Infinite Automata 2025/26

Lecture Notes 13

Henry Sinclair-Banks

Recall from Lecture 12 that we wish to prove the following theorem.

Theorem 12.7. Reachability (i.e. deciding whether the C-computed set of runs from the initial counter valuation is non-empty) in programs without zero tests with $3k+2$ counters is F_k-hard.

It remains to show that one can build an $F_k(n)$-multiplier of size *poly*(n); recall the definition of a B-multiplier.

Definition 12.10. A program M (without zero tests) with counter C that z-computes, from the zero valuations $\{\mathbf{0}\}$, the set $\mathit{Ratio}(B, \mathsf{b}, \mathsf{c}, \mathsf{d}, \mathsf{C})$, for some four of its counters $\mathsf{z}, \mathsf{b}, \mathsf{c}, \mathsf{d} \in C$ is called a *B-multiplier.*

Later, we will use the following 4-multiplier; this program is called `multiplier`$(4, \mathsf{b}, \mathsf{c}, \mathsf{d})$.

$\mathsf{b} \mathrel{+}= 4$
`loop:`
$\quad \mathsf{c} \mathrel{+}= 1, \mathsf{d} \mathrel{+}= 4.$

As mentioned in Lecture 12, in order to construct greater multiplies (of a reasonable size), we will use *amplifiers*.

Definition 13.1. Let P be a counter program without zero tests with counters C. Let $\mathsf{b}, \mathsf{c}, \mathsf{d} \in \mathsf{C}$ be three distinguished "input counters" and let $\mathsf{b}', \mathsf{c}', \mathsf{d}' \in \mathsf{C}$ be three distinguished "output counters". P is an *F-amplifier* if, for every $B \in 4\mathbb{N}$, it $\{\mathsf{d}\}$-computes, from $\mathit{Ratio}(B, \mathsf{b}, \mathsf{c}, \mathsf{d}, \mathsf{C})$, the set $\mathit{Ratio}(F(B), \mathsf{b}', \mathsf{c}', \mathsf{d}', \mathsf{C})$.

We will now give an example which provides a family of (linear) amplifiers. The following program is called `linear`$(\ell, \mathsf{b}, \mathsf{c}, \mathsf{d}, \mathsf{b}', \mathsf{c}', \mathsf{d}')$, for $\ell \in \mathbb{N}$ and six counters $\mathsf{b}, \mathsf{c}, \mathsf{d}, \mathsf{b}', \mathsf{c}', \mathsf{d}'$.

1. `loop:`
2. $\quad$ `loop:`
3. $\quad\quad \mathsf{c} \mathrel{-}= 1, \mathsf{c}' \mathrel{+}= 1, \mathsf{d} \mathrel{-}= 1, \mathsf{d}' \mathrel{+}= \ell$
4. $\quad$ `loop:`
5. $\quad\quad \mathsf{c}' \mathrel{-}= 1, \mathsf{c} \mathrel{+}= 1, \mathsf{d} \mathrel{-}= 1, \mathsf{d}' \mathrel{+}= \ell$
6. $\quad \mathsf{b} \mathrel{-}= 2, \mathsf{b}' \mathrel{+}= 2\ell$
7. `loop:`
8. $\quad \mathsf{c} \mathrel{-}= 1, \mathsf{c}' \mathrel{+}= 1, \mathsf{d} \mathrel{-}= 2, \mathsf{d}' \mathrel{+}= 2\ell$
9. $\mathsf{b} \mathrel{-}= 2, \mathsf{b}' \mathrel{+}= 2\ell$

With $\ell = 1$, `linear`$(\ell, \mathsf{b}, \mathsf{c}, \mathsf{d}, \mathsf{b}', \mathsf{c}', \mathsf{d}')$ is called the identity-amplifier.

Claim 13.2. The program `linear`$(\ell, \mathsf{b}, \mathsf{c}, \mathsf{d}, \mathsf{b}', \mathsf{c}', \mathsf{d}')$ is an L_ℓ-amplifier, where $L_\ell : 4\mathbb{N} \to 4\mathbb{N}$ is defined by $L_\ell(n) = \ell \cdot n$.

