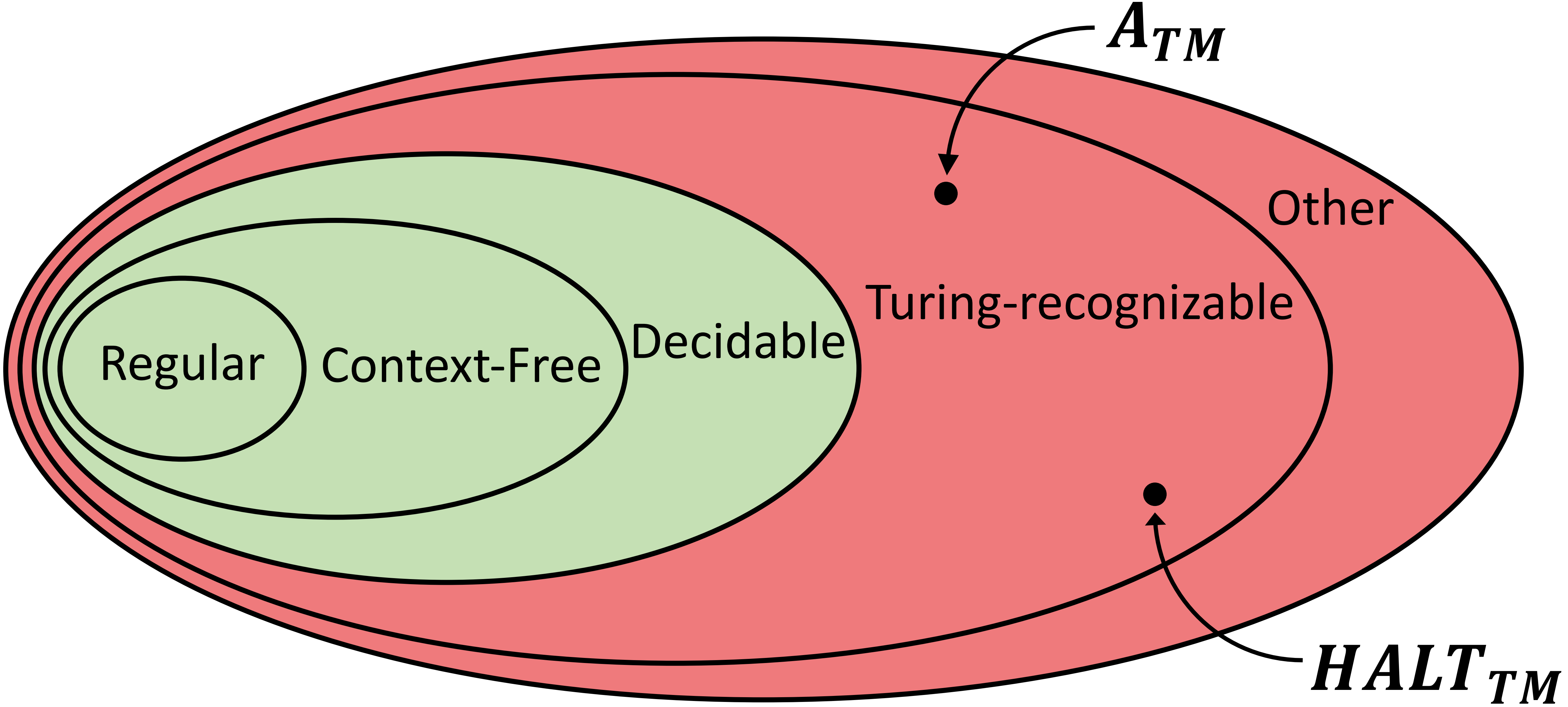
Undecidability

CSCI 338

Computability Hierarchy



Halting Problem

Claim: $HALT_{TM} = \{\langle M, \omega \rangle : M$ is a TM and M halts on ω} is undecidable.

Proof: Suppose $HALT_{TM}$ is decidable and let TM H be its decider.

Build a TM S that decides A_{TM}:

S = on input $\langle M, \omega \rangle$

1. Run H on $\langle M, \omega \rangle$.
2. If H rejects, reject (i.e. M does not halt on ω).
3. If H accepts, run M on ω until it halts.
4. If M accepts, accept. If M rejects, reject.

If M accepts ω, S accepts. If M does not accept ω, S rejects. Thus, S is a decider for A_{TM}, which is a contradiction, so $HALT_{TM}$ is undecidable.

Undecidability Proof Blueprint

Claim: $\boldsymbol{New\_Problem}$ is undecidable.

Proof: Suppose $\boldsymbol{New\_Problem}$ is decidable and let H be its decider.

