CS 121: Lecture 21
More NP-completeness by Reductions

Adam Hesterberg

https://madhu.seas.Harvard.edu/courses/Fall2020

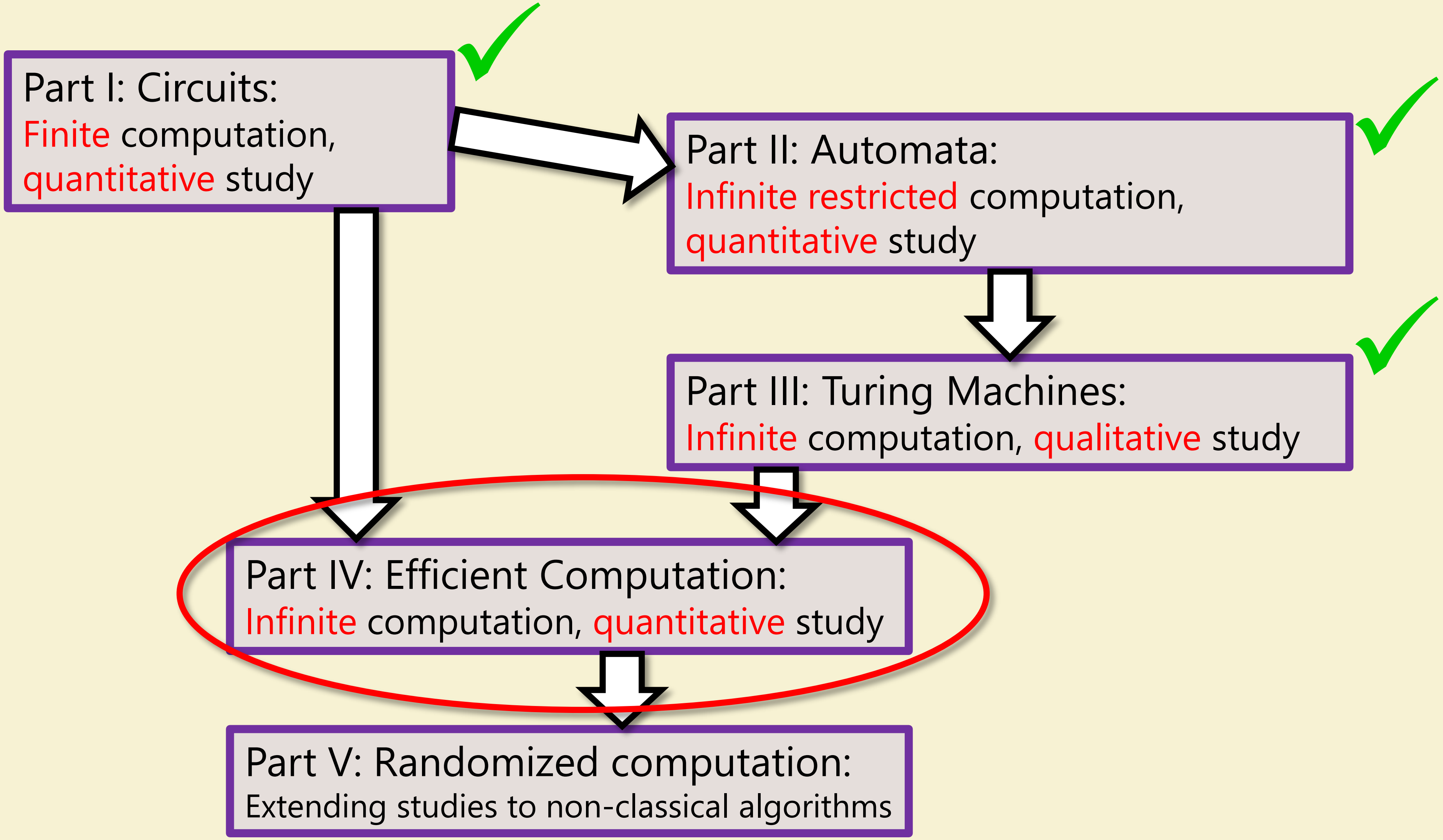
Book: https://introtcs.org

How to contact us
- The whole staff (faster response): CS 121 Piazza
- Only the course heads (slower): cs121.fall2020.course.heads@gmail.com

Announcements:

- 121.5: Nicole Immorlica: Econ and CS
- Sections: Polynomial time reductions, NP, etc.
- Homework 5 due today.
- Midterm 2 this Tuesday!
  - 90 minutes (70 if handwritten)
  - 2-sided cheatsheet, noncollaboratively made, plus Barak's textbook.
  - Material through lecture 17 (Efficient Computation: P)

Where we are:



Part I: Circuits:
Finite computation, quantitative study
Part II: Automata:
Infinite restricted computation, quantitative study
Part III: Turing Machines:
Infinite computation, qualitative study
Part IV: Efficient Computation:
Infinite computation, quantitative study
Part V: Randomized computation:
Extending studies to non-classical algorithms

Review of last lectures

(Poly-time)

- Reductions: $F \leq_P G \Leftrightarrow \exists R$ such that $\forall x\ F(x) = G\big(R(x)\big)$, R polytime.
  - 3SAT $\leq_P$ ISET
- NP: problems easy to verify.

