

CSE 2001:
Introduction to Theory of Computation
Summer 2013

Week 10: Decidability Part 2

Yves Lespérance

Course page: <http://www.cse.yorku.ca/course/2001>

Slides are mostly taken from Suprakash Datta's for Winter 2013; some slides are adapted from Wim van Dam's slides www.cs.berkeley.edu/~vandam/CS172/ (retrieved earlier)

13-07-27

CSE 2001, Summer 2013

1

Next

Towards undecidability:

- **The Halting Problem**
- **Countable and uncountable infinities**
- **Diagonalization arguments**

13-07-27

CSE 2001, Summer 2013

2

The Halting Problem

The existence of the universal TM U shows that $A_{TM} = \{ \langle M, w \rangle \mid M \text{ is a TM that accepts } w \}$ is TM-recognizable, but can we also *decide* it?

The problem lies with the cases when M does not halt on w . In short: the halting problem.

We will see that this is an insurmountable problem: in general one cannot decide if a TM will halt on w or not, hence A_{TM} is undecidable.

13-07-27

CSE 2001, Summer 2013

3

Counting arguments

- We need tools to reason about undecidability.
- The basic argument is that there are more languages than Turing machines and so there are languages than Turing machines. Thus some languages cannot be decidable

13-07-27

CSE 2001, Summer 2013

4

Baby steps

- What is counting?
 - Labeling with integers
 - Correspondence with integers
- Let us review basic properties of functions

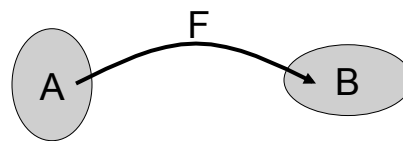
13-07-27

CSE 2001, Summer 2013

5

Mappings and Functions

The function $F:A \rightarrow B$
maps one set A to
another set B :



F is one-to-one (injective) if every $x \in A$ has a unique image $F(x)$: If $F(x)=F(y)$ then $x=y$.

F is onto (surjective) if every $z \in B$ is 'hit' by F : If $z \in B$ then there is an $x \in A$ such that $F(x)=z$.

F is a correspondence (bijection) between A and B if it is both one-to-one and onto.

13-07-27

CSE 2001, Summer 2013

6

Cardinality

A set S has k elements if and only if there exists a bijection between S and $\{1,2,\dots,k\}$.

S and $\{1,\dots,k\}$ have the same cardinality.

If there is a surjection possible from $\{1,\dots,n\}$ to S , then $n \geq |S|$.

We can generalize this way of comparing the sizes of sets to infinite ones.

13-07-27

CSE 2001, Summer 2013

7

How Many Languages?

For $\Sigma=\{0,1\}$, there are 2^k words of length k .
Hence, there are $2^{(2^k)}$ languages $L \subseteq \Sigma^k$.



Proof: L has two options for every word $\in \Sigma^k$;
 L can be represented by a string $\in \{0,1\}^{(2^k)}$.

That's a lot, but finite.

There are infinitely many languages $\subseteq \Sigma^*$.
But we can say more than that...

Georg Cantor defined a way of comparing infinities.

13-07-27

CSE 2001, Summer 2013

8

Countably Infinite Sets

A set S is infinite if there exists a surjective function $F: S \rightarrow \mathbb{N}$.

“The set \mathbb{N} has no more elements than S .”

A set S is countable if there exists a surjective function $F: \mathbb{N} \rightarrow S$

“The set S has not more elements than \mathbb{N} .”

A set S is countably infinite if there exists a bijective function $F: \mathbb{N} \rightarrow S$.

“The sets \mathbb{N} and S are of equal size.”

13-07-27

CSE 2001, Summer 2013

9

Counterintuitive facts

- Cardinality of even integers
 - Bijection $i \leftrightarrow 2i$
 - A proper subset of \mathbb{N} has the same cardinality as \mathbb{N} !
 - Same holds for odd integers
- What about pairs of natural numbers?
 - Bijection from \mathbb{N} to $\mathbb{N} \times \mathbb{N}$!!
 - Cantor’s idea: count by diagonals
 - Implies set of rational numbers is countable

13-07-27

CSE 2001, Summer 2013

10

Counterintuitive facts - 2

- Note that the ordering of \mathbb{Q} is not in increasing order or decreasing order of value.
- In proofs, you CANNOT assume that an ordering has to be in increasing or decreasing order.
- So cannot use ideas like “between any two real numbers x, y , there exists a real number $0.5(x+y)$ ” to prove uncountability.

