Extended Regular Expressions

**Definition**

- \( r, s, t := \epsilon | A | r + s | \cdot r s | r^* | r \& s | \neg r \)

- \( \Sigma \) is a potentially infinite set of symbols
- \( A, B, C \subseteq \Sigma \) range over sets of symbols
- \( [r] \subseteq \Sigma^* \) is the language of a regular expression \( r \), where \([A] = A\)

Language Inclusion

**Definition**

Given two regular expressions \( r \) and \( s \),

\[ r \subseteq s \iff [r] \subseteq [s] \]

- Decidable using standard techniques:
  - Construct DFA for \( r \& s \) and check for emptiness
- Drawback is the expensive construction of the automaton
- \( \text{PSPACE-complete} \)
Antimirov’s Algorithm

- Deciding containment for basic regular expressions
- Based on derivatives and expression rewriting
- Avoid the construction of an automaton
- \( \partial_a(r) \) computes a regular expression for \( a^{-1}[r] \) (Brzozowski) with \( a \in \Sigma \) and \( r \in \text{Re} \)

**Lemma**

For regular expressions \( r \) and \( s \),

\[
\forall u \in \Sigma^* \quad \partial_u(r) \subseteq \partial_u(s).
\]

First Symbols

**Lemma**

\[
\forall u \in \Sigma^* \quad (\nu(r) \Rightarrow \nu(s)) \land (\forall a \in \text{first}(r) \partial_a(r) \subseteq \partial_a(s))
\]

- Let \( \text{first}(r) := \{ a \mid aw \in [r] \} \) be the set of first symbols
- Restrict symbols to first symbols of the left hand side
- CC-Unfold does not have to consider the entire alphabet
- For extended regular expressions, \( \text{first}(r) \) may still be an infinite set of symbols

Choice of next step’s inequality is nondeterministic
An infinite alphabet requires to compute for infinitely many \( a \in \Sigma \)
Problems

- Antimirov’s algorithm only works with basic regular expressions or requires a finite alphabet.
- Extension of partial derivatives [Caron et al.] that computes an NFA from an extended regular expression.
- Works on sets of sets of expressions.
- Computing derivatives becomes more expensive.

Goal

- Algorithm for deciding \( J \subseteq K \) quickly.
- Handle extended regular expressions.
- Deal effectively with very large (or infinite) alphabets (e.g., Unicode character set).

Solution

- Require finitely many atoms, even if the alphabet is infinite.
- Compute derivatives with respect to literals.

Representing Sets of Symbols

A literal is a set of symbols \( A \subseteq \Sigma \).

Definition

\( A \) is an element of an effective boolean algebra \( (U, \cup, \cap, \neg, \top, \bot) \) where \( U \subseteq \wp(\Sigma) \) is closed under the boolean operations.

- For finite (small) alphabets:
  \( U = \wp(\Sigma) \), \( A \subseteq \Sigma \).
- For infinite (or just too large) alphabets:
  \( U = \{ A \in \wp(\Sigma) \mid A \text{ finite or } \Sigma \text{ finite} \} \).
- Second-level regular expressions:
  \( \Sigma \subseteq \wp(\Sigma^*) \) with \( U = \{ A \in \wp(\Sigma^*) \mid A \text{ is regular} \} \).
- Formulas drawn from a first-order theory over alphabets.
  For example, \( [a-z] \) represented by \( x \geq 'a' \land x \leq 'z' \).
Derivatives with respect to Literals

- Definition for $\partial_a(r)$?
- $\partial_a(r)$ computes a regular expression for $a^{-1}[r]$ (Brzozowski)

Desired property

$$\forall a \in A \quad [\partial_a(r)] \cup A^{-1}[r] = \bigcup_{b \in A} a^{-1}[r] \cup \bigcup_{c \in A} \partial_b(r)$$

Positive Derivatives on Literals

Definition

$$\delta^+_A(B) := \begin{cases} \epsilon, & B \cap A \neq \perp \\ \emptyset, & \text{otherwise} \end{cases}$$

Problem

With $A = \{ a, b \}$ and $r = (a \cdot c) \cup (b \cdot c)$,

$$\delta^+_A(r) = \delta^+_A(a \cdot c) \cup \delta^+_A(b \cdot c) = c \cdot c \cup \emptyset = c$$

Negative Derivatives on Literals

Definition

$$\delta^-_A(B) := \begin{cases} \epsilon, & B \cap A = \perp \\ \emptyset, & \text{otherwise} \end{cases}$$