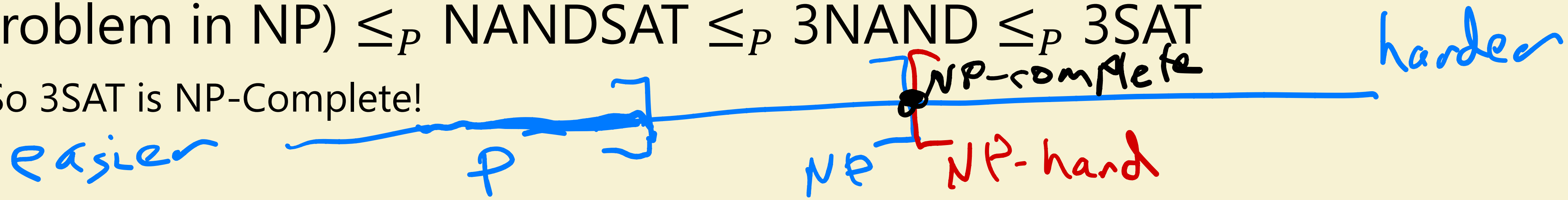
  $F: \{0,1\}^* \to \{0,1\}$ is in NP iff:

  $\exists V_F: \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$ s.t. $\forall x \in \{0,1\}^*$,

  Verifier

  $$F(x) = 1 \Leftrightarrow \exists w \in \{0,1\}^* \text{ such that } V_F(x,w) = 1$$

  and $V_F(x,w)$ computable in time poly($|x|$)
- (Any problem in NP) $\leq_P$ NANDSAT $\leq_P$ 3NAND $\leq_P$ 3SAT
  - So 3SAT is NP-Complete!



Witness, the NP concept

Function F is in NP if ∃ polytime V_F s.t. $(F(x) = 1) \Leftrightarrow (\exists w: V_F(x, w) = 1)$

Function F	**Witness w**	**Verifier V_F**
3SAT(formula)	Variable values	Check: formula satisfied?
Longpath(G, k)	Sequence of vertices	Check: is path, is long no repeated v, adj. pairs adj.
COMPOSITE(x)	Factors p, q	Check: p*q=x
COMPOSITE(x)	y, z	Check: $\frac{yz}{x} \in Z, \frac{y}{x} \notin Z, \frac{z}{x} \notin Z$