13-07-27

CSE 2001, Summer 2013

11

More Countably Infinite Sets

One can make bijections between \mathbb{N} and

1. $\{a\}^*$: $i \leftrightarrow a^i$

2. Integers (\mathbb{Z}):

| | | | | | | | | | | |
|---|----|----|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
| 0 | +1 | -1 | +2 | -2 | +3 | -3 | +4 | -4 | +5 | -5 |

13-07-27

CSE 2001, Summer 2013

12

Countable sets in language theory

- Σ^* is countable – finitely many strings of length k . Order them lexicographically.
- Set of all Turing machines countable – every TM can be encoded as a string over some Σ .

13-07-27

CSE 2001, Summer 2013

13

Summary

A set S is countably infinite if there exists a bijection between $\{0,1,2,\dots\}$ and S .

Intuitively: A set S is countable, if you can make a List (numbering) s_1, s_2, \dots of all the elements of S .

The sets \mathbb{Q} , $\{0,1\}^*$ are countably infinite.

Example for $\{0,1\}^*$: the lexicographical ordering:

$$\{0,1\}^* = \{\epsilon, 0, 1, 00, 01, 10, 11, 000, \dots\}$$

Q: Are there bigger sets?

13-07-27

CSE 2001, Summer 2013

14

Next

- Chapter 4.2:
 - Uncountable Set of Languages
 - Unrecognizable Languages
 - Halting Problem is Undecidable
 - Non-Halting is not TM-Recognizable

13-07-27

CSE 2001, Summer 2013

15

Uncountable Sets

There are infinite sets that are not countable.
Typical examples are \mathbb{R} , $\mathcal{P}(\mathbb{N})$ and $\mathcal{P}(\{0,1\}^*)$

We prove this by a diagonalization argument.
In short, if S is countable, then you can make a list s_1, s_2, \dots of all elements of S .

Diagonalization shows that given such a list, there will always be an element x of S that does not occur in s_1, s_2, \dots

13-07-27

CSE 2001, Summer 2013

16

Uncountability of $\mathcal{P}(\mathbb{N})$

The set $\mathcal{P}(\mathbb{N})$ contains all the subsets of $\{1,2,\dots\}$. Each subset $X \subseteq \mathbb{N}$ can be identified by an infinite string of bits $x_1x_2\dots$ such that $x_j=1$ iff $j \in X$.

There is a bijection between $\mathcal{P}(\mathbb{N})$ and $\{0,1\}^{\mathbb{N}}$.

Proof by contradiction: Assume $\mathcal{P}(\mathbb{N})$ countable. Hence there must exist a surjection F from \mathbb{N} to the set of infinite bit strings.

“There is a list of *all* infinite bit strings.”

13-07-27

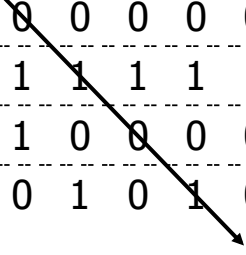
CSE 2001, Summer 2013

17

Diagonalization

Try to list all possible infinite bit strings:

| | | | | | | |
|---|---|---|---|---|---|-----|
| 0 | 0 | 0 | 0 | 0 | 0 | ... |
| 1 | 1 | 1 | 1 | 1 | 1 | ... |
| 2 | 1 | 0 | 0 | 0 | 0 | ... |
| 3 | 0 | 1 | 0 | 1 | 0 | ... |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |



Look at the bit string on the diagonal of this table: 0101... The negation of this string (“1010...”) does not appear in the

table.

CSE 2001, Summer 2013

18

No Surjection $\mathbb{N} \rightarrow \{0,1\}^{\mathbb{N}}$

Let F be a function $\mathbb{N} \rightarrow \{0,1\}^{\mathbb{N}}$.
 $F(1), F(2), \dots$ are all infinite bit strings.

