NDMI113 – Extremal Combinatorics

Mykhaylo Tyomkyn

Lecture 8 - Synchronization of automata.

Rather unexpectedly, the weak saturation of cliques/the skewed uniform Two families theorem has an application in the area of finite automata.

Definition 1 (DFA) *A* Deterministic Finite Automaton (DFA) *is a triple $G = (Q, \Sigma, \delta)$, where Q is a finite set of* states*, Σ is the* alphabet*, i.e. a finite set of* letters*, and $\delta : Q \times \Sigma \to Q$ is the* transition function*.*

We can view a DFA as a directed graph with loops and multiple edges, where the vertices are the states $q \in Q$. The digraph is out-regular, i.e., each vertex has the same number of outgoing edges. Furthermore, the alphabet Σ partitions the edges into 'colour classes', so that each vertex has exactly one outgoing edge of every colour.

The function δ can be transitively extended to $Q \times \Sigma^*$, where Σ^* are the *words*, i.e. finite sequences of letters: if $w = a_1 \dots a_k \in \Sigma^*$, we put[1]

$$\delta(q, w) := \delta(\dots \delta(q, a_1), a_2) \dots, a_k) \dots).$$

DFA's are fundamental objects in Theoretical Computer Science, with the emphasis on defining certain languages (subsets of Σ^*). To this end, a typical textbook definition of a DFA includes a starting state and a set of accepting states. Our focus here is different.

Definition 2 *A synchronizing word (SW) for a DFA $G = (Q, \Sigma, \delta)$ is a word $w \in \Sigma^*$ such that $\delta(q, w)$ is the same for all $q \in Q$. A DFA is called* synchronizing *if it admits an SW.*

Note that some DFA's are not synchronizing — take for example $Q = \{0, 1\}, \Sigma = \{a, b\}$, $\delta(0, a) = 0, \delta(1, a) = 1, \delta(0, b) = 1$ and $\delta(1, b) = 0$. On the other hand, if G *is* synchronizing, a natural task is to find a SW as short as possible. To this end, let $w(G)$ be the length of the shortest synchronizing word for G, or ∞ if none exists. And let

$$w(n) = \max\{w(G) : |Q(G)| = n,\ G \text{ is synchronizing}\}.$$

A first bound on $w(n)$ can be obtained as follows.

Definition 3 *The Power automaton of G is the DFA $P(G) = \{2^Q, \Sigma, \delta\}$, where δ is to be viewed as a set mapping: $\delta(S) = \{\delta(s) : s \in S\}$.*

Notice that G is synchronizing if and only if there exists a directed path (along any colours) from the state $Q \in 2^Q$ to a singleton state $\{q\}$. This immediately gives $w(n) \leq 2^n$, which is far from best possible though.

[1] Abusing notation, we use δ in both cases.

Lemma 1 *G is synchronizing if and only if for all pairs $\{x, y\} \in \binom{Q}{2}$, there exists a word $w(x, y) \in \Sigma^*$ such that $\delta(x, w) = \delta(y, w)$, i.e. w synchronizes the pair.*

Proof If G is synchronizing, then the synchronizing word is in particular synchronizing for any pair. For the reverse direction, suppose that G synchronizes pairs, and put $S_0 = Q$. For an arbitrary pair of states $\{x_0, y_0\} \subseteq S_0$ let $w_0 = w(x_0, y_0)$ be their synchronizing word. Let $S_1 = \delta(S_0, w_0)$, and note that $|S_1| < |S_0|$. If $|S_1| = 1$ then G is synchronizing. If $|S_1| > 1$, take an arbitrary pair $\{x_1, y_1\} \subseteq S_1$, its synchronizing word w_1, apply $w_0 w_1$ to obtain $\delta(Q, w_0 w_1) = S_2$ with $|S_2| < |S_1|$, and proceed inductively. □

A given pair of states $\{x, y\} \subseteq Q$ can be synchronized if and only if in $P(G)$ there is a directed path from state $\{x, y\}$ to a singleton state. Since this would only involve subsets of Q of size at most 2, such a path would have length at most $\binom{|Q|}{2}$. Since the iterative procedure from the proof of Lemma 1 requires at most $|Q|$ 'shrinking' steps (going from S_i to S_{i+1}), we obtain

$$w(n) \leq n\binom{n}{2} \sim \frac{n^3}{2}.$$

In particular, synchronization can be decided in polynomial time.

This is now a good opportunity to discuss the lower bounds on $w(n)$. The following is conjectured to be optimal.

Conjecture 1 (Černý '64)

$$w(n) = (n-1)^2.$$

The value of $(n-1)^2$ is realized by the *Černý automaton* C^n, which we now present. Let $Q = [n]$ and $\Sigma = \{a, b\}$. Let $\delta(q, b) = q - 1 \mod n$, for all q, let $\delta(1, a) = n$ and $\delta(q, a) = q$ for $q \neq 1$. One synchronizing word for this automaton is $ab^{n-1}ab^{n-1} \dots ab^{n-1}a$, where b^{n-1} is repeated $n-2$ times, it has length $(n-1)^2$ (note in comparison that to synchronize two states x, y of C^n it is sufficient, and in the worst case also necessary, to use roughly $n^2/2$ letters). It turns out that one cannot do better.

