| 6.890 | Lecture 6 | Sept. 23, 2014 |

<u>2 ways</u> to represent <u>variables</u> in 3SAT:

① <u>Dual-rail logic</u>:
    — <u>variable gadget</u> forces exclusive OR of 2 "semiwires" (true & false)
    — <u>semiwire</u> connects to clauses ∋ variable
    ↖ active only when chosen
  (e.g. Nintendo, pushing blocks, Phutball — most 3SAT reductions we've seen)

② <u>Binary logic</u>:   (not just Circuit SAT)
    — <u>wire gadget</u> has 2 (types of) solutions
    — <u>split gadget</u> to make copies of wire
  (e.g. flat-foldable crease patterns)

    — Circuit SAT also needs <u>terminator gadget</u> to start a variable wire

↪ in both cases, may need
    — <u>turn</u> gadget to route (semi)wires
    — <u>crossover</u> gadget to cross (semi)wires

Akari/Light Up: [Nikoli 2001]

- given square grid with some obstacles
- some obstacles have a number
  → how many (0-4) edge-adjacent lights
- light illuminates like rook, up to obstacles
- goal: place lights in blanks so that
  - black space lit
  - no lights light each other
  - satisfy numbers

NP-complete by reduction from Circuit SAT:
[McPhail 2005]

- wire, turn gadgets
- split/negation gadget
  → split & negation gadgets (via terminators)
- OR/XNOR gate
- crossover gadget: just XORs!

Minesweeper: given square grid of numbers & unknowns & possibly mines

Consistency: does there exist a solution?
- e.g. see whether mine at x is consistent with (consistent) info so far: if not, play x
  → special case of interest

NP-complete by reduction from Circuit SAT
[Kaye 2000]

- wire, terminator
- split/NOT/turn
- phase changer (shift by 2) via 2 NOTs
- AND
- crossover gadget: just use NANDs!
[Goldschlager 1977]

<u>Winning</u>: can I force a win? (no guessing)
i.e. figure out all squares?

[Hearn 2006]

<u>Inference</u>: can I figure out any squares?

[Scott, Stege, van Rooij 2011]

$\in$ <u>CoNP</u>: proof of NO = 2 differing solutions

<u>CoNP-complete</u> by reduction from
<u>Circuit UNSAT</u>: $\neg \exists x_1 \wedge x_2 \wedge \ldots \wedge x_n$ s.t. $f(\vec{x})$
$\equiv \forall x_1 \wedge x_2 \wedge \ldots \wedge x_n : \neg f(\vec{x})$

- wire, turn, terminator
- NOT, OR, shifter
- split
- crossover : just use NORs !

- special care to ensure equal # mines
  in all cases (# mines part of puzzle)
& ports aligned (middle of 3)