6 10 21

$\frac{210}{6}$ $\frac{10}{6}$ $\frac{21}{6}$

Witness, the computer game

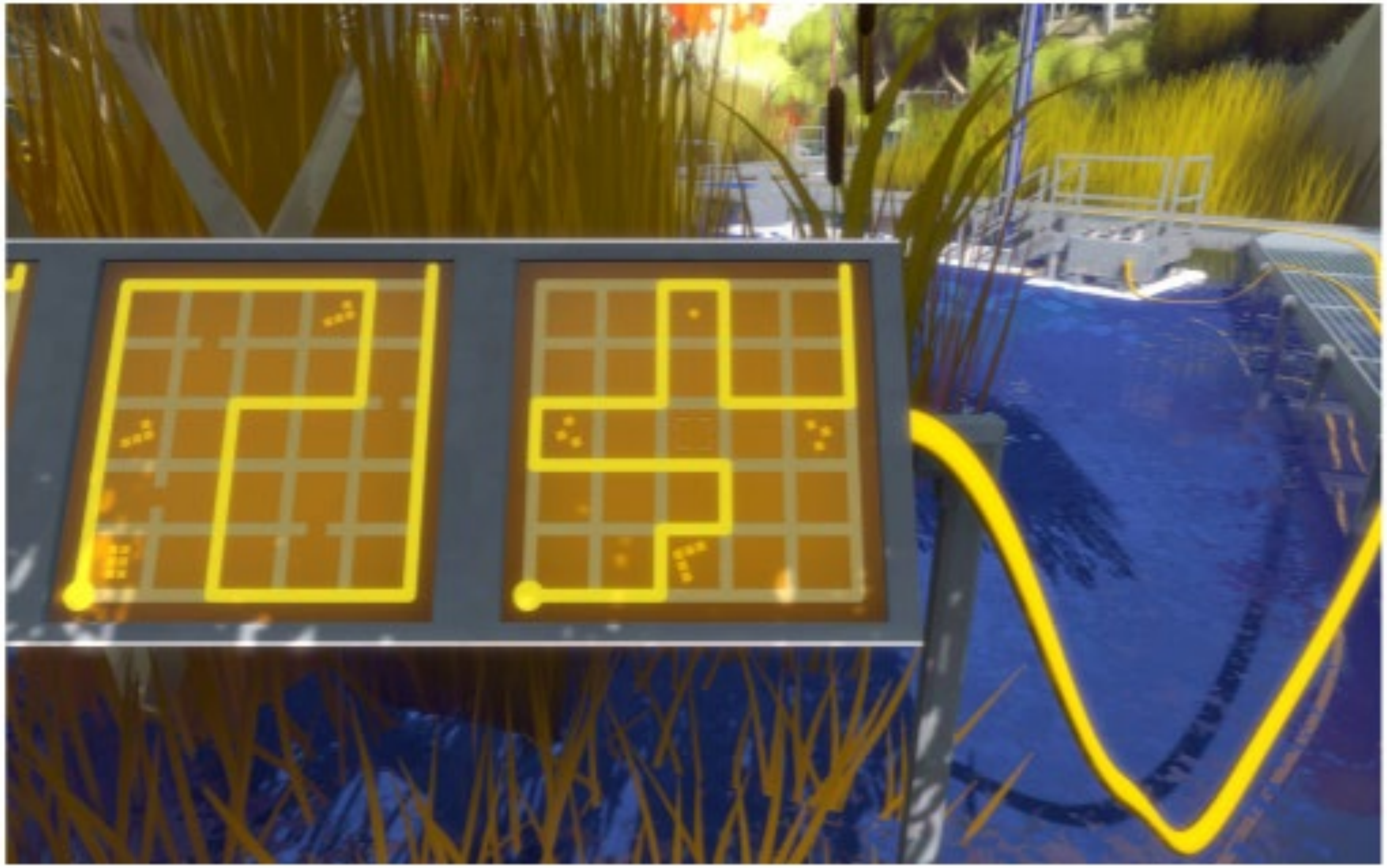


Figure 1: A screenshot from The Witness, featuring 2D puzzles in a 3D world.

Today:

- Some NP-complete problems…
- 3SAT $\leq_P$ E3SAT $\leq_P$ EU3SAT $\leq_P$ 1-in-EU3SAT $\leq_P$ SUBSETSUM
- Weak NP-hardness: hard only for big-number inputs
- Strong NP-hardness: hard even for small-number inputs.

3SAT $\leq_P$ E3SAT

Last time,

3NAND $\leq_P$ 3SAT :

$$c = NAND(a, b) \iff (\bar{a} \vee \bar{b} \vee \bar{c}) \wedge (c \vee a) \wedge (c \vee b)$$

3SAT: Formulas like $(x_7 \vee \bar{x}_{17} \vee x_{29}) \wedge (\bar{x}_7 \vee x_{15} \vee x_{22}) \wedge (x_{22} \vee \bar{x}_{29})$,

at most 3 variables/clause

Warning: sometimes "3SAT" = E3SAT.

E3SAT: Formulas like $(x_7 \vee \bar{x}_{17} \vee x_{29}) \wedge (\bar{x}_7 \vee x_{15} \vee x_{22}) \wedge (x_{22} \vee \bar{x}_{29} \vee x_{22})$,

exactly 3 variables/clause.

3SAT $\leq_P$ E3SAT

Reduction:

(x_7) ⇒ $(x_7 \lor x_7 \lor x_7)$

$(x_7 \lor \overline{x}_{17})$ ⇒ $(x_7 \lor \overline{x}_{17} \lor x_7)$

$(x_7 \lor \overline{x}_{17} \lor x_{29})$ ⇒ $(x_7 \lor \overline{x}_{17} \lor x_{29})$

$(x_7) \land (x_7 \lor \overline{x}_{17}) \land (x_7 \lor \overline{x}_{17} \lor x_{29})$ ⇒ $(x_7 \lor x_7 \lor x_7) \land (x_7 \lor \overline{x}_{17} \lor x_7) \land (x_7 \lor \overline{x}_{17} \lor x_{29})$

Add duplicate ~~variables~~ literals to fill clauses w/ <3 literals

Proof:

(Sound,

Complete)

$(x_7) \land (x_7 \lor \overline{x}_{17}) \land (x_7 \lor \overline{x}_{17} \lor x_{29})$ is satisfiable ⇔ $(x_7 \lor x_7 \lor x_7) \land (x_7 \lor \overline{x}_{17} \lor x_7) \land (x_7 \lor \overline{x}_{17} \lor x_{29})$ is satisfiable

⇑

$(x_7 \lor \overline{x}_{17})$ is satisfiable ⇔ $(x_7 \lor \overline{x}_{17} \lor x_7)$ is satisfiable with the same variable values

$3SAT \leq_p E3SAT$

$E3SAT \leq_p 3SAT$

Reduction:

$(x_7) \Rightarrow (x_7 \lor x_7 \lor x_7)$

$(x_7 \lor \overline{x}_{17}) \Rightarrow (x_7 \lor \overline{x}_{17} \lor x_7)$

$(x_7 \lor \overline{x}_{17} \lor x_{29}) \Rightarrow (x_7 \lor \overline{x}_{17} \lor x_{29})$

