L* Method

It's hard to partition state machine.

Theorem

Given a DFA that represent the linearizations of a single poset, for all states s with out-edges on symbol x to $s1$ and y to $s2$, there are out-edges from $s1$ on y to $s3$ and $s2$ on x to $s3$.

Corollary

Given a DFA, if there is a state s with out-edge on symbol x to $s1$ and y to $s2$, if there isn't out-edge from $s1$ on y to $s3$ and $s2$ on x to $s3$, then that DFA isn't a single poset.

> This is that first property we've talked about.

Theorem

Let P be a poset on S and $\mathcal{L}(P)$ be the set of linearizations of P. Let u and u' be the prefixes of some strings in $\mathcal{L}(P)$. Since the language of the linearizations of a poset is finite, it it regular and there's a DFA accepting exactly $\mathcal{L}(P)$. Let $\equiv$ be the Nerode relation with respect to the language $\mathcal{L}(P)$.

$u \equiv u'$ iff u and u' are permutations of each other.

Proof

We shall prove both ways by contradiction.

(left to right)
Suppose $u \equiv u'$ but u and u' are not permutations of each other, then there exists a string x in the symmetric difference of substrings of u and u'.

Let v be the remaining string of the linearization corresponding to u. Suppose x is a substring of u' and is not a substring of u, then, since uv is accepted, x is a substring of v, but, since $u \equiv u'$, $u'v$ is also accepted, where a string with two xs is accepted, which leads to a contradiction with the definition of poset.

(right to left)
Suppose u and u' are permutations of each other but $u \not\equiv u'$, then there exists a string v that distinguishes uv and $u'v$ for P.

Let it be the case that $uv \in \mathcal{L}(P)$ and $u'v \notin \mathcal{L}(P)$. Because u and u' are the prefixes of linearizations in $\mathcal{L}(P)$, there must be some suffixes corresponding to them. Let the said distinguishing extension, v, corresponds to u and some permutation of v, v', to u'; that is, $uv \in \mathcal{L}(P)$ and $u'v' \in \mathcal{L}(P)$. By the following lemma, $u'v \in \mathcal{L}(P)$ and $uv' \in \mathcal{L}(P)$, a contradiction.

Lemma

When u and u' are permutations of each other, if $uv \in \mathcal{L}(P)$ and $u'v' \in \mathcal{L}(P)$, then $uv' \in \mathcal{L}(P)$ and $u'v \in \mathcal{L}(P)$

Proof

Given uv and $u'v'$, construct $P' = uv \cap u'v'$.

To show that, for any P, if $uv \in \mathcal{L}(P)$ and $u'v' \in \mathcal{L}(P)$, $uv' \in \mathcal{L}(P)$ and $u'v \in \mathcal{L}(P)$, we need only to show that, if $uv \in \mathcal{L}(P')$ and $u'v' \in \mathcal{L}(P')$, $uv' \in \mathcal{L}(P')$ and $u'v \in \mathcal{L}(P')$.

This is because, for any P, since $uv \in \mathcal{L}(P)$ and $u'v' \in \mathcal{L}(P)$, $P \subseteq uv \cap u'v' = P'$, and, if a string $w \in \mathcal{L}(P')$, $w \in \mathcal{L}(P)$; that is, that is, P' is the strictest poset that contains uv and $u'v'$. (from the lemma: if a string $w \in \mathcal{L}(P1)$ and $P2 \subseteq P1$, $w \in \mathcal{L}(P2)$)

Note that $\{x \preceq y : x \in u, y \in v\} \subseteq P'$ because uv and $u'v'$ are linear orders, and that, when u,u' and v,v' are reverses of each other, P' is exactly that; that is, if, $u \cap u' = v \cap v' = \emptyset$, $P' = \{x \preceq y : x \in u, y \in v\}$.

There are four cases regarding the relation between uv and $u'v'$; namely, $u = u'$ and $v = v'$, $u \neq u'$ and $v = v'$, $u = u'$ and $v \neq v'$, and $u \neq u'$ and $v \neq v'$.

For first three cases, because of the identity relations and transitivity of identity relation, trivially, $uv' \in \mathcal{L}(P')$ and $u'v \in \mathcal{L}(P')$.

For the last case, since elements in $u \cap v = \emptyset$, whatever the arrangement of u,u' and v,v' are, we can break P' up to two posets Pu,Pv where $Pu = u \cap u'$, $Pv = v \cap v'$, and $Pu \bigoplus Pv = P'$. Because u and u' are in Pu and v and v' are in Pv, $u'v$ and uv' are in P'. (from the lemma: if $P = P1 \bigoplus P2$ and $P1 \cap P2 = \emptyset$, then $\mathcal{L}(P) = \mathcal{L}(P1)\mathcal{L}(P2)$)

Corollary

Given a DFA, if there are w and w' such that, either that (a) $w' \not\equiv w'$ and w and w' are permutations, or that (b) $w' \equiv w'$ and w and w' are not permutations, then the language of the DFA cannot be the linearizations of a single poset. Moreover, if DFA represents a set of poset cover linearizations, (b) is impossible, as $w \equiv w'$ would imply that w and w' are permutations.

