# Solution to Exercise III.3.9 c) of Odifreddi

Duarte Maia

June 3, 2024

## 1 Introduction

This document is ostensibly a solution, and explanation thereof, of the titular exercise. This particular exercise struck me as particularly challenging, and the intended solution comes a bit out of left field. Even the hint is remarkably obtuse, especially taking into account that it's basically the skeleton of a proof. Thus, I decided to make this document in an attempt to convince to myself that the solution makes sense, and explain how someone might have come up with it.

## 2 Problem Statement

Let $A$ be a c.e. set. We say that $A$ is *dense simple* if $p_{A^c}$ dominates every total computable function. We say that $A$ is *strongly effectively simple* if there is a (wlog total) computable function $f$ such that $W_e \subseteq A^c \implies \max W_e \leq f(e)$. Prove that $A$ cannot be dense simple and s.e.s. at the same time.

## 3 Pre-Solution

At first sight this problem looks relatively approachable. An obvious avenue of attack is to suppose that $A$ is s.e.s. and construct explicitly a computable function which is not dominated by $p_{A^c}$. However, one quickly finds (or at least I did) that there doesn't seem to be an easy way to do that. In hindsight, I think that's what the full solution does – More on that later – but the motivation for the solution definitely comes from attempting a proof by contradiction: Suppose that $A$ is both dense simple and s.e.s, and let us try to find a contradiction.

Let's take a look at the puzzle pieces at hand. We have:

- An enumeration of $A$,
- A computable function $f$ that witnesses that $A$ is s.e.s,
- We are given that the principal enumeration of $A^c$ dominates every computable total function.

At first sight, the last puzzle piece doesn't quite fit together with the rest. How are we going to make the principal enumeration of $A^c$ pop up? After some thought, this is where the enumeration of $A$ comes up: We can use it to approximate $p_{A^c}$!

More concretely, define $a_s(n)$ as follows. First, enumerate $s$ elements of $A$. Then, let $a_s(n)$ be the $n$-th non-enumerated element in increasing order. This two-indexed sequence has the following property:

**Fact 1.** The sequence $a_s$ is an increasing sequence of monotone injective functions whose pointwise limit is $\lim a_s(n) = p_{A^c}(n)$.

For the sake of notation, we set $a(n) = p_{A^c}(n)$, so that $a = \lim a_s$.

Now that we have a better way to tie the pieces together, let's figure out how to manufacture a contradiction. We know that $a(n)$ dominates every total computable function, so perhaps the way lies through creating a computable function $h(n)$ for which being dominated somehow causes a contradiction. Perhaps we could try to create $h$ from $a_s$? It is as yet unclear how $f$ (the function that witnesses that $A$ is s.e.s.) comes into the picture, but let us continue along this line of reasoning for the time being.

Here's a skeleton of an idea. By making use of the recursion theorem, we can use $h$ in the definition of $h$. Moreover, we know that $a(n)$ eventually dominates $h(n)$, so perhaps we can use this to create a self-contradictory function. Namely:

$$h_0(n) = \begin{cases} \text{Compute } a_s(n) \text{ for increasing values of } s \text{ until you exceed } h_0(n). \\ \text{Output the first such value of } a_s(n). \end{cases}$$

This would certainly be a contradiction if $h_0$ were total! However, there is no reason why it should be; the obvious conclusion is simply that the computation of $h_0(n)$ will never halt for *any* values of $n$. This is where the function $f$ that witnesses s.e.s. comes in, as we can use it to conjure values out of possibly non-terminating computations. Indeed, let us define $x_1(n)$ as the code "compute $a_s(n)$ for increasing values of $s$ until you exceed $h_1(n)$ and print out the first such value of $a_s(n)$", so that

$$W_{x_1(n)} = \begin{cases} \{a_s(n)\}, & \text{for the smallest } s \text{ such that } a_s(n) > h_1(n) \text{ if one exists,} \\ \emptyset, & \text{otherwise.} \end{cases}$$

Then, we may set $h_1(n) = f(x_1(n))$, and voila! We have a total function. Now, to find that contradiction...

Let's play around with the function we've defined. We know that it is total, so there must be some $n_0$ after which we have $a(n) > h_1(n)$, and so in particular we will always be in the first branch of the definition of $W_{x_1(n)}$. Let us call the element of the resulting singleton set by the name $a_{s_1(n)}(n)$. Then, the definition of $f$ tells us: If it is the case that $a_{s_1(n)}(n)$ is in $A^c$, we *must* have $f(x_1(n)) > a_{s_1(n)}(n)$. But this is impossible because, by definition of $s_1(n)$, it must happen that $a_{s_1(n)}(n) > h_1(n) = f(x_1(n))$. This is not yet a full contradiction, but we do obtain for sure:

**Fact 2.** It must be the case that $a_{s_1(n)}(n) \in A$. In particular, $a_{s_1(n)}(n)$ cannot equal $a(n)$.

This is... Something. It's not really a contradiction yet, and we begin to run out of cards to play. It starts to look like we're at a dead end, but fortunately there's a tricky way out of this situation. Let's dig a little deeper.

