

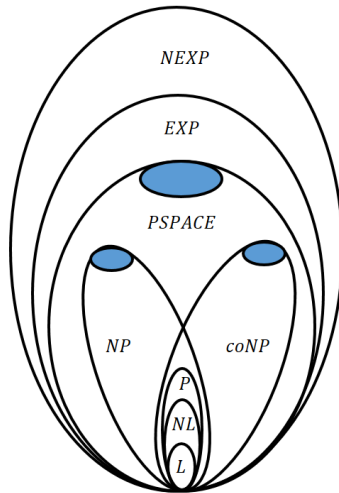
# CSL853: Complexity Theory

Ragesh Jaiswal, CSE, IIT Delhi

## Complexity Diagram

# Complexity Diagram

Our current view of complexity classes



# Space Complexity

## NL-completeness

### Definition (Implicitly logspace computable)

A function  $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$  is implicitly logspace computable, if  $f$  is polynomially bounded (i.e., there is some  $c$  such that  $|f(x)| \leq |x|^c$  for every  $x \in \{0, 1\}^*$ ) and the languages  $L_f = \{\langle x, i \rangle \mid f(x)_i = 1\}$  and  $L'_f = \{\langle x, i \rangle \mid i \leq |f(x)|\}$  are in  $L$ .

### Definition (Logspace reduction)

A language  $B$  is logspace reducible to language  $C$ , denoted by  $B \leq_l C$  if there is a function  $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$  that is implicitly logspace computable and  $x \in B$  iff  $f(x) \in C$  for every  $x \in \{0, 1\}^*$ .

### Definition (NL-complete)

A language  $L$  is said to be NL-complete if it is in NL and for every  $B \in \text{NL}$ ,  $B \leq_l L$ .

### Definition (Logspace reduction)

A language  $B$  is logspace reducible to language  $C$ , denoted by  $B \leq_l C$  if there is a function  $f : \{0, 1\}^* \rightarrow \{0, 1\}^*$  that is implicitly logspace computable and  $x \in B$  iff  $f(x) \in C$  for every  $x \in \{0, 1\}^*$ .

### Definition (NL-complete)

A language  $L$  is said to be NL-complete if it is in NL and for every  $B \in \text{NL}$ ,  $B \leq_l L$ .

- Show that if  $B \leq_l C$  and  $C \leq_l D$ , then  $B \leq_l D$ .
- Show that if  $B \leq_l C$  and  $C \in \text{L}$ , then  $B \in \text{L}$ .

# Space Complexity

## NL-completeness

### Definition (NL-complete)

A language  $L$  is said to be NL-complete if it is in NL and for every  $B \in \text{NL}$ ,  $B \leq_I L$ .

### Theorem

PATH is NL-complete.

# Space Complexity

NL-completeness

## Theorem

PATH is NL-complete.

## Proof.

- Claim 1:  $\text{PATH} \in \text{NL}$ .
- Claim 2: For every  $L \in \text{NL}$ ,  $L \leq_I \text{PATH}$ .



# Space Complexity

NL: Certificate definition

## Definition (Alternate definition of NL)

A language  $L \in \text{NL}$  if there exists a deterministic TM  $M$  (called the verifier) with an additional special read-once input tape, and a polynomial  $p : \mathbb{N} \rightarrow \mathbb{N}$  such that for every  $x \in \{0, 1\}^*$ ,

$$x \in L \Leftrightarrow \exists u \in \{0, 1\}^{p(|x|)} \text{ s.t. } M(x, u) = 1$$

where by  $M(x, u)$  we denote the output of  $M$  where  $x$  is placed on its input tape and  $u$  is placed on its special read-once tape, and  $M$  uses at most  $O(\log |x|)$  space on its read/write tapes for every input  $x$ .

Theorem (Immerman-Szelepcsenyi Theorem, Immerman 1988, Szelepcsenyi 1989)

$\overline{\text{PATH}} \in \text{NL}$ .

