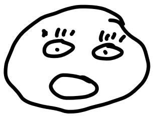
Last time: REACH AB Chap. 4.2, Papad. 19.1, Sipser 8.3, Cai 3.4

- NL, NL-completeness (under logspace reduc.)
- PSPACE, PSPACE-completeness (under poly-time reduc.)

Today:

Same readings as

- finish PSPACE-completeness of QSAT
- PSPACE + 2-player games
- Generalized Geography is PSPACE-complete
- start Immerman - Szelepcsényi thm: nondet space is closed under complement
  ↳ NL = co-NL



Admin:

- PS 2 due Wed; midterm out on Sun (no latedays)
- OH recommended this week (no OH during midterm week, i.e. next week)

Questions?

Recall our setup for showing $L \leq_p$ QSAT:

any lang. in PSPACE

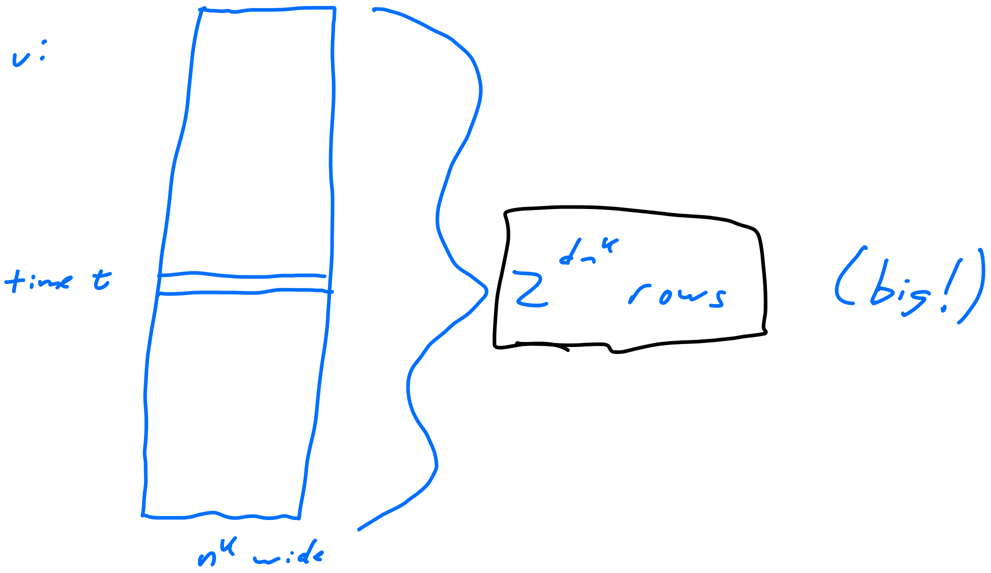
- $M = n^k$-space det TM for L (one tape)
- M always runs in $\leq 2^{dn^k}$ time, $\leq n^k$ space
- We need poly-time reduc. $x \rightarrow$ a (poly size!) QBF that's true iff $x \in L$.

3 ingredients:

① Cook-Levin thm: computation tableau of M on x. Grid T; cell $T_{t,j}$ of T is j^{th} cell of config of M on x at time t (tape cell contents, etc.)

Like in C-L thm, Bool. expressions enforce/check local consistency of tableau T (ensures T faithfully encodes comput. of M on x).

Tableau:



② Savitch's thm: recursively find midpoint of tableau.

Naively: one "$\exists$" for each midpoint/row of T: gives formula

$$\underbrace{\exists x_1 \ldots\ldots\ldots\ldots \exists}_{2^{dn^k} \text{ rows}} \varphi$$

③ Use univ. $\forall$ quant. to (expon!) save on size of formula.

Details:

Recall C-L setup: here, as there, given our input $|x| = n$, any config of M (row of tableau) has a binary encoding of length $O(n^k)$ (an asst to n^k Bool. vars).

- we assume wlog ! accepting config; of course, given x, there's a ! initial config.

