Theorem L_{Diag} is Turing-recognizable, but not decidable.

Proof It is easily seen that the following TM recognizes L_{Diag}.

M_{Diag} := " On input w,

Determine index i s.t. $w = w_i$.

Determine (the description of) TM M_i.

Simulate M_i on w_i and Accept or Reject or Runforever accordingly."

We note that it is possible for M_{Diag} to determine the index i and m/c M_i because Σ^* and $L_{TM} = \{\langle M_1\rangle, \langle M_2\rangle, \dots\}$ are both effectively enumerable. Clearly, by design,

$$L(M_{Diag}) = \{w \mid M_{Diag} \text{ accepts } w\}$$

$$= \{w_i \mid M_i \text{ accepts } w_i\}$$

$$= L_{Diag}.$$

Now we show that L_{Diag} is undecidable. It is more convenient to show that

$$\overline{L_{Diag}} = \left\{ w_i \;\middle|\; \begin{array}{l} i \geq 1, \\ M_i \text{ does not accept } w_i \end{array} \right\}$$

is undecidabe and note the easy fact:

<u>Fact</u> A language L is decidable <u>iff</u> $\overline{L}$ is.

<u>Proof</u> If a TM M decides L then a TM that decides $\overline{L}$ can be designed so as to simulate M and in the end, switch the Accept or Reject decision of M. ▨

<u>Claim</u> $\overline{L_{Diag}}$ is undecidable. In fact $\overline{L_{Diag}}$ is not even Turing recognizable.

<u>Proof</u> Suppose on the contrary that a TM M recognizes $\overline{L_{Diag}}$. Let $k \geq 1$ be the index s.t. $\langle M \rangle = \langle M_k \rangle$ in the effective enumeration of TM-descriptions. Let $w = w_k$ be the k^{th} string in effective

enumeration of Σ^*. We show that the TM M and the language $\overline{L_{Diag}}$ "disagree" on the input w, giving a contradiction (M is supposed to recognize $\overline{L_{Diag}}$). Indeed,

$$
\begin{aligned}
M \text{ accepts } w &\iff M_k \text{ accepts } w_k && \because M = M_k,\ w = w_k \\
&\iff w_k \notin \overline{L_{Diag}} && \text{By defn of } \overline{L_{Diag}}. \\
&\iff w \notin \overline{L_{Diag}}. && \blacksquare
\end{aligned}
$$

We note that

- L_{Diag} is T.R. but not decidable.
- $\overline{L_{Diag}}$ is not even T.R.

This is an example of the following general fact.

<u>Fact</u> A language L is decidable

$\iff$ Both $L, \overline{L}$ are T.R.

In particular, if L is T.R. but not decidable, then $\overline{L}$ is not even T.R.

Proof of $\Rightarrow$ This is easy. If L is decidable, then so is $\bar{L}$, and hence both are T.R. as well.

Proof of $\Leftarrow$ Suppose $L, \bar{L}$ both are T.R.

Let M, M' be TMs that recognize $L, \bar{L}$ respectively. That is $\forall x \in \Sigma^*$,

$x \in L \Rightarrow M$ accepts x (eventually).

$x \in \bar{L} \Rightarrow M'$ accepts x (").

Now the TM $\tilde{M}$ that decides L can, on input x, simulate both M and M' on x, alternately for one more step each, and Accepts if M accepts.

Rejects if M' accepts.

$\tilde{M}$ decides L because: (AS ABOVE).

$x \in L \Rightarrow M$ accepts x

$\Rightarrow \tilde{M}$ accepts x.

$x \notin L \Rightarrow x \in \bar{L} \Rightarrow M'$ accepts x

$\Rightarrow \tilde{M}$ Rejects x. ∎

Now that we know that L_{Diag} is undecidable, we can show that several problems concerning TMs are also undecidable, by reducing L_{Diag} to these problems.

Theorem Acceptance Problem for TMs.

$A_{TM} = \{ \langle M, w \rangle \mid M \text{ is a TM that accepts } w \}$

is T.R. but undecidable.

Proof It is easily seen that the following TM recognizes A_{TM}.

$M^* :=$ " on input $\langle M, w \rangle$,

Simulate M on w.

Accept or Reject or Runforever accordin-gly."

M^* accepts $\langle M, w \rangle \iff M$ accepts w

$\iff \langle M, w \rangle \in A_{TM}$. ▨

Now we show that A_{TM} is undecidable by reducing L_{Diag} to A_{TM}. Specifically, we show that if there were a (hypothetical) TM R

that decides A_{TM}, then R can be used to design a TM $\tilde{M}$ that decides L_{Diag}.

Since L_{Diag} is already known to be undecidable, it follows that A_{TM} is actually undecidable.

Towards this end, suppose (on contrary) that R decides A_{TM}. I.e.

$$R(\langle M, w \rangle) = \begin{cases} \text{Accept} & \text{if } M \text{ accepts } w. \\ \text{Reject} & \text{if } M \text{ rejects } w \\ & \text{or runs forever.} \end{cases}$$

It is easily seen now that $\tilde{M}$ as below decides L_{Diag}.