Problem

With $A = \{ a, b \}$ and $r = (a \cdot c) \cup (b \cdot c)$,

$$\delta^-_A(r) = \delta^-_A(a \cdot c) \cup \delta^-_A(b \cdot c) = \emptyset \cup \emptyset = \emptyset$$
Positive and Negative Derivatives

- Extends Brzozowski’s derivative operator to sets of symbols.
- Defined by induction and flip on the complement operator

Definition

From $\partial_a(s) = \partial_a(s)$, define:

$\delta^+_A(r) := \bigcup_{a \in A} \partial_a(r)$

$\delta^-_A(r) := \bigcap_{a \in A} \partial_a(r)$

Lemma

For any regular expression $r$ and literal $A$,

$[\delta^+_A(r)] \supseteq \bigcup_{a \in A} [\partial_a(r)]$

$[\delta^-_A(r)] \subseteq \bigcap_{a \in A} [\partial_a(r)]$

Matthias Keil, Peter Thiemann
Regular Expression Inequalities
December 15, 2014 13 / 21

Literals of an Inequality

Lemma

$r \subseteq s \iff (\nu(r) \Rightarrow \nu(s)) \land (\forall a \in \text{first}(r)) \partial_a(r) \subseteq \partial_a(s)$

- first($r$) may still be an infinite set of symbols
- Use first literals as representatives of the first symbols

Example

1. Let $r = \{a, b, c, d\} \cdot d^*$, then $\{a, b, c, d\}$ is a first literal
2. Let $s = \{a, b, c\} \cdot c^* + \{b, c, d\} \cdot d^*$, then $\{a, b, c\}$ and $\{b, c, d\}$ are first literals

Matthias Keil, Peter Thiemann
Regular Expression Inequalities
December 15, 2014 14 / 21

Literals of an Inequality (cont’d)

Problem

Let $r = \{a, b, c\} \cdot d^*$, $s = \{a, b, c\} \cdot c^* + \{b, c, d\} \cdot d^*$, and $A = \{a, b, c, d\}$, then

$\delta^+_A(r) \subseteq \delta^+_A(s)$ (1)

$\delta^-_A((a, b, c) \cdot d^*) \subseteq \delta^-_A((a, b, c) \cdot c^*) + \delta^-_A((b, c, d) \cdot d^*)$ (2)

$\delta^+_A(d^*) \subseteq c^* + d^*$ (3)

- Positive (negative) derivatives yield an upper (lower) approximation
- To obtain the precise information, we need to restrict these literals suitably to next literals, e.g. $\{(a), (b, c), (d)\}$

Matthias Keil, Peter Thiemann
Regular Expression Inequalities
December 15, 2014 15 / 21
Next Literals

next(\epsilon) = \{\emptyset\}
next(A) = \{A\}
next(r+s) = \text{next}(r) \times \text{next}(s)
next(r.s) = \text{next}(r) \times \text{next}(s), \nu(r)
next(r^*) = \text{next}(r)
next(r&s) = \text{next}(r) \cap \text{next}(s)
next(r) = \text{next}(r) \cup \left[\{A\} | A \in \text{next}(r)\}\right]

Next Literals (cont’d)

Example
Let \( s = \{a, b, c\} \cdot c^* + \{b, c, d\} \cdot d^* \); then
\[
\text{next}(s) = \text{next}\{(a, b, c) \cdot c^*\} \times \text{next}\{(b, c, d) \cdot d^*\}
= \{(a, b, c) \times (b, c, d)\} = \{(a), (b, c), (d)\}
\]

Lemma
For all \( r \),
- \( \bigcup \text{next}(r) \supset \text{first}(r)\)
- \( \text{next}(r) \) is finite
- \( (\forall A, B \in \text{next}(r)) A \cap B = \emptyset \)

Coverage

Lemma
Let \( \Sigma = \text{next}(r) \) and \( A \in \text{next}(r) \setminus \{\emptyset\} \).
- \( (\forall a \in A) \partial_A(r) = \partial_A(r) \land \delta_A^r(r) = \delta_A^r(r) = \partial_A(r)\)
- \( (\forall a \notin \Sigma) \partial_A(r) = \emptyset \)

Definition
Let \( A' \in \text{next}(r) \). For each \( 0 \neq A' \subseteq A' \) define \( \partial_A(r) = \partial_A(r) \), where \( a \in A \).
Next Literals of an Inequality

- Next literal of $\text{next}(r \subseteq s)$
- Sounds to join literals of both sides $\text{next}(r) \times \text{next}(s)$
- Contains also symbols from $s$
- First symbols of $r$ are sufficient to prove containment

**Definition**

Let $\mathcal{L}_1$ and $\mathcal{L}_2$ be two sets of disjoint literals:

$$\mathcal{L}_1 \triangleleft \mathcal{L}_2 := \{ (A_1 \cap A_2), (A_1 \cap \mathcal{L}_2) \mid A_1 \in \mathcal{L}_1, A_2 \in \mathcal{L}_2 \}$$

Left-based join corresponds to $\text{next}(r) \triangleleft \text{next}(s)$.