- unsatisfiable $\Leftrightarrow$ output forced to be F

<u>Planar Circuit SAT</u>: given noncrossing circuit
- only NAND   or   - only NOR
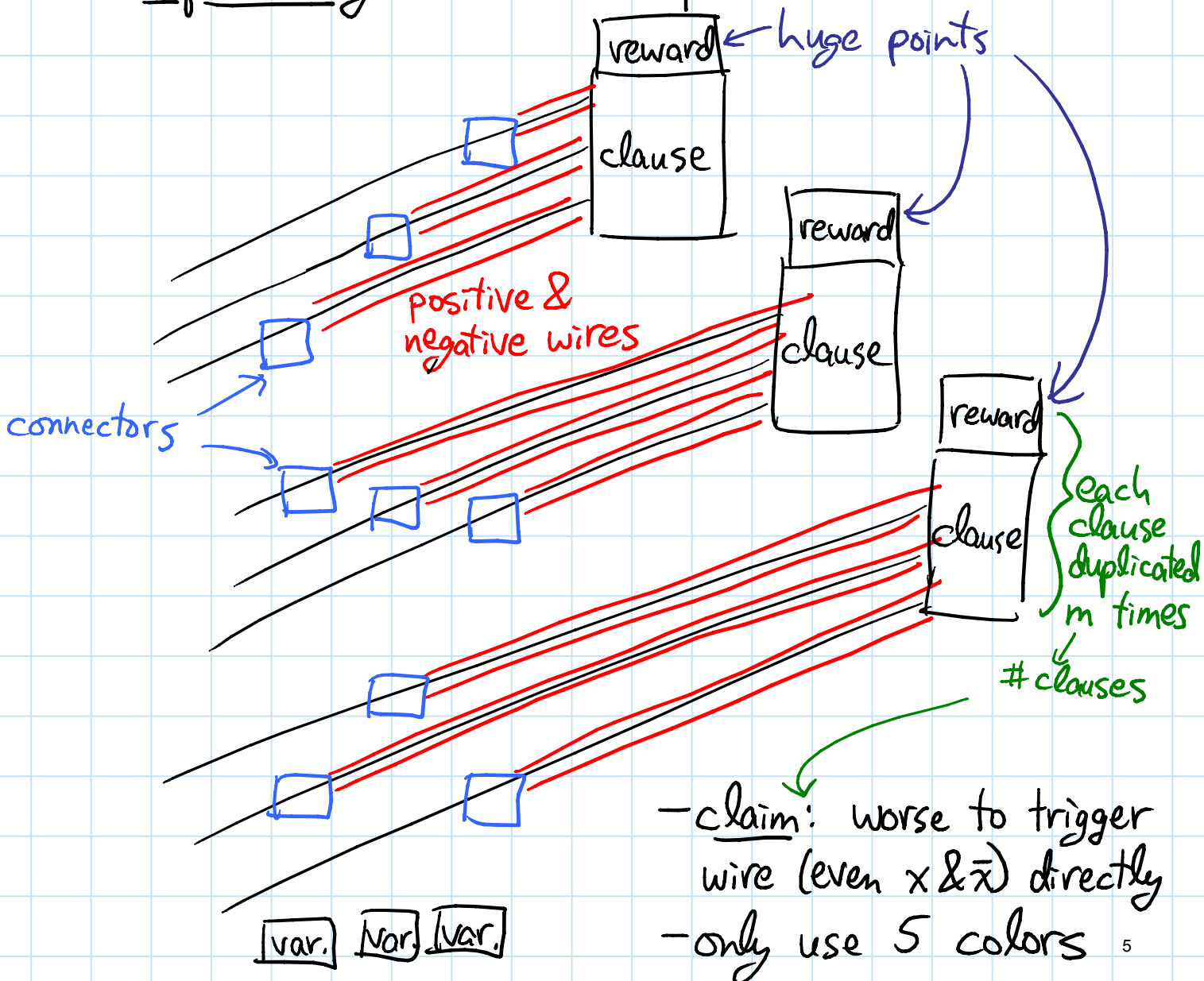(& splitters)   [new?]

4

Candy Crush / Bejeweled   (perfect information)
- given square grid of colors (among 6
- move = swap two edge-adjacent squares
- whenever 3 equal colors in a row/column:
    3 squares disappear & columns fall
                    ↳ "pop"

NP-complete to get p points with k moves
by reduction from 3SAT   [Walsh 2014]
... in model where pops happen
    sequentially bottom to top

reward ← huge points

clause

reward

positive &
negative wires

clause

reward

connectors

clause

each
clause
duplicated
m times

#clauses

- claim: worse to trigger
    wire (even x & x̄) directly
- only use 5 colors

var. var. var.

5

<u>NP-complete</u> with simultaneous pops
by reduction from 1-in-3SAT

- works for many goals:
  - p points in k moves
  - p points
  - pop p gems
  - p moves
  - pop a specific gem

6.890 Algorithmic Lower Bounds: Fun with Hardness Proofs
Fall 2014