Build a TM S that decides $\boldsymbol{Old\_Problem}$:

S = on input $\langle ? \rangle$

1. …

⋮

Input depends on the input to $\boldsymbol{Old\_Problem}$.

S is a decider for $\boldsymbol{Old\_Problem}$, which is a contradiction.

$\therefore$ $\boldsymbol{New\_Problem}$ is undecidable.

$\boldsymbol{New\_Problem}$: Problem we are trying to show is undecidable.
$\boldsymbol{Old\_Problem}$: Problem we already know to be undecidable.

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M \text{ is a TM and } L(M) = \emptyset\}$ is undecidable.

Proof:

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

1.

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

1.

$$\boxed{\text{To show } E_{TM} \text{ is undecidable, use it to decide } A_{TM}.}$$

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

1.

To show E_{TM} is undecidable, use it to decide A_{TM}.

$$L(M) \neq \emptyset \quad \Longleftrightarrow \quad N \text{ accepts } \omega$$

Question: Does the government have aliens?

Question: Does the government have aliens?

Frank

Can always perfectly answer the question:

“Does a recipe use eggs?”

Question: Does the government have aliens?

Bread Recipe:

1. Add flour, water, sugar, yeast, salt.
2. If the government has aliens, add an egg.
3. Mix, bake, eat.

Frank

Can always perfectly answer the question:
“Does a recipe use eggs?”

Question: Does the government have aliens?

<u>Bread Recipe</u>:

1. Add flour, water, sugar, yeast, salt.
2. If the government has aliens, add an egg.
3. Mix, bake, eat.

Frank

Can always perfectly answer the question:

"Does a recipe use eggs?"

Question: Does the government have aliens?

Bread Recipe:

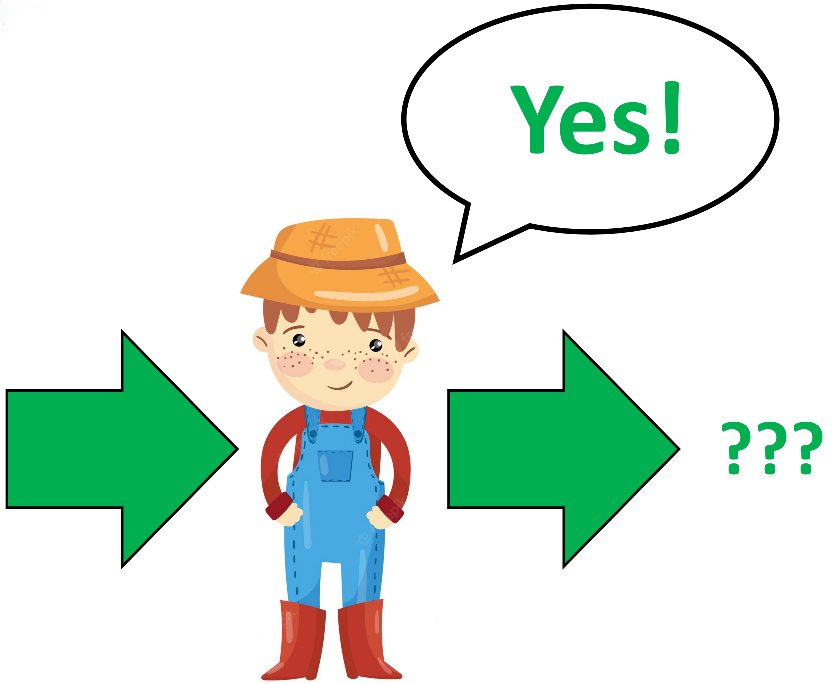
1. Add flour, water, sugar, yeast, salt.
2. If the government has aliens, add an egg.
3. Mix, bake, eat.

Frank

Can always perfectly answer the question:
"Does a recipe use eggs?"

Question: Does the government have aliens?

Bread Recipe:
1. Add flour, water, sugar, yeast, salt.
2. If the government has aliens, add an egg.
3. Mix, bake, eat.