$\tilde{M}$:= "On input w,

Determine index i s.t. $w = w_i$.

Determine the TM $\langle M_i \rangle$.

Use R to decide whether M_i accepts w_i.

If M_i accepts w_i, accept.

If M_i does not accept w_i, reject."

Clearly,

$\tilde{M}$ accepts w if $w = w_i$, M_i accepts w_i

i.e. if $w_i \in L_{Diag}$.

$\tilde{M}$ rejects w if $w = w_i$, M_i does not accept w_i

i.e. if $w = w_i \notin L_{Diag}$.

Thus $\tilde{M}$ decides L_{Diag} (a contradiction). ∎

<u>Corollary</u> $\overline{A_{TM}}$ is not even T.R.

<u>Theorem</u> Halting Problem for TMs.

$$HALT_{TM} = \{ \langle M, w \rangle \mid M \text{ is a TM that halts on } w \}$$

is T.R. but undecidable.

<u>Proof</u> Clearly the following TM recognizes $HALT_{TM}$.

$M^* :=$ " On input $\langle M, w \rangle$
Simulate M on w.
If M halts (i.e. accepts or rejects), accept. "

$\langle M, w \rangle \in HALT_{TM} \Rightarrow M$ halts on w

$\Rightarrow M^*$ accepts $\langle M, w \rangle$.

$\langle M, w \rangle \notin HALT_{TM} \Rightarrow M$ runs forever on w

$\Rightarrow M^*$ runs forever on $\langle M, w \rangle$ ($\because M^*$ just simulates M).

This shows that indeed M^* recognizes $HALT_{TM}$ ▨.

Now $HALT_{TM}$ is undecidable because if there were a (hypothetical) decider R for $HALT_{TM}$, one can design a decider $\tilde{M}$ for A_{TM}. The latter is not possible since A_{TM} is already known to be undecidable.

Indeed $\tilde{M}$, on input $\langle M, w \rangle$ for A_{TM}, uses R to decide whether M halts on w.

If M does not halt on w, $\tilde{M}$ rejects.
If M halts on w, $\tilde{M}$ simulates M on w and accepts or rejects accordingly.

Clearly $\tilde{M}$ accepts $\langle M, w \rangle$ if M accepts w and rejects otherwise, and hence decides A_{TM}.

▨

Corollary $\overline{HALT_{TM}}$ is not even T.R.

Theorem (Non-) Emptiness Problem for TMs.

$$E_{TM} = \{ \langle M \rangle \mid M \text{ is a TM s.t. } L(M) \neq \phi \},$$

is T.R. but undecidable.

Proof Showing that E_{TM} is T.R. is left as a nice exercise :).

We show that E_{TM} is undecidable by reducing A_{TM} to it. Suppose (on contrary) that E_{TM} is decidable and a TM R decides it.

We will design a TM $\tilde{M}$ that decides A_{TM} (reaching a contradiction).

$\tilde{M}$:= "On input $\langle M, w \rangle$,

Construct a TM D that behaves as follows.

D := "On input x,

If $x \neq w$, reject.

If $x = w$, simulate

M on w and accept / reject / runforever accordingly."

By running R on $\langle D \rangle$, determine whether $L(D) \neq \phi$.

If $L(D) \neq \phi$, Accept.
If $L(D) = \phi$, Reject. "

We note that $\tilde{M}$ indeed decides A_{TM}.

$$\begin{aligned}
\langle M, w \rangle \in A_{TM} \quad &\Rightarrow \quad M \text{ accepts } w \\
&\Rightarrow \quad L(D) = \{w\} \neq \phi \\
&\Rightarrow \quad \tilde{M} \text{ accepts } \langle M, w \rangle.
\end{aligned}$$

$$\begin{aligned}
\langle M, w \rangle \notin A_{TM} \quad &\Rightarrow \quad M \text{ rejects / runs forever on } w \\
&\Rightarrow \quad L(D) = \phi \\
&\Rightarrow \quad \tilde{M} \text{ rejects } \langle M, w \rangle.
\end{aligned}$$

The main point is that from the perspective of the m/c D that is designed, all inputs other than w are rejected outright. Thus

$L(D) = \{w\}$ <u>or</u> $L(D) = \phi$

depending on whether

M accepts w or not. $\blacksquare$

Corollary $\overline{E_{TM}} = \{ \langle M \rangle \mid M \text{ is a TM s.t. } L(M) = \phi \}$

is not even T.R.

