CS 4124
Solutions to Homework Assignment 4
Collin McDevitt

March 29, 2024

[30] 1. Let $\Sigma = \{0, 1\}$, the binary alphabet. Let

$$F = \{h \mid h : \Sigma^\star \to \Sigma^\star\},$$

the set of functions from strings to strings.

Prove that F is uncountable.

HINT: **Use the fact that $\Sigma^\star$ is countably infinite and employ a proof by diagonalization.**

My proof strategy:

I will prove that the set of functions $F' = \{h : h : \Sigma^\star \to \{0, 1\}\}$ is uncountable and as this set is a subset of F that would imply that F is uncountable. I will do this by constructing a bijection between $H : F' \to \mathcal{P}(\Sigma^\star)$ (the power set of $\Sigma^\star$) in the last homework we proved that no set can have a bijection with its power set and as $\Sigma^\star$ is countable we get that $\mathcal{P}(\Sigma^\star)$ is uncountable hence if a bijection H exists we would have that F' is uncountable.

Let $H : F' \to \mathcal{P}(\Sigma^\star)$ where $H(f) = \{w \in \Sigma^\star : f(w) = 1\}$ (informally this is the set of elements from $\Sigma^\star$ who get mapped to 1).

Now to show that this is a bijection I will come up with an inverse. We create the function

$$H^{-1} : \mathcal{P}(\Sigma^\star) \to F'$$

where for $A \in \mathcal{P}(\Sigma^\star)$ we have $H^{-1}(A) = f$ where for all $w \in \Sigma^\star$ we have $f(w) = 1$ if and only if $w \in A$. The fact that H^{-1} is not a multivalued function can be proven by assuming that for some $A \in \mathcal{P}(\Sigma^\star)$ that there exists two $f_1, f_2 \in F'$ such that $H^{-1}(A) = f_1$ and $H^{-1}(A) = f_2$ then we have that for all $w \in \Sigma^\star$ that $f_1(w) = 1$ and $f_2(w) = 1$ if and only if $w \in A$ which directly implies that for any $\omega \in \Sigma^\star$ where $\omega \notin A$ that $f_1(\omega) = 0$ and $f_2(\omega) = 0$. As f_1, f_2 are equal at every input we have that $f_1 = f_2$.

Now to show these functions are each others inverses.

Let $f \in F'$ then we have $H(f) = \{w \in \Sigma^\star : f(w) = 1\}$. Then take $H^{-1}(\{w \in \Sigma^\star : f(w) = 1\}) = f_0$ for some $f_0 \in F'$. But from the definition of H^{-1} we get that $f_0(\omega) = 1$ if and only if $\omega \in \{w \in \Sigma^\star : f(w) = 1\}$ which is the same as $f(\omega) = 1$ so we have that $f_0 = f$. This implies that $H^{-1} \circ H = \mathrm{Id}_{F'}$. Now let $A \in \mathcal{P}(\Sigma^\star)$ then we have that $H^{-1}(A) = f$ for some $f \in F'$. Then taking $H(f) = \{w \in \Sigma^\star : f(w) = 1\}$ but from the definition of f we have that for all $\omega \in \Sigma^\star$ that $f(\omega) = 1$ if and only if $\omega \in A$. This implies that $H(f) = A$ so we have $H \circ H^{-1}(A) = A$ so we have that H is the inverse of H^{-1} so we have that H is a bijection.

Then as a bijection between $F' \subset F$ to $\mathcal{P}(\Sigma^\star)$ exists and $\mathcal{P}(\Sigma^\star)$ is uncountable we get F' is uncountable and so F is uncountable as well.

```
Program P_x
Input: y
if x halts on input x
   if y has even length
     then accept
     else reject
else loop forever
```

[30] 2. Let

$$E \quad = \quad \{w \in \{0,1\}^\star \mid \ell(w) \text{ is even}\},$$

the language of binary strings of even length. See Figure 2 for a sample program that decides E, just for LaTeX formatting purposes. Let

$$L \quad = \quad \{x \mid \text{program } x \text{ decides } E\}$$

be the language of programs that decide E.

Prove that L is undecidable by using m-reducibility.

We have that DHP is undecidable hence I will create the m-reduction $DHP \leq_M L$. We do this by taking any program $x \to P_x$ where P_x is the program above. Note that E is regular as a simple FTA that has two states could recognize it. Hence I am justified in asking weather y has even length in P.

Now let $x \in \Sigma^\star$ and suppose that $x \in DHP$ then we have that from the definition of P_x that for any input y that $P_x(y)$ will accept if y has even length and reject if y has odd length. This implies that P_x decides E so we have that $P_x \in L$. Now for any $x \in \Sigma^\star$ consider $P_x \in L$ then we have that P_x would have decided E hence it halts so we have that x halts on input x hence $x \in DHP$. This implies that $x \in DHP$ if and only if $P(x) \in L$ so we have that $DHP \leq_M L$ so L is undecidable as DHP is undecidable.

---

[40] 3. Let

$$\text{DOUBLE} \quad = \quad \{ww \mid w \in \{a,b\}^\star\}.$$

Let

$$W \quad = \quad \{x \mid \text{program } x \text{ decides DOUBLE}\}$$

be the language of programs that decide DOUBLE.

A. Is DOUBLE **decidable? Justify your answer.**

B. Is W decidable? Prove your answer.

```
Program P_x
Input: y
if x halts on input x
   if y ∈ DOUBLE
     then accept
     else reject
else loop forever
```

A: Yes DOUBLE is decidable. We could create the program that takes in an input and checks if the input has even length and checks if the input is a palindrome. If both of these are true we accept otherwise we reject. This program would be able to decide DOUBLE as the only strings that are in DOUBLE are palindromes of even length.

B: We have that W is not decidable. I will show this by creating the m-reduction $DHP \leq_M W$. We do this by taking any program $x \to P_x$ where P_x is the program above. Note that DOUBLE is decidable using the same reasoning as in part A. Hence I am justified in asking weather $y \in$ DOUBLE in P_x.

Now to show that this is a valid m-reduction. Suppose that $x \in DHP$ then we have from the definition of P_x that for any input y that it either accepts or rejects as DOUBLE is decidable hence we have that $P_x \in W$. Now suppose that $P_x \in W$ then we have that P_x would have decided DOUBLE hence it halts so we have that x halts on input x hence $x \in DHP$. This implies that $x \in DHP$ if and only if $P(x) \in W$ so we have that $DHP \leq_M W$ so W is undecidable as DHP is undecidable.