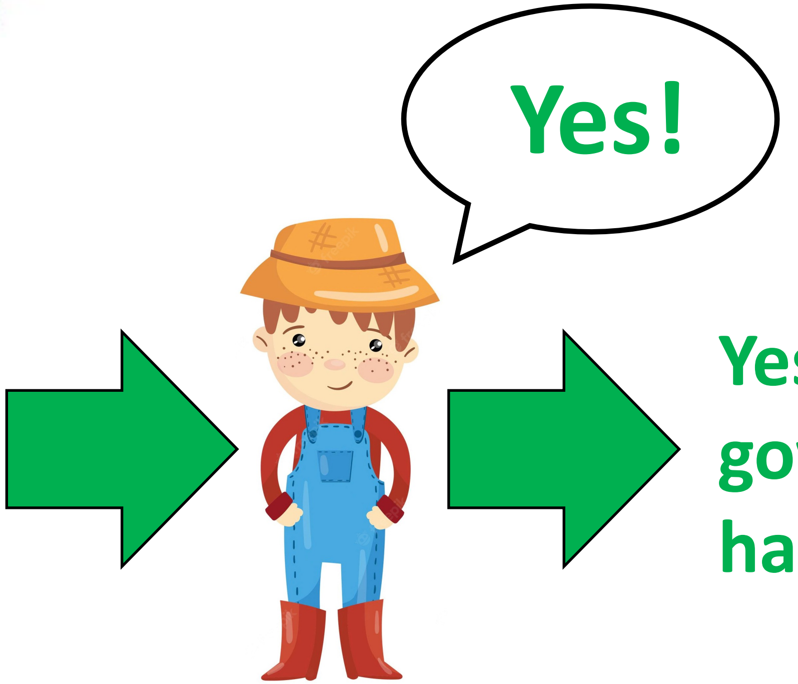
Frank

Can always perfectly answer the question:
"Does a recipe use eggs?"

Question: Does the government have aliens?

Bread Recipe:

1. Add flour, water, sugar, yeast, salt.
2. If the government has aliens, add an egg.
3. Mix, bake, eat.



Yes, the government has aliens.

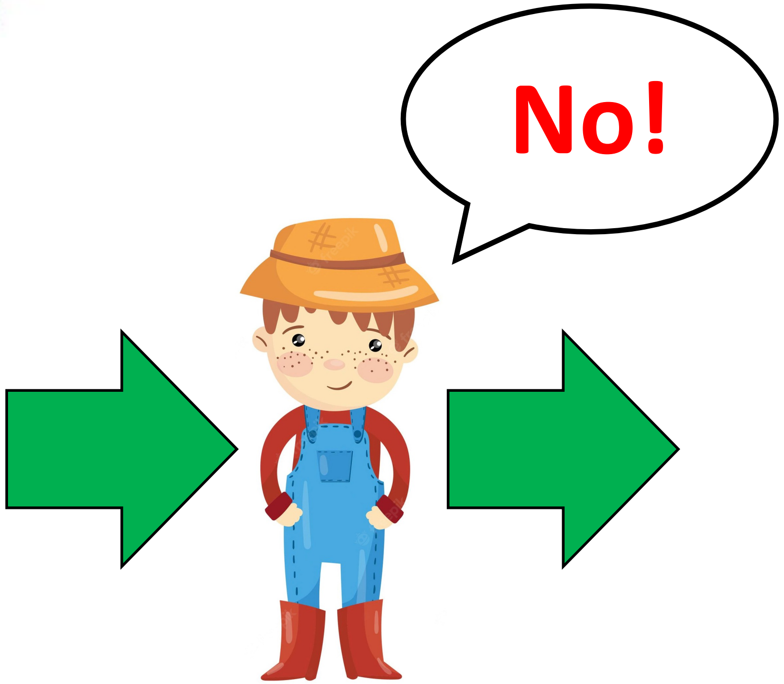
Frank

Can always perfectly answer the question:
"Does a recipe use eggs?"

Question: Does the government have aliens?

<u>Bread Recipe</u>:

1. Add flour, water, sugar, yeast, salt.
2. If the government has aliens, add an egg.
3. Mix, bake, eat.



Frank

Can always perfectly answer the question:
“Does a recipe use eggs?”

Question: Does the government have aliens?

<u>Bread Recipe</u>:

1. Add flour, water, sugar, yeast, salt.
2. If the government has aliens, add an egg.
3. Mix, bake, eat.

No, the government doesn't have aliens.

Frank

Can always perfectly answer the question:
"Does a recipe use eggs?"

Does the government have aliens?:

Does the government have aliens?:

1. Write down the following recipe:

<u>Bread Recipe</u>:

1. Add flour, water, sugar, yeast, salt.
2. If the government has aliens, add an egg.
3. Mix, bake, eat.

Does the government have aliens?:

1. Write down the following recipe:

   > <u>Bread Recipe</u>:
   > 1. Add flour, water, sugar, yeast, salt.
   > 2. If the government has aliens, add an egg.
   > 3. Mix, bake, eat.

2. ???

Does the government have aliens?:

1. Write down the following recipe:

   <u>Bread Recipe</u>:
   1. Add flour, water, sugar, yeast, salt.
   2. If the government has aliens, add an egg.
   3. Mix, bake, eat.