$(x_7) \land (x_7 \lor \overline{x}_{17}) \land (x_7 \lor \overline{x}_{17} \lor x_{29}) \Rightarrow (x_7 \lor x_7 \lor x_7) \land (x_7 \lor \overline{x}_{17} \lor x_7) \land (x_7 \lor \overline{x}_{17} \lor x_{29})$

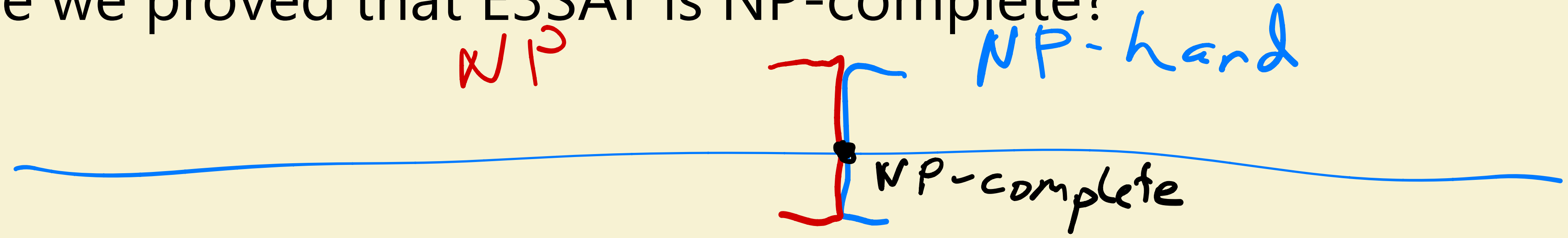
Proof:

(Sound,

Complete)

$(x_7) \land (x_7 \lor \overline{x}_{17}) \land (x_7 \lor \overline{x}_{17} \lor x_{29})$ is satisfiable $\Leftrightarrow$ $(x_7 \lor x_7 \lor x_7) \land (x_7 \lor \overline{x}_{17} \lor x_7) \land (x_7 \lor \overline{x}_{17} \lor x_{29})$ is satisfiable

Q: Have we proved that E3SAT is NP-complete?



E3SAT $\leq_P$ EU3SAT

3SAT: Formulas like $(x_7 \lor \bar{x}_{17} \lor x_{29}) \land (\bar{x}_7 \lor x_{15} \lor x_{22}) \land (x_{22} \lor \bar{x}_{29})$,

at most 3 variables/clause

literals

E3SAT: Formulas like $(x_7 \lor \bar{x}_{17} \lor x_{29}) \land (\bar{x}_7 \lor x_{15} \lor x_{22}) \land (x_{22} \lor \bar{x}_{29} \lor x_{22})$,

Exactly 3 variables/clause.

literals

EU3SAT: Formulas like $(x_7 \lor \bar{x}_{17} \lor x_{29}) \land (\bar{x}_7 \lor x_{15} \lor x_{22}) \land (x_{22} \lor \bar{x}_{29} \lor x_{23})$,

exactly 3 *unique* variables/clause.

literals

E3SAT $\leq_P$ EU3SAT

$a \vee a \vee a \to (a \vee b \vee c) \wedge (4 \text{ clauses}: b = a) \wedge (4 \text{ clauses}: c = b)$

Reduction:

$(x_7 \vee \overline{x}_{17} \vee x_7)$ ⇒

$(x_7 \vee \overline{x}_{17} \vee y_7) \wedge$
$(x_7 \vee \overline{y}_7 \vee temp) \wedge$
$(x_7 \vee \overline{y}_7 \vee \overline{temp}) \wedge$
$(\overline{x}_7 \vee y_7 \vee temp) \wedge$
$(\overline{x}_7 \vee y_7 \vee \overline{temp})$

$a \vee b \vee c \to a \vee b \vee c$

(Wherever we have t copies of a variable in a clause, change t-1 of them and add 4(t-1) clauses.)

Proof:

(Sound,

Complete)

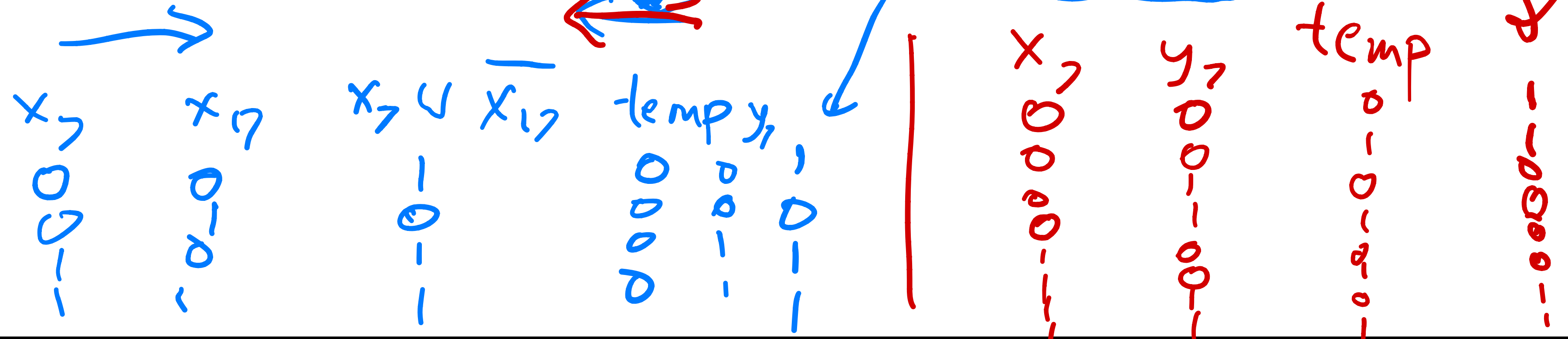
E3SAT formula with clauses like $(x_7 \vee \overline{x}_{17} \vee x_7)$ is satisfiable ⇔ ? EU3SAT formula with clauses like

