Insertion Operations on Deterministic Reversal-Bounded Counter Machines

Joey Eremondi,
Department of Computing Science, Utrecht University
j.s.eremondi@students.uu.nl

Oscar Ibarra,
Department of Computer Science, University of California, Santa Barbara
ibarra@cs.ucsb.edu

Ian McQuillan,
Department of Computer Science, University of Saskatchewan
Canada,
mcquillan@cs.usask.ca

Background: Deterministic Reversal-bounded Counter Machines

DCMs

- Deterministic Finite Automaton with n unary counters
- Each counter can switch from incrementing to decrementing or vice versa at most r times
- DCM(k, l): set of all languages accepted by k-counter, l-reversal DCM

DCMs

- Main definition: one-way input head, with a special "end-of-tape" marker
- Many variants: 2-way, input head reversal bounded, no end-of-tape marker, etc.
- Notable variant 2DCM(1): 2-way with one reversal-bounded counter

DCMs

- The F problems are all decidable: containment, disjointness, emptiness, finiteness, universe, equivalence
- Any changes (non-deterministic, remove reversal bounds, etc.) make some F-problems undecidable

Motivation/Applications

- Frontier of decidability: how far can we push DCMs while keeping things decidable
- Want to understand relationship between determinism and insertion
- DCMs can be applied in modelling, verification of infinite-state systems

Background: Insertion Operations on Languages

Insertion Operations

- Different ways of adding symbols to strings in a language
- Can be viewed as the inverse of quotient/deletion operations
- Deletion Operations explored at TAMC 2015

Inverse Prefix

- Words having L as a prefix
- $\text{pref}^{-1}(L) = \{wx \mid w \in L, x \in \Sigma^*\} = L\Sigma^*$
- Inserts on the end of the string
- Specific form of right language concatenation

Inverse Suffix

- Words having L as a suffix
- $\text{suff}^{-1}(L) = \{xw \mid w \in L, x \in \Sigma^*\} = \Sigma^* L$
- Inserts on the beginning of the string
- Specific form of left language concatenation

Inverse Infix

- Words having L as an infix
- $\mathrm{inf}^{-1}(L) = \{xwy \mid w \in L, x, y \in \Sigma^*\}$
  $= \Sigma^* L \Sigma^*$
- Inserts on either (possibly both) ends of the string

Inverse Outfix

- Words having L as an outfix
- $\text{outf}^{-1}(L) = \{uxv \mid uv \in L, x \in \Sigma^*\}$
- Inserts into the middle of the string
- No longer easy to express as concatenation of languages

Generalized m-Embedding

- $\text{emb}^{-1}(L, m) = \{w_0 x_1 \cdots w_{m-1} x_m w_m \mid w_0 \cdots w_m \in L, w_i \in \Sigma^*, 0 \le i \le m, x_j \in \Sigma^*, 1 \le j \le m\}$
- Insert at most m strings at any point into a word in L

Insertion Operations

- All very simple operations
- Non-deterministic case: closed under these operations
- How does determinism affect these closure properties?

Research Questions

Research Questions

- Under which insertion operations is DCM closed?
- Under which insertion operations is 2DCM(1) closed?

Research Questions

- Which restricted variants of DCM are closed under these operations?
- Related: is DCM equivalent to a version with no end-marker
- How does DCM differ from other models, like deterministic pushdown?

Closure Results

End-markers

- DCM : allows counter and state transitions after reading the end-marker
- DCM_{NE} languages accepted without operations on end-marker
- NE $\rightarrow$ "No End-marker"
- Result: $\text{DCM}(1, l) = \text{DCM}_{\text{NE}}(1, l)$ for any l.

Proof Intuition

- Language of accepting counter values for each state: regular, unary
- Make *DFA* for accepting counter values
- Product construction, store in state info which DFA state current counter value is in

Non-Exiting

- If $L \in \mathrm{DCM}_{\mathrm{NE}}$, $R \in \mathrm{REG}$, then $LR \in \mathrm{DCM}_{\mathrm{NE}}$
- Track subset of states of R can be in, add initial of R every time in final of L
- Corollary: $\mathrm{DCM}(1, k)$ closed under right concatenation with REG

Limited Results

- If $L_1 \in \text{DCM}_{\text{NE}}$ and prefix-free, $L_2 \in \text{DCM}$, then $L_1 L_2 \in \text{DCM}$
- Means we always know when to swtich from L_1 to L_2

Showing non-closure with undecidable properties

Abstract Properties

- DCM is closed under boolean operations, with decidable emptiness
- 2DCM(1), closed under the same properties
- We use this to show that neither class is closed under inverse-infix

Non-closure under inverse infix

- We show that there is $L \in \text{DCM}(1,1)$ such that $\Sigma^* L \Sigma^* \notin \text{DCM} \cup \text{2DCM}(1)$

A language not in either class

- Turing machine equivalent to a 2-counter machine (no reversal bounds)
- Take 2-counter machine T accepting RE language that's not R.
- Let L_0 be the language of valid (partial) runs of T