Define the infinite string $Y = Y_1 Y_2 \dots$ by
 $Y_j = \text{NOT}(j\text{-th bit of } F(j))$

On the one hand $Y \in \{0,1\}^{\mathbb{N}}$, but on the other hand: for every $j \in \mathbb{N}$ we know that $F(j) \neq Y$ because $F(j)$ and Y differ in the j -th bit.

F cannot be a surjection: $\{0,1\}^{\mathbb{N}}$ is uncountable.

13-07-27

CSE 2001, Summer 2013

19

Generalization

- We proved that $\mathcal{P}(\{0,1\}^*)$ is uncountably infinite.
- Can be generalized to $\mathcal{P}(\Sigma^*)$ for any finite Σ .

13-07-27

CSE 2001, Summer 2013

20

R is uncountable

- Similar diagonalization proof. We will prove $[0,1)$ uncountable
- Let F be a function $\mathbb{N} \rightarrow \mathbb{R}$
 $F(1), F(2), \dots$ are all infinite digit strings (padded with zeroes if required).
- Define the infinite string of digits $Y = Y_1 Y_2 \dots$ by
$$Y_j = \begin{cases} F(i)_i + 1 & \text{if } F(i)_i < 8 \\ 7 & \text{if } F(i)_i \geq 8 \end{cases}$$

Q: Where does this proof fail on \mathbb{N} ?

13-07-27

CSE 2001, Summer 2013

21

Other infinities

- We proved $2^{\mathbb{N}}$ uncountable. We can show that this set has the same cardinality as $\mathcal{P}(\mathbb{N})$ and \mathbb{R} .
- What if we take $\mathcal{P}(\mathbb{R})$?
- Can we build bigger and bigger infinities this way?
- Euler: Continuum hypothesis – YES!

13-07-27

CSE 2001, Summer 2013

22

Uncountability

We just showed that there it is impossible to have a surjection from \mathbb{N} to the set $\{0,1\}^{\mathbb{N}}$.

What does this have to do with Turing machine computability?

13-07-27

CSE 2001, Summer 2013

23

Counting TMs

Observation: Every TM has a finite description; there is only a countable number of different TMs. (A description $\langle M \rangle$ can consist of a finite string of bits, and the set $\{0,1\}^*$ is countable.)

Our definition of Turing recognizable languages is a mapping between the set of TMs $\{M_1, M_2, \dots\}$ and the set of languages $\{L(M_1), L(M_2), \dots\} \subseteq \mathcal{P}(\Sigma^*)$.

Question: How many languages are there?

13-07-27

CSE 2001, Summer 2013

24

Counting Languages

There are uncountably many different languages over the alphabet $\Sigma=\{0,1\}$ (the languages $L\subseteq\{0,1\}^*$). With the lexicographical ordering $\varepsilon,0,1,00,01,\dots$ of Σ^* , every L coincides with an infinite bit string via its characteristic sequence χ_L .

Example for $L=\{0,00,01,000,001,\dots\}$ with $\chi_L = 0101100\dots$

| Σ^* | ε | 0 | 1 | 00 | 01 | 10 | 11 | 000 | 001 | 010 | ... |
|------------|---------------|---|---|----|----|----|----|-----|-----|-----|-----|
| L | | X | | X | X | | | X | X | X | ... |
| χ_L | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | ... |

13-07-27

CSE 2001, Summer 2013

25

Counting TMs and Languages

There is a bijection between the set of languages over the alphabet $\Sigma=\{0,1\}$ and the uncountable set of infinite bit strings $\{0,1\}^{\mathbb{N}}$.

- There are uncountable many different languages $L\subseteq\{0,1\}^*$.
- Hence there is no surjection possible from the countable set of TMs to the set of languages. Specifically, the mapping $L(M)$ is not surjective.

Conclusion: There are languages that are not Turing-recognizable. (A lot of them.)

13-07-27

CSE 2001, Summer 2013

26

Is This Really Interesting?

We now know that there are languages that are not Turing recognizable, but we do not know what kind of languages are non-TM-recognizable.

Are there interesting languages for which we can prove that there is no Turing machine that recognizes it?