$(x_7 \vee \overline{x}_{17} \vee y_7) \wedge$
$(x_7 \vee \overline{y}_7 \vee temp) \wedge$
$(x_7 \vee \overline{y}_7 \vee \overline{temp}) \wedge$
$(\overline{x}_7 \vee y_7 \vee temp) \wedge$
$(\overline{x}_7 \vee y_7 \vee \overline{temp}) \wedge$

is satisfiable

a var.

a, ā literals



EU3SAT $\leq_P$ 1-in-EU3SAT

3SAT: Formulas like $(x_7 \vee \overline{x}_{17} \vee x_{29}) \wedge (\overline{x}_7 \vee x_{15} \vee x_{22}) \wedge (x_{22} \vee \overline{x}_{29})$,
at most 3 variables/clause, clause is satisfied iff **at least** one literal is true.

E3SAT: Formulas like $(x_7 \vee \overline{x}_{17} \vee x_{29}) \wedge (\overline{x}_7 \vee x_{15} \vee x_{22}) \wedge (x_{22} \vee \overline{x}_{29} \vee x_{22})$,
Exactly 3 variables/clause, clause is satisfied iff **at least** one literal is true.

EU3SAT: Formulas like $(x_7 \vee \overline{x}_{17} \vee x_{29}) \wedge (\overline{x}_7 \vee x_{15} \vee x_{22}) \wedge (x_{22} \vee \overline{x}_{29} \vee x_{23})$,
exactly 3 *unique* variables/clause, clause is satisfied iff **at least** one literal is true.

1-in-EU3SAT: Formulas like $\mathrm{ONEOF}(x_7, \overline{x}_{17}, x_{29}) \wedge \mathrm{ONEOF}(\overline{x}_7, x_{15}, x_{22}) \wedge \mathrm{ONEOF}(x_{22}, \overline{x}_{29}, x_{23})$,
exactly 3 *unique* variables/clause, clause is satisfied iff **exactly** one literal is true.

EU3SAT $\leq_P$ 1-in-EU3SAT

Reduction:

$(a \vee b \vee c)$ ∧ $\Rightarrow$ $ONEOF(\overline{a}, w, x) \wedge ONEOF(b, y, x) \wedge ONEOF(\overline{c}, w, z)$

$ONEOF(a, w_2, x_2) \wedge$ … new vars. specific to this clause

$(\overline{a} \vee d \vee \overline{e}) \wedge$

$(a \vee b \vee \overline{e})$

Proof:

(Sound,

Complete)

$(a \vee b \vee c)$ is satisfiable $\Leftrightarrow$? $ONEOF(\overline{a}, w, x) \wedge ONEOF(b, y, x) \wedge ONEOF(\overline{c}, y, z)$ is satisfiable