Proof sketch. We shall write counter valuations as vectors $(\mathsf{b}, \mathsf{c}, \mathsf{d}, \mathsf{b}', \mathsf{c}', \mathsf{d}')$ in $\mathbb{N}^6$. We start at the counter valuation $(B, s, Bs, 0, 0, 0)$. Since we wish to look at runs that get d to 0, we can observe that the outer loop (defined by line `1` to line `6`) needs to be executed one less than is maximally possible, and inside each iteration of the outer loop, the two inner loops (defined by lines `2` and `3` as well as lines `4` and `5`) must be executed maximally. This leads to reaching the counter valuation $(2, s, 2s, \ell(B-2), 0, \ell s(B-2)$ before executing line `7`. After, the final loop (defined by lines `7` and `8`) is then executed maximally to reach the counter valuation $(0, 0, 0, \ell B, s, \ell s B) \in \mathit{Ratio}(B, \mathsf{b}', \mathsf{c}', \mathsf{d}', \mathsf{C})$. $\square$

15th Janurary 2025

Recall the definition of the fast growing function (Definition 12.1). Specifically, recall that we defined $F_1(n) = 2n$ (so $F_1 = L_2$), and

$$F_k(n) = \underbrace{F_{k-1} \circ \ldots \circ F_{k-1}}_{\frac{n}{4} \text{ times}}(4).$$

In fact, we shall use introduce *amplifier lifting notation.*

$$\widetilde{F}(n) = \underbrace{F \circ \ldots \circ F}_{\frac{n}{4} \text{ times}}(4).$$

This means that we can define the family of fast growing functions by $F_1(n) = 2n$ and $F_{k+1} = \widetilde{F_k}$.

Now, Let P be a program without zero tests with counters C; let $\mathsf{b}_1, \mathsf{c}_1, \mathsf{d}_1 \in \mathsf{C}$ be three distinguished input counters and let $\mathsf{b}_2, \mathsf{c}_2, \mathsf{d}_2 \in \mathsf{C}$ be three distinguished output counters. We will now describe the transformation of P into $\widetilde{P}$ (which also does not contain zero tests) with the following property. If P is an F-amplifier, then $\widetilde{P}$ is an $\widetilde{F}$-amplifier (for some function $F : 4\mathbb{N} \to 4\mathbb{N}$). The program P uses counters $\widetilde{\mathsf{C}} = \mathsf{C} \cup \{\mathsf{b}, \mathsf{c}, \mathsf{d}\}$; where b, c, and d are three fresh counters only used by $\widetilde{P}$. The three distinguished input counters of $\widetilde{P}$ are b, c, and d, and the three distinguished output counters of $\widetilde{P}$ are b_2, c_2, and d_2.

Roughly speaking, $\widetilde{P}$ will implement the computation of $\widetilde{F}$ according to its definition; with $2\ell+1$ zero tests it $\{\mathsf{d}_1\}$-computes, from $\{\mathbf{0}\} \subseteq \mathbb{N}^{\mathsf{C}}$, the set $\mathit{Ratio}(F^{(\ell+1)}, \mathsf{b}_2, \mathsf{c}_2, \mathsf{d}_2, \mathsf{C})$. We use the triplet of counters $\mathsf{b}, \mathsf{c}, \mathsf{d}$ to handle these $2\ell + 1$ zero tests.

To construct the program $\widetilde{P}$, we will use the identity-amplifier $I = \mathtt{linear}(1, \mathsf{b}_2, \mathsf{c}_2, \mathsf{d}_2, \mathsf{b}_1, \mathsf{c}_1, \mathsf{d}_1)$ and the 4-multiplier $M = \mathtt{multiplier}(4, \mathsf{b}_1, \mathsf{c}_1, \mathsf{d}_1)$ (that was presented on Page 1). The following program is $\widetilde{P}$. We will be using the zero test gadgets on counter d_1 and d_2 to ensure certain computations (like P and I are fully executed); this means that d_1 and d_2 will take on roles like x and y from Lecture 12 when they had their zero tests simulated by a triplet of counters. Here, we will use b, c, and d to handle the zero tests of d_1 and d_2.

```
1. M'        # M' is obtained from M by using c as an upper bound counter (as seen in Lecture 12)
2. loop:
3.     P'
4.     zero-test(d1)
5.     I'     # I' is obtained from I by using c as an upper bound counter (as seen in Lecture 12)
6.     zero-test(d2)
7. P'
8. zero-test(d1)
9. flush(c)
```

Lemma 13.3. If P is an F-amplifier, then $\widetilde{P}$ is an $\widetilde{F}$-amplifier.