13-07-27

CSE 2001, Summer 2013

27

Proving Undecidability (1)

Recall the language

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

Proof that A_{TM} is not TM-decidable (Thm. 4.11)

(Contradiction) Assume that TM G decides A_{TM} :

$$G\langle M, w \rangle = \begin{cases} \text{"accept"} & \text{if } M \text{ accepts } w \\ \text{"reject"} & \text{if } M \text{ does not accept } w \end{cases}$$

From G we construct a new TM D that will get us into trouble...

13-07-27

CSE 2001, Summer 2013

28

Proving Undecidability (2)

The TM D works as follows on input $\langle M \rangle$ (a TM):

1) Run G on $\langle M, \langle M \rangle \rangle$

2) Disagree with the answer of G

(The TM D always halts because G always halts.)

In short: $D\langle M \rangle = \begin{cases} \text{"accept"} & \text{if } G \text{ rejects } \langle M, \langle M \rangle \rangle \\ \text{"reject"} & \text{if } G \text{ accepts } \langle M, \langle M \rangle \rangle \end{cases}$

Hence: $D\langle M \rangle = \begin{cases} \text{"accept"} & \text{if } M \text{ does not accept } \langle M \rangle \\ \text{"reject"} & \text{if } M \text{ does accept } \langle M \rangle \end{cases}$

Now run D on $\langle D \rangle$ (“on itself”)...

13-07-27

CSE 2001, Summer 2013

29

Proving Undecidability (3)

Result: $D\langle D \rangle = \begin{cases} \text{"accept"} & \text{if } D \text{ does not accept } \langle D \rangle \\ \text{"reject"} & \text{if } D \text{ does accept } \langle D \rangle \end{cases}$

This does not make sense: D only accepts if it rejects, and vice versa.

(Note again that D always halts.)

Contradiction: **A_{TM} is not TM-decidable.**

This proof used diagonalization implicitly...

13-07-27

CSE 2001, Summer 2013

30

Review of Proof (1)

| | $\langle M_1 \rangle$ | $\langle M_2 \rangle$ | $\langle M_3 \rangle$ | $\langle M_4 \rangle$ | ... |
|----------|-----------------------|-----------------------|-----------------------|-----------------------|----------|
| M_1 | accept | | accept | | |
| M_2 | accept | accept | accept | accept | |
| M_3 | | | | | ... |
| M_4 | accept | accept | | | |
| \vdots | | | \vdots | | \ddots |

'Acceptance behavior' of M_i on $\langle M_j \rangle$

13-07-27

CSE 2001, Summer 2013

31

Review of Proof (2)

| | $\langle M_1 \rangle$ | $\langle M_2 \rangle$ | $\langle M_3 \rangle$ | $\langle M_4 \rangle$ | ... |
|----------|-----------------------|-----------------------|-----------------------|-----------------------|----------|
| M_1 | accept | reject | accept | reject | |
| M_2 | accept | accept | accept | accept | |
| M_3 | reject | reject | reject | reject | ... |
| M_4 | accept | accept | reject | reject | |
| \vdots | | | \vdots | | \ddots |

'Deciding behavior' of G on $\langle M_i, \langle M_j \rangle \rangle$

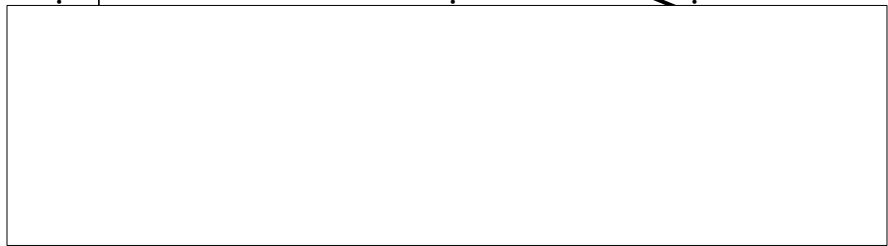
13-07-27

CSE 2001, Summer 2013

32

Review of Proof (3)

| | $\langle M_1 \rangle$ | $\langle M_2 \rangle$ | $\langle M_3 \rangle$ | $\langle M_4 \rangle$ | ... | $\langle D \rangle$ | ... |
|-------|-----------------------|-----------------------|-----------------------|-----------------------|-----|---------------------|-----|
| M_1 | accept | reject | accept | reject | | | |
| M_2 | accept | accept | accept | accept | | | |
| M_3 | reject | reject | reject | reject | ... | | |
| M_4 | accept | accept | reject | reject | | | |



