..., representing proofs of premises
  - Task variables T_0, T_1,... for incomplete tasks

Representing Proofs

$$\forall x_1 \ldots \forall x_m \;\; \phi_1 \to \phi_2 \to \cdots \to \phi_n \to \psi$$

Proof term is well-typed

$$\lambda x_1 \ldots \lambda x_m . \lambda P_1 \ldots \lambda P_n . (T P_1 \cdots P_n)$$

By Curry-Howard isomorphism, the type of term is the proof

A Sample Proof

- All of the following are equivalent

$$\begin{aligned} Up &= UpV \\ Up\overline{V} &= 0 \\ Up &\leq pV \end{aligned}$$

Verified Proofs

- Hoare While rule:

$$\frac{\{B;C\}p\{C\}}{\{C\}\text{while } B \text{ do } p\{C;\overline{B}\}}$$

- Need to show:

$$B;C;p = B;C;p;C \to C;(B;p)^*;\overline{B} = C;(B;p)^*;\overline{B};C;\overline{B}$$

Modeling Programs

[Fischer + Ladner 74]

$$\begin{aligned}
x := e \quad &\equiv \quad a \\
e < f \quad &\equiv \quad A \\
\text{if } B \text{ then } p \text{ else } q \quad &\equiv \quad Bp + \overline{B}q \\
\text{while } B \text{ do } p \quad &\equiv \quad (Bp)^*\overline{B}
\end{aligned}$$

Schematic KAT

$$
\begin{array}{rcll}
x := s; y := t & = & y := t[x/s]; x := s & y \notin FV(s) \\
x := s; y := t & = & x := s; y := t[x/s] & x \notin FV(s) \\
x := s; x := t & = & x := t[x/s] & \\
\varphi[x/t]; x := t & = & x := t; \varphi & \\
x := x & = & 1 &
\end{array}
$$

Future Additions

- First-order constructs
  - Add interpreted level
- “Adaptive” heuristics
- Readable printing of proofs
- Online database of proofs

Download

http://www5.cs.cornell.edu/~kamal/kat