Let c_1, c_2 be two configs. (really encoded into Bool vars; each c_i is an n^k-tuple of Bool. vars/bits).

$$\text{Let } LEGIT(c_1, c_2, i) = \begin{cases} T & \text{if } M \text{ goes from } c_1 \text{ to } c_2 \text{ in } \le 2^i \text{ steps} \\ F & \text{otherwise.} \end{cases}$$

So, M acc x iff $LEGIT(c_{init}, c_{accept}, d \cdot n^k) = T$ (here $c_{init}, c_{accept} \in \{0,1\}^{n^k}$ are bin. enc. of ! init + final-accept configs of M on x).

almost-tot. c_1, c_2 are free; this way,

Want a quant. Bool. form. for $LEGIT(c_1, c_2, i)$

when we plug in $c_1 = c_{init}$ + $c_2 = c_{accept}$, $i = dn^k$, it'll be either T or F.

We'll build rec. like in Savitch. $2^0 = 1$

$i=0$: have $LEGIT(c_1, c_2, 0) = \emptyset_0(c_1, c_2) \lor \emptyset_1(c_1, c_2)$ where

- $\emptyset_0(c_1, c_2) = T$ if c_2 follows from c_1 $\overset{\text{under } M}{\text{in}}$ 0 steps, i.e. $c_2 \equiv c_1$ $\boxed{(c_{2,1} = c_{1,1}) \land (c_{2,2} = c_{1,2}) \land \ldots}$
- $\emptyset_1(c_1, c_2) = T$ if c_2 follows from c_1 $\overset{\text{under } M}{\text{in}}$ 1 steps. As in C-L, there's a Bool. formula for this

$i>0$: Most naive: try

$LEGIT(a,b,i) =$

$\exists c_1 \exists c_2 \ldots \exists c_{2^i-1}\ LEGIT(a,c_1,0) \wedge LEGIT(c_1,c_2,0) \wedge \ldots \wedge LEGIT(c_{2^i-1},b,0).$

Correct, but much too big: 2^i-length formula. Bad idea.

Better idea: Savitch-style rec. midpoint:

$$\boxed{LEGIT(c_1,c_2,i) = \exists c'\ LEGIT(c_1,c',i-1) \wedge LEGIT(c',c_2,i-1).}$$

Correct, but still too big: doubles in size as i incr. by 1 ☹

Right approach: use $\forall$ to avoid doubling form. size:

$$LEGIT(c_1,c_2,i) = \exists c' \forall c_3 \forall c_4$$
$$\Big[\underbrace{((c_3=c_1 \wedge c_4=c') \vee (c_3=c' \wedge c_4=c_2))}_{O(n^k)} \Rightarrow \underbrace{LEGIT(c_3,c_4,i-1)}_{\text{length of } (i-1) \text{ version}}\Big]$$

$O(n^k)$ + length of $(i-1)$ version

Correct! How big?

length

$L(i) = O(n^k) + L(i-1)$

so $L(i) = O(i \cdot n^k)$

So $\downarrow$ gives us a poly length QBF for $LEGIT(c_1,c_2,i)$ (+ only non-quant. part is c_1, c_2, as it should be). ∎

$\longrightarrow$ QBF is PSPACE-complete!!

Pf

b/c PSPACE rel. to winning 2-player games.

QSAT: a 2-player game between A + E

$$\underline{\Phi} = \exists x_1 \forall x_2 \exists x_3 \ldots \qquad \phi(x_1, x_2, \ldots)$$

$\uparrow$ T or F both poss.

They jointly build an asst;

E tries to make ϕ
T, A " " " F.

E's turn: he chooses 0/1 for x_1
A's " : she " " " x_2
⋮

At end: have 0/1 values for all vars,
E wins if T, A if false

E has a winning strat. iff Φ is true.

Many other games, suitably abstracted, + gen., are PSPACE-complete.

