Computability III

If C is a class of languages, then co-$C = \{\overline{L} | L \in C\}$, where $\overline{L} = \Sigma^* - L =$ complement of L. Note that co-C is *not* the same as $\overline{C}$. To see this consider $C = \{L\}$, i.e. the class C consists of a single language L: $\overline{C}$ is the class of all languages L' which are not equal to L, but co-C is just the single language $\overline{L}$.

Make sure you understand the following assertions:

1. co-P = P
2. Conjecture: co-NP $\neq$ NP
3. co-decidable = decidable
4. co-semi-decidable $\neq$ semi-decidable. For example $\overline{A_{\mathrm{TM}}}$ is co-semi-decidable but not semi-decidable — we will talk about this below.

Theorem 7 L is decidable iff L is both semi-decidable and co-semi-decidable.

PROOF: $\Rightarrow$ Suppose L is decidable. Then $\overline{L}$ is decidable. Every decidable set is also semi-decidable.

$\Leftarrow$ Assume that both L and $\overline{L}$ are semi-decidable. Say $L = L(M_1)$ and $\overline{L} = L(M_2)$. Construct a decider M_3 so $L(M_3) = L$, where M_3 halts on all inputs. M_3 is given as follows:

> `On input` x`, run` M_1 `and` M_2 `''in parallel'' on input` x`.  If` M_1 `accepts` x`, then accept.  If` M_2 `accepts` x`, then reject.`

Either M_1 or M_2 must accept x, so M_3 always halts. To run M_1 and M_2 in parallel, M_3 could use two tapes, one for each machine (a 1-tape machine can simulate a 2-tape machine). Or on input $x\#x$, simulate M_1 on the first copy of x, M_2 on the second, alternating between M_1 and M_2. $\square$

Corollary 2 $\overline{A_{\mathrm{TM}}}$ is not semi-decidable.

PROOF: We know that A_{TM} is not decidable, and we also know that A_{TM} is semi-decidable. If $\overline{A_{\mathrm{TM}}}$ were semi-decidable, then by the theorem A_{TM} would be decidable, but it is not. $\square$

Definition 6 (Emptiness) $E_{\mathrm{TM}} = \{\langle M \rangle | L(M) = \emptyset\}$

Claim 2 E_{TM} is not semi-decidable.

Proof: We will show $\overline{A_{\mathrm{TM}}} \leq_m E_{\mathrm{TM}}$. Given an input $\langle M, w\rangle$ to $\overline{A_{\mathrm{TM}}}$ construct and input $\langle M'\rangle$ to E_{TM}, such that M does not accept w iff $L(M') = \emptyset$. I.e., M accepts w iff $L(M') \neq \emptyset$. M' is given as follows:

```
On input x, erase x, print w on input tape and run M.
```

If M accepts w, then $L(M') = \Sigma^*$. If M does not accept w, $L(M') = \emptyset$. □

On the other hand E_{TM} is co-semi-decidable, i.e. $\overline{E_{\mathrm{TM}}}$ is semi-decidable. To show this we construct a TM M such that $\overline{E_{\mathrm{TM}}} = L(M)$, i.e. M accepts $\overline{E_{\mathrm{TM}}}$. M is given as follows:

Let s_0, s_1, s_2, ... be an enumeration of all strings in Σ^*.

```
On input ⟨M⟩
for i : 1..∞,
    for j : 1..i
        Simulate M on input s_j for i steps.
        if M accepts s_j, then ACCEPT.
    end for
end for
```

Definition 7 (Equality) $EQ_{\mathrm{TM}} = \{\langle M_1, M_2\rangle | L(M_1) = L(M_2)\}$

Theorem 8 EQ_{TM} is not semi-decidable, and neither is $\overline{EQ_{\mathrm{TM}}}$.

Proof: We'll show that $E_{\mathrm{TM}} \leq_m EQ_{\mathrm{TM}}$ (so EQ_{TM} is not semi-decidable), and that $E_{\mathrm{TM}} \leq_m \overline{EQ_{\mathrm{TM}}}$ (so EQ_{TM} is not co-semi-decidable).

Given an input $\langle M\rangle$ to E_{TM}, construct an input $\langle M_1, M_2\rangle$ to EQ_{TM} s.t. $L(M) = \emptyset$ iff $L(M_1) = L(M_2)$. Let $M_1 = M$, and M_2 some fixed machine s.t. $L(M_2) = \emptyset$. Then $L(M) = \emptyset \Leftrightarrow L(M) = L(M_2) = \emptyset$.

For the second statement, $L(M) = \emptyset$ iff $L(M_1) \neq L(M_2)$. Take $L(M_2) = \Sigma^*$. Design M_1 so that it accepts everything iff M accepts something. □

Theorem 9 All semi-decidable sets are many-one reducible to A_{TM}.

In other words, if L is semi-decidable, then $L \leq_m A_{\mathrm{TM}}$. Analog in the theory of NP-completeness: If L is in NP, then $L \leq_m$ SAT.

Proof: Let L be semi-decidable, and suppose that M_0 semi-decides (accepts) L. In order to show that $L \leq_m A_{\mathrm{TM}}$ we must, for each $w \in \Sigma^*$, construct $\langle M, w'\rangle$ such that $w \in L$ iff $\langle M, w'\rangle \in A_{\mathrm{TM}}$ (i.e., iff M accepts w'). Take $w' = w$, $M = M_0$. This finishes the proof. □