complete

sound

a	b	c	w	x	y	z
0	0	1	0	0	1	0

a	b	c	w	x	y	z
0	0	0	0	0		

More SAT variants…

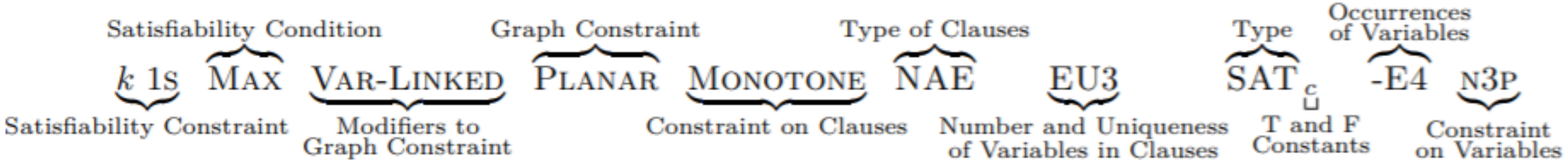
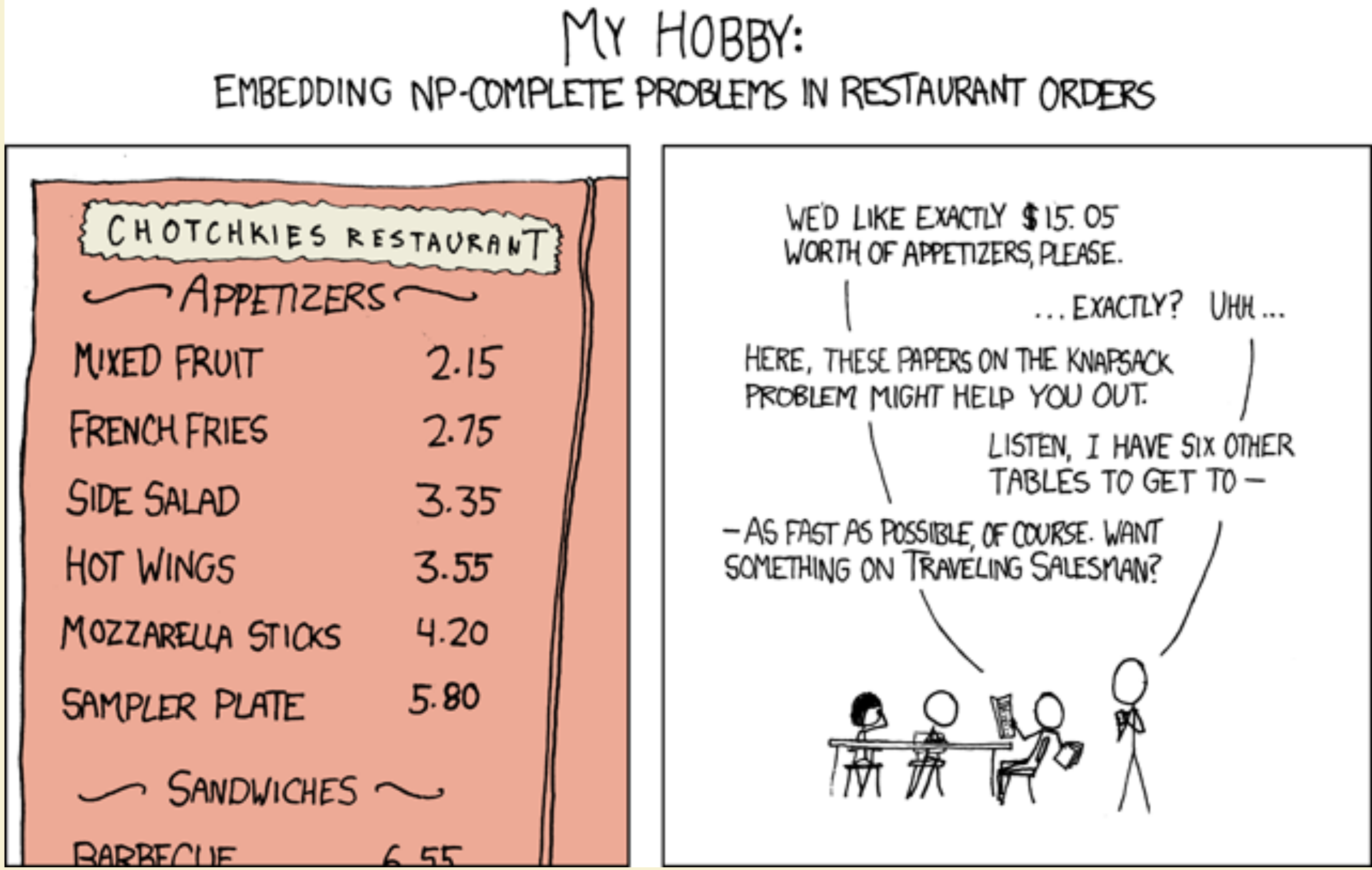


Figure 2-1: SAT notation example.

Knapsack Problem:

Given items with costs $a_0, a_1, \ldots, a_{k-1}$ and values $v_0, v_1, \ldots, v_{k-1}$, a budget b, and a target value t, choose a subset of the items with total cost at most b and value at least t.