Proof sketch. For every $B \in 4\mathbb{N}$, it is true that from $\mathit{Ratio}(B, \mathsf{b}_1, \mathsf{c}_1, \mathsf{d}_1, \mathsf{C})$, P $\{\mathsf{d}_1\}$-computes the set $\mathit{Ratio}(F(B), \mathsf{b}_2, \mathsf{c}_2, \mathsf{d}_2, \mathsf{C})$. This computation is simulated by lines 3 and 4 as well as lines 7 and 8 in $\widetilde{P}$. The program $I = \mathtt{linear}(1, \mathsf{b}_2, \mathsf{c}_2, \mathsf{d}_2, \mathsf{b}_1, \mathsf{c}_1, \mathsf{d}_1)$ is just used to shift the value of b_2 to b_1, the value of c_2 to c_1, and the value of d_2 to d_1; in other words, from $\mathit{Ratio}(B, \mathsf{b}_2, \mathsf{c}_2, \mathsf{d}_2, \mathsf{C})$, I $\{\mathsf{d}_2\}$-computes the set $\mathit{Ratio}(B, \mathsf{b}_1, \mathsf{c}_1, \mathsf{d}_1, \mathsf{C})$. This computation is simulated by lines 5 and 6 in $\widetilde{P}$.

We therefore observe that executing the main loop (defined by lines 2 to 6) a total of ℓ times and then executing lines 7 and 8 will lead to the computation of $\mathit{Ratio}(F^{(\ell+1)}(4), \mathsf{b}_2, \mathsf{c}_2, \mathsf{d}_2, \widetilde{\mathsf{C}})$. Thus if we set $B = 4(\ell + 1) \in 4\mathbb{N}$, we know that by initialising b, c, and d to $\mathsf{b} = B$, $\mathsf{c} = s$, and $\mathsf{d} = Bs$, then we can simulate these $2\ell + 1$ zero tests required to compute $\widetilde{F}(B) = F^{(\ell+1)}(4)$. Precisely, from $\mathit{Ratio}(B, \mathsf{b}, \mathsf{c}, \mathsf{d}, \widetilde{\mathsf{C}})$, $\widetilde{P}$ $\{\mathsf{d}\}$-computes the set $\mathit{Ratio}(\widetilde{F}(B), \mathsf{b}_2, \mathsf{c}_2, \mathsf{d}_2, \widetilde{\mathsf{C}})$ (which makes $\widetilde{P}$ an $\widetilde{F}$-amplifier). $\square$

We are now finally able to prove Theorem 12.7.

Sketch of proof of Theorem 12.7. We can use Lemma 13.3 multiple times to lift amplifiers to F_k. Let $k \in \mathbb{N}$ and $n \in 4\mathbb{N}$, we compute (in linear with with respect to k and n) an F_k-amplifier P_k with $3k+3$ counters C by applying the amplifier lifting transformation $P \to \widetilde{P}$ (stated just before Lemma 13.3) starting from the F_1-amplifier $L_2 = \texttt{linear}(2, \mathsf{b}_1, \mathsf{c}_1, \mathsf{d}_1, \mathsf{b}_2, \mathsf{c}_2, \mathsf{d}_2)$.

Let $\mathsf{b}, \mathsf{c}, \mathsf{d} \in \mathsf{C}$ be the three distinguished input counters of P_k. Using Claim 12.11, composing $\texttt{multiplier}(n, \mathsf{b}, \mathsf{c}, \mathsf{d})$ with P_k yields an $F_k(n)$-multiplier which outputs $\mathit{Ratio}(F_k(n), \mathsf{b}_k, \mathsf{c}_k, \mathsf{d}_k, \mathsf{C})$.

Now we can use this $F_k(n)$-multiplier to simulate a counter machine which has two zero-testable counters and which uses at most $\frac{F_k(n)-1}{2}$ zero tests. Hence reachability in counter programs without zero tests with $3k+3$ counters is F_k-hard.

Lastly, we observe that the first counter b used in $\texttt{multiplier}(n, \mathsf{b}, \mathsf{c}, \mathsf{d})$ is bounded by n; b has it value set to n and after never receives an incremental update to its value. This means that we can include the current value b in the control states of the program; this only multiplicatively increases the number of control states by $n+1$ (a linear factor). Hence the F_k lower bounds holds for $3k+2$ counters. □