**Definition**

Let $r \triangleleft s$ be an inequality, define:

$$\text{next}(r \triangleleft s) := \text{next}(r) \triangleleft \text{next}(s)$$

**Solving Inequalities**

**Lemma**

$$r \subseteq s \iff (\nu(r) \Rightarrow \nu(s)) \land (\forall a \in \text{first}(r)) \partial_a(r) \subseteq \partial_a(s)$$

To determine a finite set of representatives:

- select one symbol $a$ from each equivalence class $A \in \text{next}(r)$
- calculate with $\delta_a^r(r)$ or $\delta_a^s(r)$ with $A \in \text{next}(r)$

**Theorem (Containment)**

$$r \subseteq s \iff (\nu(r) \Rightarrow \nu(s)) \land (\forall A \in \text{next}(r \triangleleft s)) \partial_A(r) \subseteq \partial_A(s)$$

**Conclusion**

- Generalize Brzozowski’s derivative operator
- Extend Antimirov’s algorithm for proving containment
- Provides a symbolic decision procedure that works with extended regular expressions on infinite alphabets
- Literals drawn from an effective boolean algebra
- Main contribution is to identify a finite set that covers all possibilities
The language \([r] \subseteq \Sigma^*\) of a regular expression \(r\) is defined inductively by:

\[
\begin{align*}
[r] & = \{\epsilon\} \\
[A] & = \{a \mid a \in A\} \\
[r + s] & = [r] \cup [s] \\
[r \cdot s] & = [r] \cdot [s] \\
[r^*] & = [r] \cup [r^*] \\
[r \& s] & = [r] \cap [s] \\
\neg r & = \neg [r]
\end{align*}
\]

The nullable predicate \(\nu(r)\) indicates whether \([r]\) contains the empty word, that is, \(\nu(r)\) iff \(\epsilon \in [r]\).

\[
\begin{align*}
\nu(\epsilon) & = \text{true} \\
\nu(A) & = \text{false} \\
\nu(r + s) & = \nu(r) \lor \nu(s) \\
\nu(r \cdot s) & = \nu(r) \land \nu(s) \\
\nu(r^*) & = \text{true} \\
\nu(r \& s) & = \nu(r) \land \nu(s) \\
\nu(\neg r) & = \neg \nu(r)
\end{align*}
\]

\(\partial_a(r)\) computes a regular expression for the left quotient \(a^{-1}[r]\).

\[
\begin{align*}
\partial_a(\epsilon) & = \emptyset \\
\partial_a(A) & = \{a \in A\} \\
\partial_a(r + s) & = \partial_a(r) + \partial_a(s) \\
\partial_a(r \cdot s) & = \partial_a(r) \cdot \partial_a(s), \quad \nu(r) \\
\partial_a(r^*) & = \partial_a(r)^* \\
\partial_a(\neg r) & = \neg \partial_a(r) \\
\partial_a(r \& s) & = \partial_a(r) \cdot \partial_a(s) \\
\partial_a(\neg r) & = \neg \partial_a(r)
\end{align*}
\]
First Symbols

Let \( \text{first}(r) := \{ a \mid aw \in \mathcal{L}(r) \} \) be the set of first symbols derivable from regular expression \( r \).

\[
\begin{align*}
\text{first}(\epsilon) &= \emptyset \\
\text{first}(A) &= \{ A \} \\
\text{first}(r+s) &= \text{first}(r) \cup \text{first}(s) \\
\text{first}(rs) &= \{ \text{first}(r) \cup \text{first}(s) \} \\
\text{first}(r^*) &= \text{first}(r) \\
\text{first}(r\&s) &= \text{first}(r) \cap \text{first}(s) \\
\text{first}(r\mid) &= \{ a \in \text{first}(r) \mid \partial_a(r) \neq \Sigma^* \}
\end{align*}
\]

First Literals

Let \( \text{lLiteral}(r) := \{ a \mid aw \in \mathcal{L}(r) \} \) be the set of first symbols derivable from regular expression \( r \).