Review of Proof (4)

| | $\langle M_1 \rangle$ | $\langle M_2 \rangle$ | $\langle M_3 \rangle$ | $\langle M_4 \rangle$ | ... | $\langle D \rangle$ | ... |
|----------|-----------------------|-----------------------|-----------------------|-----------------------|-----|---------------------|-----|
| M_1 | accept | reject | accept | reject | | | |
| M_2 | accept | accept | accept | accept | | | |
| M_3 | reject | reject | reject | reject | ... | | |
| M_4 | accept | accept | reject | reject | | | |
| \vdots | | | \vdots | | | \vdots | |
| D | reject | reject | accept | accept | ... | ? | |
| \vdots | | | | | | | |

Contradiction for D on input $\langle D \rangle$.

Another View of the Problem

The “**Self-referential paradox**” occurs when we force the TM D to disagree with itself.

On the one hand, D knows what it is going to do on input $\langle D \rangle$, but then it decides to do something else instead.

“You cannot know for sure what you will do in the future, because then you could decide to change your actions and create a paradox.”

13-07-27

CSE 2001, Summer 2013

35

Self-Reference in Math

The diagonalization method implements the self-reference paradox in a mathematical way.

In logic this approach is often used to prove that certain things are impossible.

Kurt Gödel gave a mathematical equivalent of “This sentence is not true.”

Old puzzle: In a town, there is a barber who shaves all those who do not shave themselves.

Who shaves the barber ?

13-07-27

CSE 2001, Summer 2013

36

Self-Reference in CSE

What happens if a computer program M tries to answer questions about itself $\langle M \rangle$?

Sometimes this is perfectly okay:

- How big is $\langle M \rangle$?
- Is $\langle M \rangle$ a proper TM?

Other questions lead to paradoxes:

- Does $\langle M \rangle$ halt or not?
- Is there a smaller program M' that is equivalent?

13-07-27

CSE 2001, Summer 2013

37

TM-Unrecognizable

A_{TM} is not TM-decidable, but it is TM-recognizable. What about a language that is not recognizable?

Theorem 4.22: If a language A is recognizable and its complement \bar{A} is recognizable, then A is Turing machine decidable.

Proof: Run the recognizing TMs for A and \bar{A} in parallel on input x . Wait for one of the TMs to accept. If the TM for A accepted: “accept x ”; if the TM for \bar{A} accepted: “reject x ”.

13-07-27

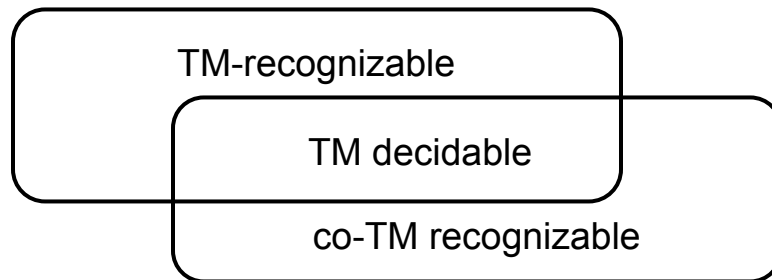
CSE 2001, Summer 2013

38

\bar{A}_{TM} is not TM-Recognizable

By the previous theorem it follows that \bar{A}_{TM} cannot be TM-recognizable, because this would imply that A_{TM} is TM decidable (Corollary 4.23).

We call languages like \bar{A}_{TM} co-TM recognizable.



13-07-27

CSE 2001, Summer 2013

39

Things that TMs Cannot Do:

The following languages are also unrecognizable:

$$E_{TM} = \{ \langle G \rangle \mid G \text{ is a TM with } L(G) = \emptyset \}$$

$$EQ_{TM} = \{ \langle G, H \rangle \mid G \text{ and } H \text{ are TMs} \\ \text{with } L(G) = L(H) \}$$

To be precise:

- E_{TM} is co-TM recognizable
- EQ_{TM} is not even co-Turing recognizable

How can we prove these facts?

13-07-27

CSE 2001, Summer 2013

40

Next: reducibility

- We still need to *prove* that the Halting problem is undecidable.
- Do more examples of undecidable problems.
- Try to get a general technique for proving undecidability.