# Space Complexity

NL = coNL

Theorem (Immerman-Szelepcsenyi Theorem, Immerman 1988, Szelepcsenyi 1989)

$\overline{\text{PATH}} \in \text{NL}.$

Corollary

$\text{NL} = \text{coNL}.$

Theorem (Immerman-Szelepcsenyi Theorem, Immerman 1988, Szelepcsenyi 1989)

$\overline{\text{PATH}} \in \text{NL}$ .

Proof.

- Read-once certificate for no path: We will show that there is a read-once verifier that takes as input  $(G, s, t)$  and a polynomial size certificate  $u$  (this is read-once certificate). There is NO path from  $s$  to  $t$  in  $G$  if and only if there is a  $u$  such that the verifier accepts. Moreover, this verifier should work in logarithmic space.

# Space Complexity

NL = coNL

Theorem (Immerman-Szelepcsenyi Theorem, Immerman 1988, Szelepcsenyi 1989)

$\overline{\text{PATH}} \in \text{NL}.$

Proof.

- Read-once certificate for no path:
  - What is a short certificate of the fact that there is no path from  $s$  to  $t$  in  $G$ ?

Theorem (Immerman-Szelepcsényi Theorem, Immerman 1988, Szelepcsényi 1989)

$\overline{\text{PATH}} \in \text{NL}$ .

Proof.

- Read-once certificate for no path:
  - Let  $C_i$  denote the set of vertices that are reachable from  $s$  in at most  $i$  steps.
  - Claim 1: There is a short read-once certificate that a vertex  $v$  is in  $C_i$ .

Theorem (Immerman-Szelepcsenyi Theorem, Immerman 1988, Szelepcsenyi 1989)

$\overline{\text{PATH}} \in \text{NL}.$

Proof.

- Read-once certificate for no path:
  - Let  $C_i$  denote the set of vertices that are reachable from  $s$  in at most  $i$  steps.
  - Claim 1: There is a short read-once certificate that a vertex  $v$  is in  $C_i$ .
  - Claim 2: Given that there is a short read-once certificate that  $C_i = c$ , there is a short read-once certificate that  $v \notin C_i$  for any vertex  $v$ .

# Space Complexity

NL = coNL

Theorem (Immerman-Szelepcsenyi Theorem, Immerman 1988, Szelepcsenyi 1989)

$\overline{\text{PATH}} \in \text{NL}$ .

Proof.

- Read-once certificate for no path:
  - Let  $C_i$  denote the set of vertices that are reachable from  $s$  in at most  $i$  steps.
  - Claim 1: There is a short read-once certificate that a vertex  $v$  is in  $C_i$ .
  - Claim 2: Given that there is a short read-once certificate that  $C_i = c$ , there is a short read-once certificate that  $v \notin C_i$  for any vertex  $v$ .
  - Claim 3: Given that there is a short read-once certificate that  $C_{i-1} = c$ , there is a short read-once certificate that  $v \notin C_i$  for any vertex  $v$ .

# Space Complexity

NL = coNL

Theorem (Immerman-Szelepcsenyi Theorem, Immerman 1988, Szelepcsenyi 1989)

$\overline{\text{PATH}} \in \text{NL}$ .

Proof.

- Read-once certificate for no path:
  - Let  $C_i$  denote the set of vertices that are reachable from  $s$  in at most  $i$  steps.
  - Claim 1: There is a short read-once certificate that a vertex  $v$  is in  $C_i$ .
  - Claim 2: Given that there is a short read-once certificate that  $C_i = c$ , there is a short read-once certificate that  $v \notin C_i$  for any vertex  $v$ .
  - Claim 3: Given that there is a short read-once certificate that  $C_{i-1} = c$ , there is a short read-once certificate that  $v \notin C_i$  for any vertex  $v$ .
  - Claim 4: There is a short read-once certificate that  $C_i = c$ .



End