We know that $a_{s_1(n)}(n)$ cannot equal $a(n)$. In other words, $a_{s_1(n)}$ hasn't finished converging at $n$. By inspecting the definition of $a_s$, we reach another conclusion however: It *must* be the case that $a_{s_1(n)}(n+1) \leq a(n)$. This seems useless for now, but it is interesting because it allows us to jump "between values of $n$", which is something we have yet to do. It requires some tinkering to make the following idea work, but hear me out: What if we can ensure that $a_{s_1(n)}(n+1) \geq h(n+1)$? This seems undoable at first sight, as the argument $s_1(n)$ has no reason to be large enough, but on a second look it requires a very simple modification in the definition of $x_1(n)$:

$$W_{x_2(n)} = \begin{cases} \{a_s(n)\}, & \text{for the smallest } s \text{ such that } a_s(n) > h_2(n) \text{ and } a_s(n+1) > h_2(n+1)\text{, if one exists,} \\ \emptyset, & \text{otherwise,} \end{cases}$$

followed by setting $h_2(n) = f(x_2(n))$. Let us see if we get a contradiction this time.

As before, if we call $s_2(n)$ the time obtained in $W_{x_2(n)}$ we obtain for $n > n_0$ (the value after which $a$ dominates $h_2$) that $a_{s_2(n)}(n) \in A$, hence $a(n) \geq a_{s_2(n)}(n+1) > h_2(n+1)$. Can we keep this inequality going for $n+2$, $n+3$, etc? It... Doesn't seem likely. For a very simple reason in fact: By construction, we will always have $a_s(n) \geq n$, and thus $h_2(n) \geq n$ for all $n > n_0$. A last modification is necessary; it will let us transpose inequalities not just from $n$ (the point of evaluation) to $n+1$, but from $y$ to $y+1$ where $y$ is not necessarily the point of evaluation. We begin by decoupling two uses of $n$ in the definition of $x_2$ which need not be the same. Define $x_3(n,y)$ by:

$$W_{x_3(n,y)} = \begin{cases} \{a_s(y)\}, & \text{for the smallest } s \text{ such that } a_s(y) > h_3(n) \text{ and } a_s(y+1) > h_3(n+1) \text{, if one exists,} \\ \emptyset, & \text{otherwise,} \end{cases}$$

and set $h_3(n) = \max_{y \le n} f(x_3(n,y))$. Why this particular computation? Well, we want to have access to values of $a_s(y)$ for $y < n$, otherwise we will definitely run into the issue we pointed out for $h_2$. As for why we take a max and not a min, say, is because otherwise the computation in the next paragraph would not work.[1]

Let $n > n_0$ as before. Moreover, define $s_3(n,y)$ as the one that shows up in the computation of $W_{x_3(n,y)}$, if one exists. First, we conclude that $h_3(n+1) < a(n)$ basically as before: If $a_{s_3(n,n)}(n)$ were in $A^c$,[2] it would be the case that $h_3(n) \ge f(x_3(n,n)) > a_{s_3(n,n)}(n) > h_3(n)$, a contradiction. Thus, $a(n) > a_{s_3(n,n)}(n)$, hence $a(n) \ge a_{s_3(n,n)}(n+1) > h_3(n+1)$.

Now we want to continue the argument by using $h_3(n+1) \ge f(x_3(n+1,n))$. There is a subtlety in this step, however: We need to ensure that $s_3(n+1,n)$ is well-defined, or equivalently that $a(n) > h_3(n+1) \wedge a(n+1) > h_3(n+2)$. Fortunately this is the case; we have just proven that $a(n) > h_3(n+1)$ for *every* $n > n_0$, including $n+1$, and so the argument continues: If $a_{s_3(n+1,n)}(n)$ were in $A^c$, we would have $h_3(n+1) \ge f(x_3(n+1,n)) > a_{s_3(n+1,n)}(n) > h_3(n+1)$, and so $a(n) > a_{s_3(n+1,n)}(n)$ whence $a(n) \ge a_{s_3(n+1,n)}(n+1) > h_3(n+2)$.

Ah, so now we're getting somewhere! Continuing by induction (using the IH to get well-definedness of $s_3(n+k,n)$) we conclude that $a(n) \ge h_3(n+k)$ for every $k \in \mathbb{N}$. In particular, $h_3$ *is a bounded function!*

This starts getting into contradiction territory, but we haven't hit the goal quite yet. We have yet to show that $h_3$ cannot be bounded. This isn't too difficult, however; we need only find a collection of values on which we can control the value of $f(x_3(n,y))$. For example, let us suppose that $h_3$ is bounded by $M$. Then, since $A$ is simple, its complement contains some element $z > M$. For this element, we have $a_0(z) = z > h_3(z)$, and so $h_3(z) \ge f(x_3(z,z)) > a_0(z) = z > M$, a contradiction! Finally, the proof is complete.

# 4 Final Notes

Even though the above proof was constructed as a proof by contradiction, we could have made it "direct" in the following way. The definition of $h_3$ only uses the fact that $A$ is c.e, as well as the definition of $f$. Thus, we could have begun by assuming that $A$ is strongly effectively simple, constructed the function $h$ as we did above, and proven that $h$ is not dominated by $A^c$. Well, I suppose this part would have been by contradiction... Still, that's another way to look at the proof, I suppose.

[1] Indeed, the author initially wrote the definition with a minimum, changing it only when he realized that he needed the opposite inequalities afterwards...

[2] Note: $s_3(n,n)$ is well-defined because $n > n_0$ and so $a(n) > h_3(n)$.