Theorem 1 *$w(C^n) \geq (n-1)^2$. Hence, $w(n) \geq w(C^n) = (n-1)^2$.*

Proof Let w be a SW for C^n and for $1 \leq k \leq |w|$ let w_k be its prefix of length k. Consider the following one-player game. Suppose that initially on each state $q \in [n]$ there is a coin. At every stage of the game applying letter b takes the configuration of coins S and maps it to $\delta(S, b) = \{s - 1 \mod n : s \in S\}$. Informally, we place the states on a circle in anti-clockwise order, and b 'rotates' the coins by 1 clockwise. Applying letter a similarly maps S to $\delta(S, a)$. Note that here we lose one coin when S contained both 1 and n. In such case we view the coin that was on 1 as 'disappearing' and the coin on 1 as 'remaining'.

After applying all of w there will be one last coin standing, which we call the *golden coin*. At time k (that is, after the application of w_k), for any coin c, define $p_k(c) \in [n]$ to be its state and let $d_k(c) = p_k(c) - p_k(g) \mod n$, where g is the golden coin – this is the 'distance' between c and g. Define [2]

$$Z_k := \max_c \{n d_k(c) + p_k(c)\},$$

[2]Important: in this definition the addition is in $\mathbb{Z}$, not in Z_n!

where the maximum is taken over all coins still on the board after we have applied w_k.

Observe that initially we have $Z_0 \geq (n-1)n+1$, while eventually we will have $Z_{|w|} \leq n$, since there will be only one coin left, the golden coin. Hence, $Z_0 - Z_{|w|} \geq (n-1)^2$. We now claim that

$$Z_k - Z_{k-1} \geq -1$$

for all k. This would readily imply the theorem statement.

Indeed, let c be the coin realizing Z_{k-1}. If the next letter of w is a b, then $d_k(c) - d_{k-1}(c) = 0$, whereas $p_k(c) - p_{k-1}(c)$ is either -1 or positive. Hence,

$$Z_k \geq nd_k(c) + p_k(c) \geq Z_k - 1.$$

On the other hand, if the next letter of w is a, then the following cases can happen. If both g and c do not move, then $d_k(c) = d_{k-1}(c)$ and $p_k(c) = p_{k-1}(c)$, so $Z_k \geq Z_{k-1}$. If g moves and c does not, then $d_k(c) > d_{k-1}(c)$, so $Z_k > Z_{k-1}$. If c moves and g does not, then $d_k(c) = d_{k-1}(c) - 1$, while $p_k(c) = p_{k-1}(c) + n - 1$. So, again

$$Z_k \geq nd_k(c) + p_k(c) = n(d_{k-1}(c) - 1) + p_{k-1}(c) + n - 1 = Z_k - 1.$$

In this case it might have happen that c was subsumed by another coin sitting at state n. In this case, repeat the above calculation with that coin, for the same result. Lastly, it cannot happen that both g and c move, since a changes the value of only one state. □

Returning to the upper bounds, the following improvement due to Frankl remained the best known bound until more recently it was further improved (Shitov '17) to roughly $n^3/6 - n^3/1000$.

Theorem 2 (Frankl '82)

$$w(n) \leq \frac{n^3 - n}{6}.$$

In order to prove this, we inspect the proof of Lemma 1 and bound the length of a word synchronizing a pair from a given set of states.

Lemma 2 *Let G be synchronizing and $S \subseteq Q$ with $|S| = s \geq 2$. Then there exists a word $w \in \Sigma^*$ of length at most $\binom{n-s+2}{2}$ synchronizing some pair of states in S.*

With this lemma in hand, Theorem 2 follows by the argument of Lemma 1. If G is synchronizing, it admits a synchronizing word w, which is a concatenation of words $w_0 \dots w_{n-2}$, where each w_i synchronizes some pair in a set $S_i \subseteq Q$ of size $n-i$ (note that some w_i might be empty), and we have

$$|w| = \sum_{i=0}^{n-2} |w_i| \leq \sum_{i=0}^{n-2} \binom{n-(n-i)+2}{2} = \binom{n+1}{3} = \frac{n^3-n}{6}.$$

Proof [of Lemma 2.] Let w_S be the shortest synchronizing for any pair of states in S (such a word exists since G is synchronizing), and let x_0 and y_0 be states of $S^0 = S$ that are synchronized by w_S. For each $0 \leq j \leq |w_S| - 1$ let w^j be the prefix of w_S of length j, and define $S^j = \delta(S, w^j)$, $x_j = \delta(x_0, w^j)$, $y_j = \delta(y, w^j)$ and $XY_j = \{x_j, y_j\}$.

Notice now that for all $j < \ell$ we have $XY_\ell \not\subseteq S^j$, as otherwise there would be some pair of states $\{x', y'\} \subseteq S$ with

$$\delta(\{x', y'\}, w^j) = XY_\ell = \delta(\{x_0, y_0\}, w^\ell),$$

so we could replace in w_S the longer prefix w^ℓ with the shorter w^j (equivalently, delete all letters in positions $j+1$ through ℓ) to obtain a shorter synchronizing word for x' and y'.

Now, setting $T_j = Q \setminus S^j$ for all j (note that $|T_j| = n-s$), we get $XY_j \cap T_j = \emptyset$ for all j and $XY_\ell \cap T_j \neq \emptyset$ for all $j < \ell$. Therefore, the conditions of the skewed uniform Two families theorem are fulfilled, and the theorem gives

$$|w_S| \leq \binom{(n-s)+2}{2}.$$

□

We finish with two open questions.

Question: Is it true that for every state q in a synchronizing DFA there is a word w of length $O(n)$ such that $q \notin \delta(Q, w)$. That is, we are looking for an *avoiding* word for q of linear length.

Question: Is it true that for every state x in a synchronizing DFA, there is a state y such that $\{x, y\}$ can be synchronized by a word of length $O(n)$?