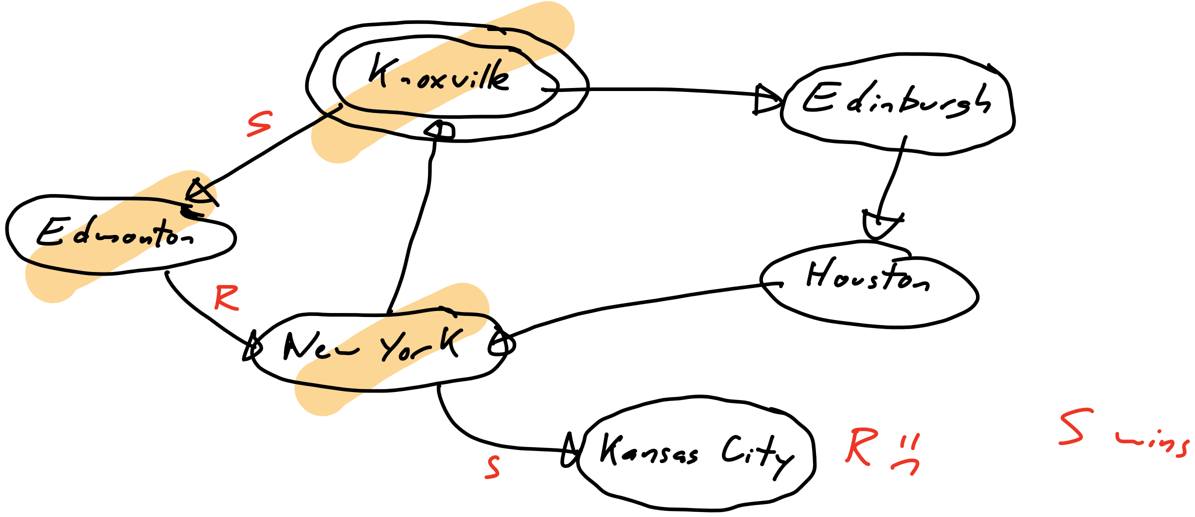
<u>Geography:</u>

PARIS
SYDNEY
YPSILANTI
⋮

Given a
Fixed set of cities: game corr. to a digraph

- nodes for cities
- edges based on 1st / last letters.

Players alternate moves: first one who can't move (no unoccupied nodes reachable from curr. node, or curr. node has no out-edges) loses.



Lang : $GG = \{ (\overset{\text{digraph}}{G}, \overset{\text{start vtx}}{v}) :$ player 1 has a winning strat. on G with v as start vtx.$\}$

<u>Thm</u>: GG is PSPACE-complete.

<u>Pf</u> 1) $GG \in PSPACE$: alg A to decide (G, v):

```
on input (G, v)
    • if no out-edges from V: return F
    • o/w, remove v + all v ⟶ [illegible] edges to get G'
      for each v' ∈ G' st v ⟶ v' was an edge in G:
      call A on (G', v')  (reusing space for each call)
```

If any call $\downarrow$ returns F: other player can be made to lose so return T.

If every call | T: other player | always win so return F.

PSPACE alg: depth of rec. tree $\leq$ # nodes in G.

---

2) $\forall L \in PSPACE, \ L \leq_p GG.$

suff to show $QSAT \leq_p GG.$

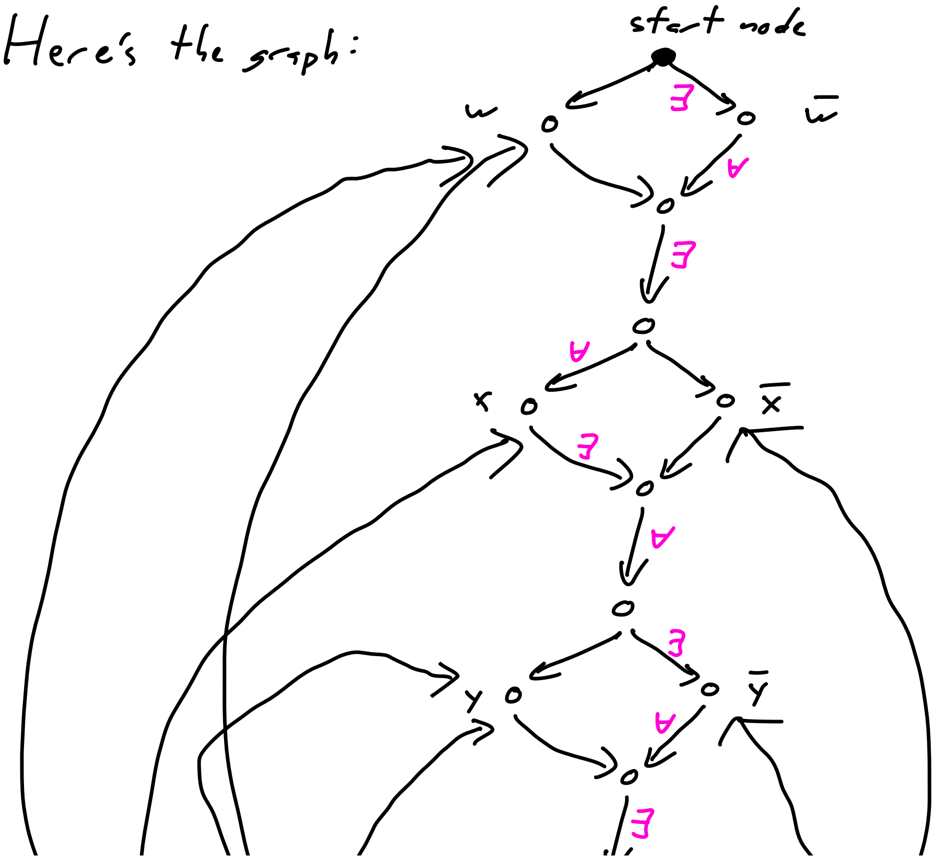
"starts w/ $\exists$"