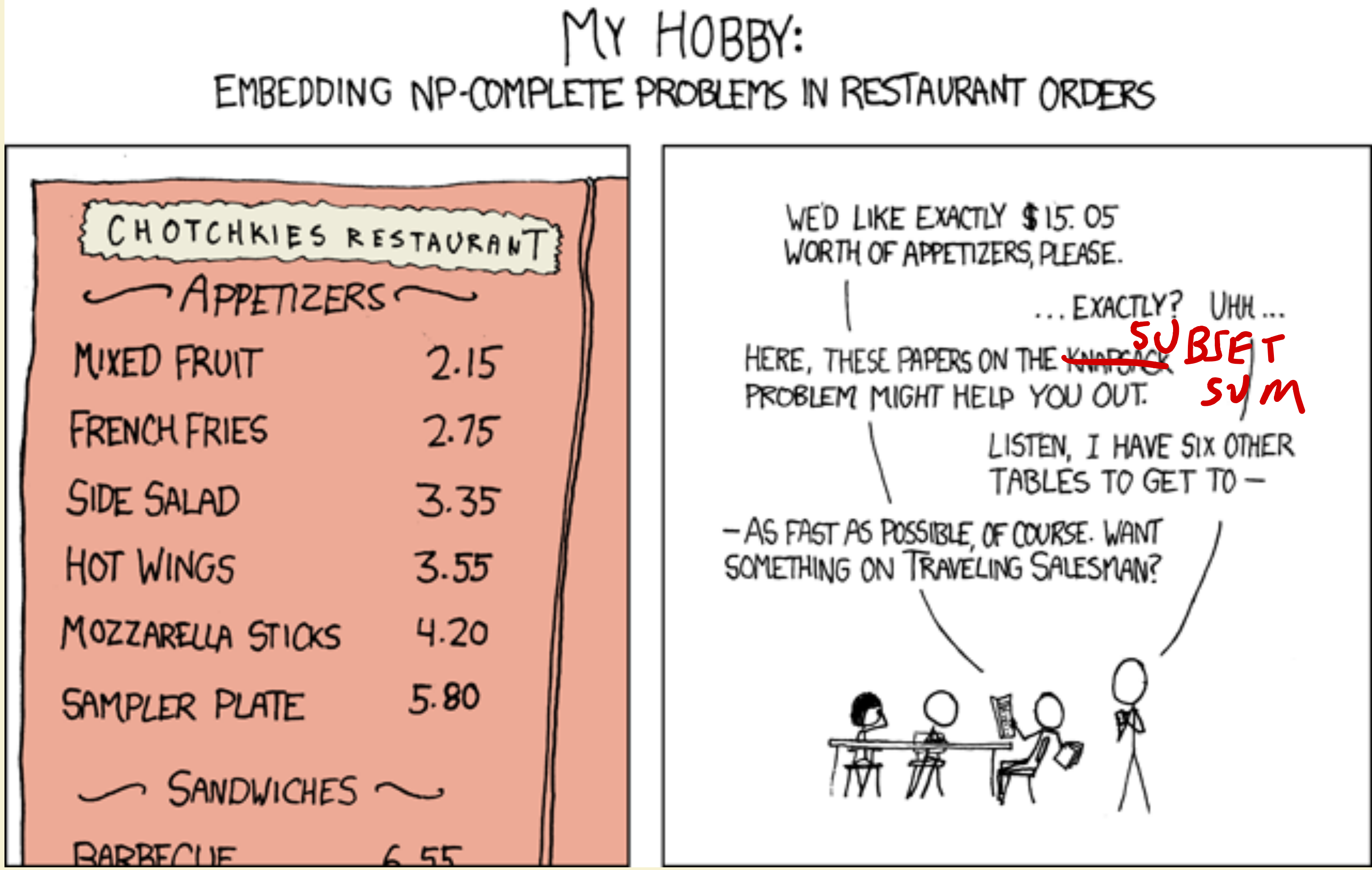


Knapsack Problem:

Given items with costs $a_0, a_1, \ldots, a_{k-1}$ and values $v_0, v_1, \ldots, v_{k-1}$, a budget b, and a target value t, choose a subset of the items with total cost at most b and value at least t.

Subset Sum:

Given items with costs $a_0, a_1, \ldots, a_{k-1}$ and a target value t, choose a subset of the items with total cost exactly t.



1-in-EU3SAT $\leq_P$ Subset Sum

Formulas like
$\text{ONEOF}(x_7, \overline{x}_{17}, x_{29}) \wedge$
$\text{ONEOF}(\overline{x}_7, x_{15}, x_{22}) \wedge$
$\text{ONEOF}(x_{22}, \overline{x}_{29}, x_{23})$

Given items with costs $a_0, a_1, \ldots, a_{k-1}$ and a target value t, choose a subset of the items with total cost exactly t.

Reduction:

1-in-EU3SAT formula
m clauses (here m=3)
n variables (here n=6)

$\text{ONEOF}(x_7, \overline{x}_{17}, x_{29})$
$\wedge\ \text{ONEOF}(\overline{x}_7, x_{15}, x_{22})$
$\wedge\ \text{ONEOF}(x_{22}, \overline{x}_{29}, x_{23})$

Subset Sum numbers (written in base $n+1$)

clauses / variable / variable

0	0	1	0		0	1	a_0
0	1	0	0		0	1	a_1
0	0	0	0		1	0	a_2
0	0	1	0		1	0	a_3
1	0	0	1		0	0	a_{2n-2}
0	0	0	1		0	0	a_{2n-1}
1	1	1	1		1	1	t

$> (n+1)^{n+m-1}$

x_7: T
x_7: F
x_{17}: T
x_{17}: F
x_{23}: T
x_{23}: F

literal
literal

Proof of Correctness?

Weak NP-hardness

Subset sum: Given items with costs $a_0, a_1, \ldots, a_{k-1}$ and a target value t, choose a subset of the items with total cost exactly t.