A language not in either class

- Suppose $L_0 \in$ DCM or DCM(1)
- Make $R \in$ REG, runs starting starting with input n, end in final state
- $L_0 \cap R = \emptyset \iff n \notin L(T)$, contradicts undecidability
- Same argument for 2DCM(1)

Intuition for non-closure

- Words $\#I\#I'$ where I to I' is not a valid transition of T.
- Easy to accept with DCM$(1, 1)$
- Take inf^{-1}, intersect with "structure" language, complement
- Gives us L_0, set of all valid runs

Non-closure under inverse prefix

- We show that there is $L \in \mathrm{DCM}(2,1) \cap \mathrm{2DCM}(1)$ such that $L\Sigma^* \notin \mathrm{DCM} \cup \mathrm{2DCM}(1)$
- Notable: different than deterministic pushdown

Proof Intuition

- Let $L = \{\#w\# \mid |w|_a \neq |w|_b\}$
- Take use inverse prefix, complement, intersect with REG to get $\{\#a^{k_1}b^{k_1}\# \cdots \#a^{k_m}b^{k_m}\# \mid m > 0\}$
- Intuitive: Can't compare $m + 1$ values if we're only allowed m reversals
- Proof shows how we can use closure properties to construct DCM for L_0

Easy Corollaries

End-marker matters

- $\mathsf{DCM}_{\mathsf{NE}}$ closed under right concatenation with REG
- Showed $L \in \mathsf{DCM}(2,1)$ where $\mathsf{pref}^{-1}(L) \notin \mathsf{DCM}$
- So $L \notin \mathsf{DCM}_{\mathsf{NE}}$
- $\mathsf{DCM} \neq \mathsf{DCM}_{\mathsf{NE}}$

Non-closure under inverse suffix

- This immediately follows from our previous results
- We know $\mathsf{DCM}(1, 1)$ closed under pref^{-1}
- If it were closed under suff^{-1}, it would be closed under inf^{-1}

Non-closure under inverse outfix

- Suppose it is closed. Then if $L \in$ DCM, then $\{\%yx \mid x \in L, y \in \Sigma^*\}$ is also
- Take left-quotient with %, gives $\text{suff}^{-1}(L)$
- Implies non-closure for embedding

Conclusions

- Completely characterized closure of main insertion operations for DCM, 2DCM(1)
- Determinism ruins insertion closure for DCM
- End-marker operations make DCM strictly more powerful
- Can construct concatenations in some special cases

Summary for DCM

The question: For all $L \in \mathrm{DCM}(k, l)$:

Operation	is $Op(L) \in \mathrm{DCM}(k, l)$?	is $Op(L) \in \mathrm{DCM}$?
$\mathrm{pref}^{-1}(L)$	Yes if $k = 1, l \geq 1$ No if $k \geq 2, l \geq 1$	Yes if $k = 1, l \geq 1$ Yes if $L \in \mathrm{DCM}_{\mathrm{NE}}$ No otherwise if $k \geq 2, l \geq 1$
$\mathrm{suff}^{-1}(L)$	No if $k, l \geq 1$	No if $k, l \geq 1$
$\mathrm{inf}^{-1}(L)$	No if $k, l \geq 1$	No if $k, l \geq 1$
$\mathrm{outf}^{-1}(L)$	No if $k, l \geq 1$	No if $k, l \geq 1$
LR	Yes if $k = 1, l \geq 1$ Yes if $L \in \mathrm{DCM}_{\mathrm{NE}}$ No otherwise if $k \geq 2, l \geq 1$	Yes if $k = 1, l \geq 1$ Yes if $L \in \mathrm{DCM}_{\mathrm{NE}}$ No otherwise if $k \geq 2, l \geq 1$
RL	Yes if R prefix-free No otherwise if $k, l \geq 1$	Yes if R prefix-free No otherwise if $k, l \geq 1$
$L_{\mathrm{DCM}} L$	No if $k, l \geq 1$	No if $k, l \geq 1$
$L_{\mathrm{DCM}_{\mathrm{NE}}} L$	No if $k, l \geq 1$	Yes if $L_{\mathrm{DCM}_{\mathrm{NE}}}$ prefix-free No otherwise if $k, l \geq 1$

Summary for 2DCM(1)

- There exists $L \in \text{DCM}(1,1)$ (one-way), s.t. $\text{suff}^{-1}(L) \notin \text{2DCM}(1)$
- There exists $L \in \text{DCM}(1,1)$ (one-way) , R regular, s.t. $RL \notin \text{2DCM}(1)$
- There exists $L \in \text{DCM}(1,1)$ (one-way), s.t. $\text{outf}^{-1}(L) \notin \text{2DCM}(1)$
- There exists $L \in \text{DCM}(1,1)$ (one-way), s.t. $\text{inf}^{-1}(L) \notin \text{2DCM}(1)$
- There exists $L \in \text{2DCM}(1)$, 1 input turn, 1 counter reversal, s.t. $\text{pref}^{-1}(L) \notin \text{2DCM}(1)$
- There exists $L \in \text{2DCM}(1)$, 1 input turn, 1 counter reversal, R regular, s.t. $LR \notin \text{2DCM}(1)$

Questions?