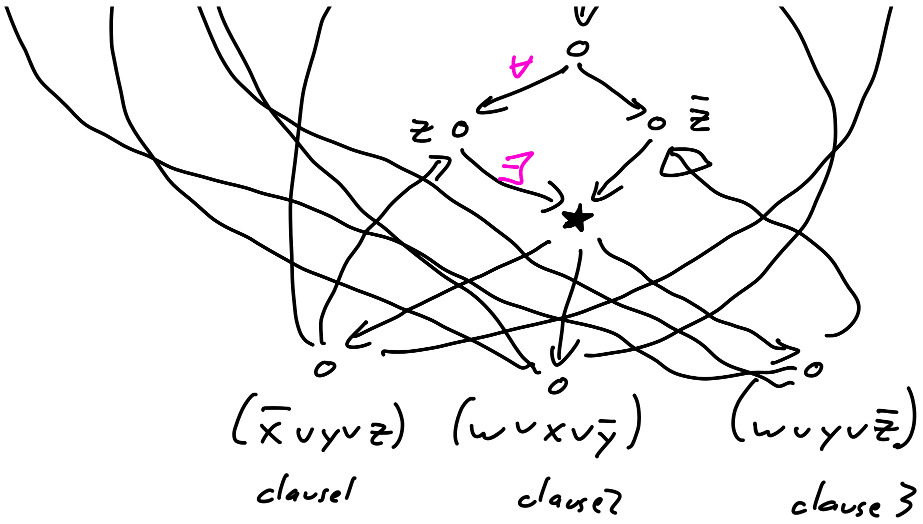
Reduc. by example: let

$$\Phi = \exists w \forall x \exists y \forall z \ (\overset{\checkmark}{\bar{x} \vee y \vee z}) \wedge (\overset{\checkmark}{w \vee x \vee \bar{y}}) \wedge (\overset{\checkmark}{w \vee y \vee \bar{z}}).$$

↳ T: take $w=1$: then $\forall x$, take $y=1$ → T.

Here's the graph:





G has •4 nodes for each var: diamond as above.
L node for var v is v, R node is $\bar{v}$

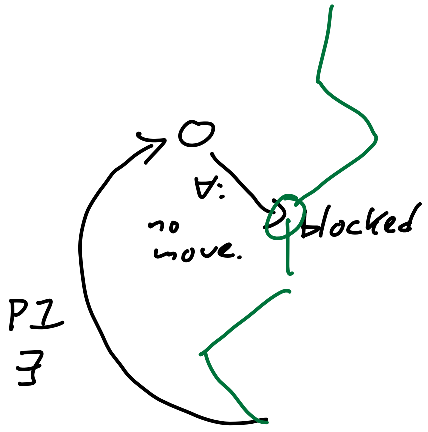
• a node for each clause C_i ; each C_i has an edge to each literal in it.

Any path thru G picks an asst 0/1 for each var: if player wants to set x=T, she chooses $\bar{x}$ side for path.

player 1 sets ∃ vars
" 2 " ∀ vars.

WLOG player 2's turn at ★: tries to pick a C_i that's unsat by the asst that's been built.

Player 1: can choose a lit. in C_i, but if C_i was not sat, player 1 loses w/ every choice; if some lit is sat, 1 chooses it & player 2 loses in next move



Next time: I-S thm.