Some numbers (costs) in reduction were exponential in n. (Poly length!)

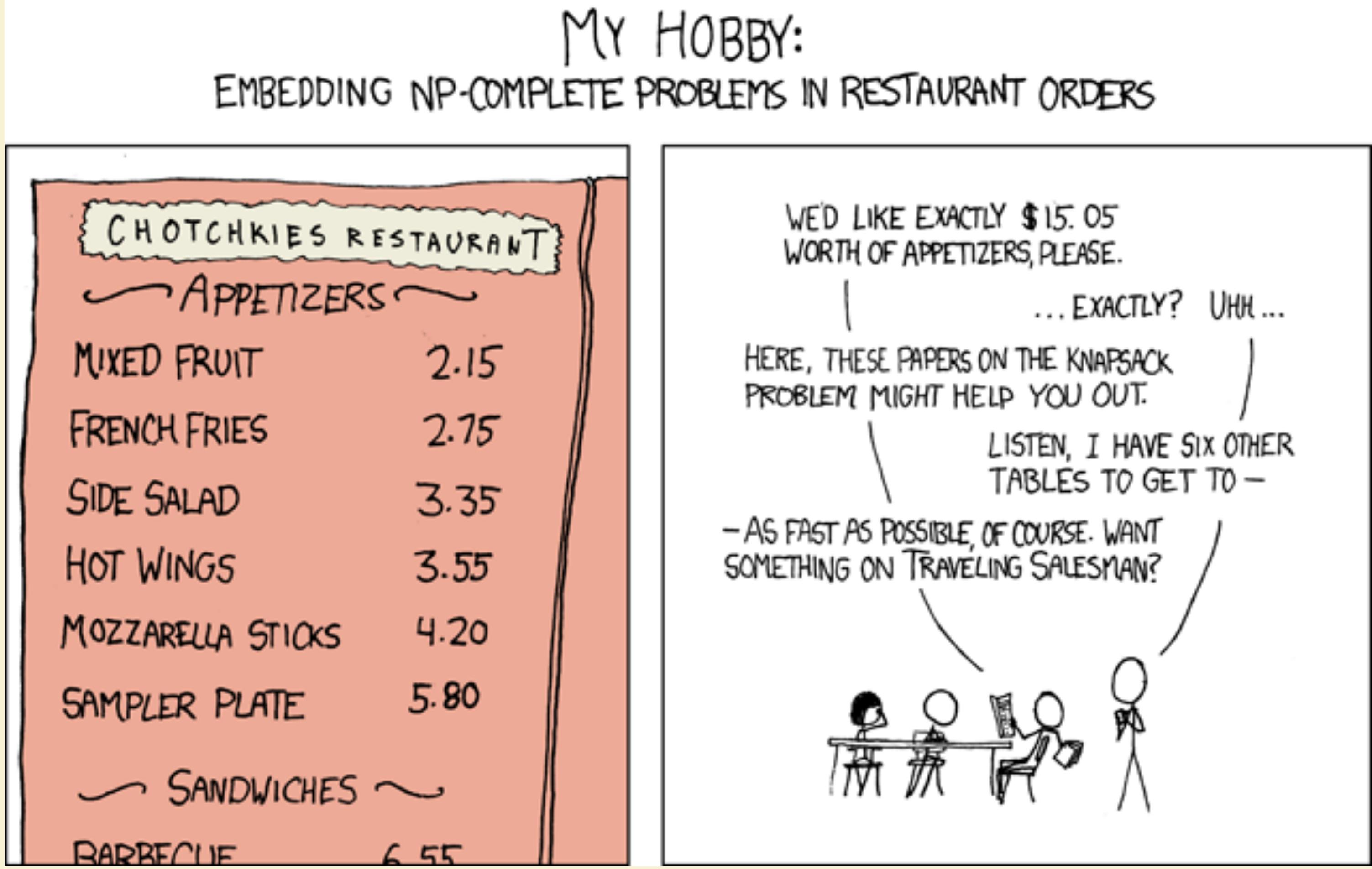
If all inputs were polynomial in n, Subset Sum isn't NP-hard.

"Weakly NP-hard"

"Strongly NP-hard": NP-hard even if all numerical inputs are polynomial-sized.

Traveling Salesman:

Given a (directed or undirected) graph G, a "distance" d_e for each edge e, and a target t, is there a walk visiting all the vertices of G whose total distance is at most t?



Strongly NP-hard
(NP-hard even if t and every d_e is small).

Hint: Reduce from Longpath:
Given a (directed or undirected) graph G and a target t, is there a path visiting at least t vertices? (Paths can't revisit vertices.)

Summary of Lecture:

- 3SAT $\leq_P$ E3SAT $\leq_P$ EU3SAT $\leq_P$ 1-in-EU3SAT $\leq_P$ SUBSETSUM
- Weak NP-hardness: hard only for big-number inputs
- Strong NP-hardness: hard even for small-number inputs.