> This is pretty useless for checking if some subset are in a single poset, since checking permutations requires factorial time and the intersect and generate method is much better in polynomial time. I've come the conclusion that DFA is too lenient for our problem, and I've pretty much given up on it. At this point, if there is indeed some property that could be exploited, I doubt I'll find it.

Swap Graph Method

- Pruesse and Ruskey: Generating Linear Extensions Fast
- Heath and Nema: The Poset Cover Problem

Theorem

Let Υ be a set of linearizations, let $\mathcal{G}(\Upsilon)$ be the adjacent transposition graph for Υ, and let $\mathcal{P}$ be a minimal poset cover for Υ. For linearizations w and w', if there exists some $P \in \mathcal{P}$ such that $\{w, w'\} \subseteq \mathcal{L}(P)$, then w and w' are connected in $\mathcal{G}(\Upsilon)$.

Proof

Per the following lemma, since $\{w, w'\} \subseteq \mathcal{L}(P)$, w and w' are connected in \mathcal{G}§. Since Υ is covered by $\mathcal{P}$, by definition, $\Upsilon = \bigcup_{P\in\mathcal{P}} \mathcal{L}(P)$, $\mathcal{G}(\Upsilon) = \bigcup_{P\in\mathcal{P}} \mathcal{G}(P)$. Thus, w and w' are connected in \mathcal{G}(\Upsilon).

Lemma

Adjacent transposition graph for the linearizations of a poset is connected.

Proof

Per *Pruesse and Ruskey*(without proof; trivial? idk), the transposition graph for the linear extensions of a single poset is connected. Per *Heath and Nema*(without proof; citing Pruesse and Ruskey), the adjacent transposition graph for the linear extensions of a single poset is connected.

The adjacent transposition graph is obtained by taking the subgraph of the transposition graph such that only the edges for juxtaposed transposition remain. Note that the nodes are the same for the two graphs. I shall prove the claim by Heath and Nema by granting that Pruesse and Ruskey are indeed correct.

Given a poset P, let $\mathcal{G}(P)$ be its transposition graph and $\mathcal{G}'(P)$ be its adjacent transposition graph. Supposed $\mathcal{G}(P)$ is connected and $\mathcal{G}'(P)$ is not connected, then there are two distinct nodes n and n' such that there is at least a path between them in $\mathcal{G}(P)$ but none in $\mathcal{G}'(P)$.

If there is a path between n and n' in $\mathcal{G}(P)$, then you can obtain n' from n by transpose some pairs of elements some number of times. Since any one step of non-immediate transposition can be achievement with some number of steps of immediate transposition, as in the swap operation of bubble sort, you can obtain n' from n from some number of immediate transpositions.

Furthermore, for any edge (v, v') in $\mathcal{G}(P)$, if it is not constructed by immediate transposition, then there is a path from v to v' taking only those that are, because, suppose $v = uawbv$ and $v' = ubwav$, since they are the linearizations of a single poset, a and b must be incomparable with each other and everything in w in that poset, so the linearizations from every step of immediate transposition from v to v' are also in that poset.

Since there is a sequence of immediate transpositions that can permute n to n', there is a path taking only edges constructed by immediate transpositions in $\mathcal{G}(P)$ between n and n'. Since making $\mathcal{G}'(P)$ out of $\mathcal{G}(P)$ does not remove edges constructed by immediate transpositions, that path is still present in $\mathcal{G}'(P)$, but then there is a path between n and n' in $\mathcal{G}'(P)$, a contradiction.

Since, if $\mathcal{G}(P)$ is connected, then $\mathcal{G}'(P)$ is connected, and, according to Pruesse and Ruskey, $\mathcal{G}(P)$ is connected, $\mathcal{G}'(P)$ is connected.

> The proof is, essentially, (1) $\mathcal{G}(P)$ is connected; (2) they are connected by immediate transposition edges; (3) the process of making $\mathcal{G}'(P)$ doesn't remove those edges; (4) $\mathcal{G}'(P)$ is connected.

Corollary

If w and w' are not connected, they are not of the same poset in the minimal poset cover.

Proof

This is immediate from contraposition of the theorem.

> This is the most useful thing from swap graph. It's like divide and conquer. The complexity of contructing swap graph is pretty good: $\mathcal{O}(m^2 \cdot n)$, where n is the size of the universe and m is the size of Υ.

Goal

Let Υ be a set of linearizations, let $\mathcal{G}(\Upsilon)$ be the adjacent transposition graph for Υ, and let $\mathcal{P}$ be a minimal poset cover for Υ. For any connected component $\mathcal{C} \subseteq \mathcal{G}(\Upsilon)$, there exists a set of posets $\mathcal{PC} \subseteq \mathcal{P}$ such that $\mathcal{C} = \bigcup_{P \in \mathcal{PC}} \mathcal{L}(P)$ and that, for all $P \neq P' \in \mathcal{PC}$, $\mathcal{L}(P) \cap \mathcal{L}(P') = \emptyset$.

> I don't know how to prove this but I also couldn't find a counter-example. If the non-overlapping property holds, then finding poset cover in connected subgraph can be reduced to graph partitioning problem, which would probably help SAT encoding.