Alex Hubers
CSC:341 Spring 2024

A (More) Constructive Proof of Corollary 5.23

Due: Apr 6, 2024

#1

Sipser §5.3, pp. 236 gives a proof for the following theorem.

Theorem 1 (#5.22)**.** *If $A \leq_m B$ and B is decidable, then A is a decidable.*

Sipser gives a constructive proof for this claim, meaning: if we have a decider for B and a mapping reduction f from A to B, we can "build" precisely a decider for A.

Sipser then argues that, by corollary, so too is the following claim true.

Theorem 2 (#5.23)**.** *If $A \leq_m B$ and A is undecidable, then B is undecidable.*

You could argue that this follows by contrapositive, and maybe you are right. But the problem is, it tells you nothing about how why B is undecidable. Normally, a proof like this will (towards a contradiction) suppose that B is decidable and show that, in such a case, so too is A decidable. A contradiction! Therefore B is undecidable.

We can in fact do just this:

Proof. Suppose $A \leq_m B$ and A is undecidable. Towards a contradiction, suppose B is decided by machine M_B. By assumption, we have $f : \Sigma^* \to \Sigma^*$ a mapping reduction. We will use M_B and M_f (the machine that decides f) to build M_A, a decider for A.

We do so like this.

- Let M_A take input string x.
- M_A simulates M_f on x, giving us $f(x)$ to play with.
- M_A passes this output $f(x)$ as input to M_B.
- if M_B accepts, then M_A accepts. If M_B rejects, then M_A rejects. Because we have supposed M_B is decidable, these are the only two options to consider for the behavior of M_B.

We now argue that M_A decides A.

- Suppose $x \in A$. Then $f(x) \in B$, because f is a mapping reduction. So M_b will accept $f(x)$, and therefore M_A will accept x as an input.
- Suppose $x \notin A$. Then $f(x) \notin B$, because f is a mapping reduction. So M_b will reject $f(x)$, and therefore M_B will reject x as an output.

So it looks like we have decided A! Yet we know A to be undecidable, so this is a contradiction. It follows that B must too be undecidable. Again, I will spell it clearly: if we *were* to have a decider for M_B, then we would get a decider for M_A for free. So we know M_B cannot be decidable. □

Now, keep in mind that when you give a "proof by reduction" that B is undecidable, you only have to give the mapping reduction f. The *rest* of the proof is above. This is why it does not always feel like you have really proven that B is undecidable—you have merely given a component needed for the above proof to go through.