Exercise Using a proof similar as above show

that - ~~All_{TM} is T.R. but undecidable~~

- All_{TM} is not ~~even T.R~~ decidable. Here

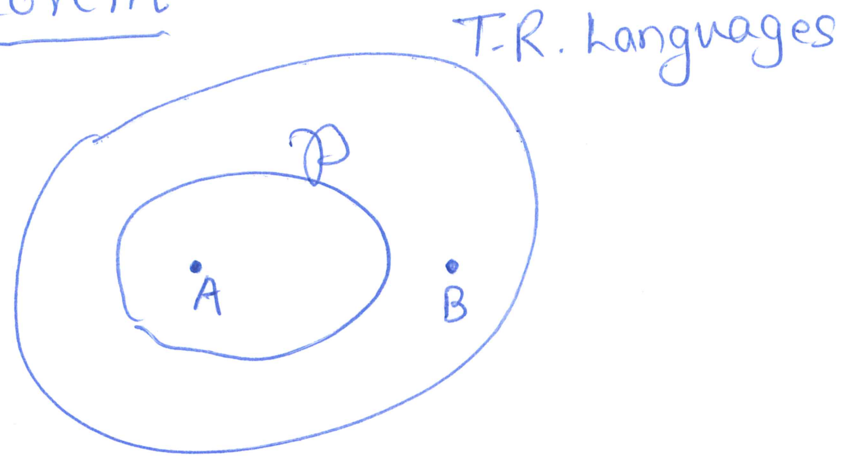
$All_{TM} = \{ \langle M \rangle \mid M \text{ is a TM s.t. } L(M) = \Sigma^* \}$.

Exercise Show that

$EQ_{TM} = \left\{ \langle M, M' \rangle \;\middle|\; \begin{array}{c} M, M' \text{ are TM,} \\ L(M) = L(M') \end{array} \right\}$ is not T.R.

Finally, we prove a rather general result known as Rice's Theorem. It states that any non-trivial "property" of the language $L(M)$ recognized by a TM M, given the description $\langle M \rangle$, is undecidable.

Rice's Theorem



Let $\mathcal{P}$ be a subclass of the class of T.R. languages. $\mathcal{P}$ is non-trivial if

- there is a lang. $A \in \mathcal{P}$ (i.e. $\mathcal{P}$ is non-empty)
- " $B \notin \mathcal{P}$ (i.e. $\mathcal{P}$ is not all the TR languages),

Then the following lang. is undecidable.

$$L_{\mathcal{P}} = \{ \langle M \rangle \mid M \text{ is a TM s.t. } L(M) \in \mathcal{P} \}.$$

<u>Note</u> The theorem, in one sweep, shows that all these languages are undecidable

$\{\langle M \rangle \mid L(M) \neq \phi\}$ $\quad \mathcal{P}$ = All T.R. languages except ϕ.

$\{\langle M \rangle \mid L(M) = \Sigma^*\}$ $\quad \mathcal{P} = \{\Sigma^*\}$.

$\{\langle M \rangle \mid L(M) \text{ is regular}\}$ $\quad \mathcal{P}$ = Class of regular langs.

$\{\langle M \rangle \mid L(M) \text{ is context free}\}$ $\quad \mathcal{P}$ = " c.f. languages.

Proof. W.l.o.g. we can assume that (the empty language) $\phi \notin \mathcal{P}$. Otherwise we could consider the property $\bar{\mathcal{P}}$, observe that $L_{\bar{\mathcal{P}}} = \overline{L_{\mathcal{P}}}$, and $L_{\bar{\mathcal{P}}}$ is undecidable iff $L_{\mathcal{P}}$ is.

So let's assume $\phi \notin \mathcal{P}$. Since $\mathcal{P}$ is non-trivial, there is some T.R. language $A \in \mathcal{P}$. Suppose a TM M_A recognizes the language A.

We now show that $L_{\mathcal{P}}$ is undecidable by reducing A_{TM} to it. Specifically,

Given input $\langle M, w \rangle$ for lang. A_{TM} $\xrightarrow{\text{"we" design}}$ A TM R such that

$\langle M, w \rangle \in A_{TM} \Rightarrow L(R) = A$

In particular $L(R) \in \mathcal{P}$.

$\langle M, w \rangle \notin A_{TM} \Rightarrow L(R) = \phi$.

In particular $L(R) \notin \mathcal{P}$.

Thus, if $L_{\mathcal{P}}$ were decidable, "one" could decide whether $L(R) \in \mathcal{P}$ or not

i.e. $L(R) = A$ or $L(R) = \phi$

i.e. $\langle M, w \rangle \in A_{TM}$ or $\langle M, w \rangle \notin A_{TM}$,

and thus decide A_{TM}, reaching a contradiction.

The m/c R is designed as follows:

R := " On input x,

Ignore x for now and first simulate M on w.

If M rejects w, reject.

If M runsforever on w, run forever.

Else M accepts w. In this case simulate M_A on x and Accept or Reject or Run forever accordingly."

Clearly if M rejects w or runs forever on w then R rejects every input x or runs forever on every input x.

In either case $L(R) = \phi$. Thus $\langle M, w \rangle \notin A_{TM} \Rightarrow L(R) = \phi$ as needed.

On the other hand

if M accepts w then

Behavior of R on every input x is same as M_A on that input since R just simulates M_A on x.

Hence $L(R) = A$, the lang. $\frac{\text{accepted}}{\text{recognized}}$ by the machine M_A.

Thus $\langle M, w \rangle \in A_{TM} \Rightarrow L(R) = A$ as needed. ▨

Remark In the proof "we design" or "one could decide" are to be interpreted as "a TM can design" or "a TM could decide" respectively.