Time Complexity

- We use a multitape Turing machine
- We count the number of steps until a string is accepted
- We use the O(k) notation

1

Example: $L = \{a^n b^n : n \geq 0\}$

Algorithm to accept a string w :

- Use a two-tape Turing machine
- Copy the a on the second tape
- Compare the a and b

2

$L = \{a^n b^n : n \geq 0\}$

Time needed:

- Copy the a on the second tape $O(|w|)$
- Compare the a and b $O(|w|)$

Total time: $O(|w|)$

3

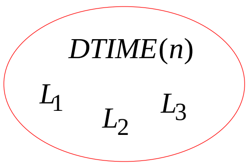
$L = \{a^n b^n : n \geq 0\}$

For string of length n

time needed for acceptance: $O(n)$

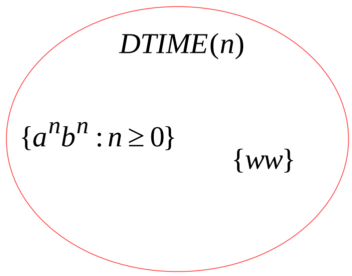
4

Language class: $DTIME(n)$



A Deterministic Turing Machine accepts each string of length n in time $O(n)$

5



6

In a similar way we define the class

$$DTIME(T(n))$$

for any time function: $T(n)$

Examples: $DTIME(n^2), DTIME(n^3), \ldots$

7

Example: The membership problem for context free languages

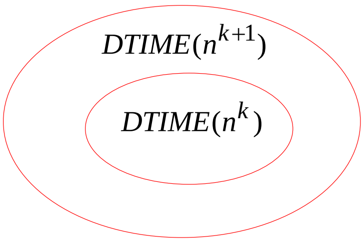
$$L = \{w : w \text{ is generated by grammar } G\}$$

$L \in DTIME(n^3)$ (CYK - algorithm)

Polynomial time

8

Theorem: $DTIME(n^{k+1}) \subset DTIME(n^k)$



9

Polynomial time algorithms: $DTIME(n^k)$

Represent tractable algorithms:
For small k we can compute the result fast

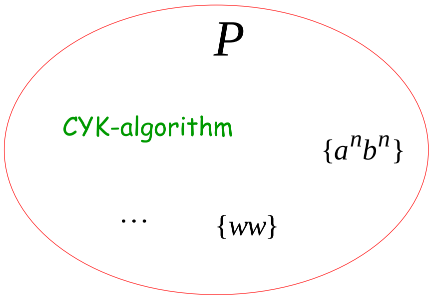
10

The class P

$P = \cup DTIME(n^k)$ for all k

- Polynomial time
- All tractable problems

11



12

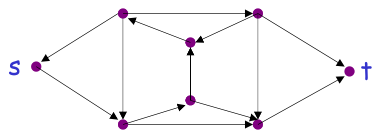
Exponential time algorithms: $DTIME(2^n)$

Represent intractable algorithms:

Some problem instances
may take centuries to solve

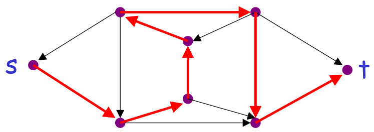
13

Example: the Hamiltonian Problem



Question: is there a Hamiltonian path
from s to t?

14



YES!

15

A solution: search exhaustively all paths

L = {<G,s,t>: there is a Hamiltonian path
in G from s to t}

$$L \in DTIME(n!) \approx DTIME(2^n)$$

Exponential time

Intractable problem

16

Example: The Satisfiability Problem

Boolean expressions in
Conjunctive Normal Form:

$$t_1 \wedge t_2 \wedge t_3 \wedge \cdots \wedge t_k$$

$$t_i = x_1 \vee \bar{x}_2 \vee x_3 \vee \cdots \vee \bar{x}_p$$

Variables

Question: is expression satisfiable?

17

Example: $(\bar{x}_1 \vee x_2) \wedge (x_1 \vee x_3)$

Satisfiable: $x_1 = 0, \; x_2 = 1, \; x_3 = 1$

$$(\bar{x}_1 \vee x_2) \wedge (x_1 \vee x_3) = 1$$

18

Example: $(x_1 \vee x_2) \wedge \bar{x}_1 \wedge \bar{x}_2$

Not satisfiable

$L = \{w : \text{expression } w \text{ is satisfiable}\}$

For n variables: $L \in DTIME(2^n)$

exponential

Algorithm:

search exhaustively all the possible binary values of the variables

Non-Determinism

Language class: $NTIME(n)$

$NTIME(n)$

L_1 L_2 L_3

A Non-Deterministic Turing Machine accepts each string of length n in time $O(n)$

Example: $L = \{ww\}$

Non-Deterministic Algorithm to accept a string ww :

- Use a two-tape Turing machine
- Guess the middle of the string and copy w on the second tape
- Compare the two tapes

$L = \{ww\}$

Time needed:

- Use a two-tape Turing machine
- Guess the middle of the string and copy w on the second tape $O(|w|)$
- Compare the two tapes $O(|w|)$

Total time: $O(|w|)$

$NTIME(n)$

$L = \{ww\}$

In a similar way we define the class

$$NTIME(T(n))$$

for any time function: $T(n)$

Examples: $NTIME(n^2),\ NTIME(n^3), \ldots$

25

Non-Deterministic Polynomial time algorithms:

$$L \in NTIME(n^k)$$

26

The class NP

$$NP = \cup NTIME(n^k) \quad \text{for all } k$$

Non-Deterministic Polynomial time

27

Example: The satisfiability problem

$$L = \{w : \text{expression } w \text{ is satisfiable}\}$$

Non-Deterministic algorithm:

- Guess an assignment of the variables
- Check if this is a satisfying assignment

28

$$L = \{w : \text{expression } w \text{ is satisfiable}\}$$

Time for n variables:

- Guess an assignment of the variables $O(n)$
- Check if this is a satisfying assignment $O(n)$

Total time: $O(n)$

29

$$L = \{w : \text{expression } w \text{ is satisfiable}\}$$

$$L \in NP$$

The satisfiability problem is an NP - Problem

30

Observation:

$$P \subseteq NP$$

Deterministic Polynomial

Non-Deterministic Polynomial

31

Open Problem: $P = NP$?

WE DO NOT KNOW THE ANSWER

32

Open Problem: $P = NP$?

Example: Does the Satisfiability problem have a polynomial time deterministic algorithm?

WE DO NOT KNOW THE ANSWER

33