2. Ask Frank if the recipe uses eggs.

Does the government have aliens?:

1. Write down the following recipe:

> <u>Bread Recipe</u>:
> 1. Add flour, water, sugar, yeast, salt.
> 2. If the government has aliens, add an egg.
> 3. Mix, bake, eat.

2. Ask Frank if the recipe uses eggs.
3. If Frank says yes, <u>???</u>

Does the government have aliens?:

1. Write down the following recipe:

   > <u>Bread Recipe</u>:
   > 1. Add flour, water, sugar, yeast, salt.
   > 2. If the government has aliens, add an egg.
   > 3. Mix, bake, eat.

2. Ask Frank if the recipe uses eggs.
3. If Frank says yes, <u>yes, the government has aliens</u>. If Frank says no, <u>no, the government does not have aliens</u>.

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Frank

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Frank

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

The question I care about

E_{TM}

Claim: $E_{TM} = \{\langle M\rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Frank

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega\rangle$

1. Construct TM M_2 on input $\langle x\rangle$:

$\boldsymbol{N}$ **accepts** $\boldsymbol{\omega} \Leftrightarrow \boldsymbol{L(M_2) \neq \emptyset}$

The question I care about

The recipe that forces Frank to answer the question I care about.

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Frank

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

1. Construct TM M_2 on input $\langle x \rangle$:

   $\boldsymbol{N}$ **accepts** $\boldsymbol{\omega} \Leftrightarrow \boldsymbol{L(M_2) \neq \emptyset}$

2. ??

The question I care about

The recipe that forces Frank to answer the question I care about.

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Frank

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

1. Construct TM M_2 on input $\langle x \rangle$:

   $\boldsymbol{N}$ **accepts** $\boldsymbol{\omega} \Leftrightarrow \boldsymbol{L(M_2) \neq \emptyset}$

2. Run H on $\langle M_2 \rangle$.
3. ??

The question I care about

The recipe that forces Frank to answer the question I care about.

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M \text{ is a TM and } L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Frank

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

1. Construct TM M_2 on input $\langle x \rangle$:

   $\boldsymbol{N}$ **accepts** $\boldsymbol{\omega} \Leftrightarrow \boldsymbol{L(M_2) \neq \emptyset}$

2. Run H on $\langle M_2 \rangle$.
3. If H accepts, <u>reject</u>. If H rejects, <u>accept</u>.

The question I care about

The recipe that forces Frank to answer the question I care about.

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

1. Construct TM M_2 on input $\langle x \rangle$:
2. Run H on $\langle M_2 \rangle$.
3. If H accepts, <u>reject</u>. If H rejects, <u>accept</u>.

$\boldsymbol{N}$ **accepts** $\boldsymbol{\omega}$

$\Updownarrow$

$\boldsymbol{L(M_2) \neq \emptyset}$

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M \text{ is a TM and } L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

1. Construct TM M_2 on input $\langle x \rangle$:
    1. Run N on ω and accept if N does.
2. Run H on $\langle M_2 \rangle$.
3. If H accepts, reject. If H rejects, accept.

N accepts ω

$$\Updownarrow$$

$$\boldsymbol{L(M_2) \neq \emptyset}$$

E_{TM}

Claim: $E_{TM} = \{\langle M \rangle : M \text{ is a TM and } L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega \rangle$

1. Construct TM M_2 on input $\langle x \rangle$:
   1. Run N on ω and <u>accept</u> if N does.
2. Run H on $\langle M_2 \rangle$.
3. If H accepts, <u>reject</u>. If H rejects, <u>accept</u>.

N accepts ω

$\Updownarrow$

$\boldsymbol{L(M_2) \neq \emptyset}$

$\boldsymbol{L(M_2) = ??}$

E_{TM}

Claim: $E_{TM} = \{\langle M\rangle : M$ is a TM and $L(M) = \emptyset\}$ is undecidable.

Proof: Suppose E_{TM} is decidable and let TM H be its decider.

Build a TM S that decides A_{TM}:

S = on input $\langle N, \omega\rangle$

1. Construct TM M_2 on input $\langle x\rangle$:
   1. Run N on ω and accept if N does.
2. Run H on $\langle M_2\rangle$.
3. If H accepts, reject. If H rejects, accept.

$\boldsymbol{N}$ **accepts** $\boldsymbol{\omega}$

$\Updownarrow$

$\boldsymbol{L(M_2) \neq \emptyset}$

$\boldsymbol{L(M_2) = \Sigma^*}$ **or** $\boldsymbol{\emptyset}$