\[
\begin{align*}
\text{lLiteral}(\epsilon) &= \emptyset \\
\text{lLiteral}(A) &= \{ A \} \\
\text{lLiteral}(r+s) &= \text{lLiteral}(r) \cup \text{lLiteral}(s) \\
\text{lLiteral}(rs) &= \{ \text{lLiteral}(r) \cup \text{lLiteral}(s) \} \\
\text{lLiteral}(r^*) &= \text{lLiteral}(r) \\
\text{lLiteral}(r\&s) &= \text{lLiteral}(r) \cap \text{lLiteral}(s) \\
\text{lLiteral}(r\mid) &= \{ a \in \text{lLiteral}(r) \mid \partial_a(r) = \Sigma \}
\end{align*}
\]

Coverage

Lemma (Coverage)

For all \( a, u, \) and \( r \) it holds that:

\[
u \in [\partial_u(r)] \iff \exists A \in \text{next}(r): a \in A \land u \in [\delta_u^A(r)] \land u \in [\delta_u^r(r)]
\]
Theorem (Finiteness)
Let $R$ be a finite set of regular inequalities. Define
$$F(R) = R \cup \{ \partial A(r \subseteq s) \mid r \subseteq s \in R; A \in \text{next}(r \subseteq s) \}$$
For each $r$ and $s$, the set $\bigcup_{i \in \mathbb{N}} F(i(r \subseteq s))$ is finite.

Decision Procedure for Containment

\[ \begin{array}{ll}
\text{(Disprove)} & \nu(s) \\
\Gamma \vdash \neg \nu(s) & r \subseteq s : \text{false} \\
\text{(Cycle)} & r \subseteq s \in \Gamma \\
\Gamma \vdash r \subseteq s & r \subseteq s : \text{true} \\
\text{(Unfold-True)} & r \subseteq s \notin \Gamma \\
\forall A \in \text{next}(r \subseteq s) : \Gamma \cup \{ r \subseteq s \} \vdash \partial A(r \subseteq s) & r \subseteq s : \text{true} \\
\text{(Unfold-False)} & r \subseteq s \notin \Gamma \\
\exists A \in \text{next}(r \subseteq s) : \Gamma \cup \{ r \subseteq s \} \vdash \partial A(r \subseteq s) & r \subseteq s : \text{false} \\
\end{array} \]

Prove and Disprove Axioms

\[ \begin{array}{ll}
\text{(Prove-Identity)} & \Gamma \vdash r \subseteq r : \text{true} \\
\text{(Prove-Empty)} & \Gamma \vdash \emptyset \subseteq s : \text{true} \\
\text{(Prove-Nullable)} & \nu(s) \\
\Gamma \vdash \varepsilon \subseteq s & \varepsilon \subseteq s : \text{true} \\
\text{(Disprove-Empty)} & \exists A \in \text{next}(r) : A \neq \emptyset \\
\Gamma \vdash r \subseteq \emptyset & r \subseteq \emptyset : \text{false} \\
\end{array} \]
Soundness

**Theorem (Soundness)**

For all regular expression \( r \) and \( s \):

\[
\emptyset \vdash r \trianglelefteq s : \top \iff r \trianglelefteq s
\]

Matthias Keil, Peter Thiemann
Regular Expression Inequalities
December 15, 2014 10 / 13

Negative Derivatives

**Counterexample**

Let \( r = \{a, b, c, d\}d^* \), \( s = \{a, b, c\}d^* + \{b, c, d\}d^* \), and \( A = \{a, b, c, d\} \), then

\[
\delta^-(r) \subseteq \delta^+(s) \\
\delta^-(\{a, b, c, d\}d^*) \subseteq \delta^+(\{a, b, c\}d^* + \{b, c, d\}d^*)
\]

\[
\emptyset \trianglelefteq \emptyset \ (6)
\]

Matthias Keil, Peter Thiemann
Regular Expression Inequalities
December 15, 2014 12 / 13

Next Literals of an Inequality

**Example**

Let \( r = \{a, b, c, d\}d^* \), \( s = \{a, b, c\}c^* + \{b, c, d\}d^* \) then

\[
\text{next}(r \sqsubseteq s) = \text{next}(\{a, b, c, d\}d^*) = \text{next}(\{a, b, c\}d^* + \{b, c, d\}d^*) = \{\{a\}, \{b, c\}, \{d\}\}
\]

Matthias Keil, Peter Thiemann
Regular Expression Inequalities
December 15, 2014 13 / 13
Incomplete Containment

Conjecture

\[ r \subseteq s \iff (\nu(r) \Rightarrow \nu(s)) \land ((\forall A \in \text{Mora}(r)) \text{delta}^*_A(r) \subseteq \text{delta}^*_A(s)) \]