Algorithmic Lower Bounds: Fun with Hardness Proofs

Super Mario Bros.

Rush Hour

Minesweeper

Hardness Made Easy*

Learn when to give up the search for efficient algorithms; see connections between computational problems; solve puzzles to prove theorems, solve open problems, and write papers.

Topics: NP, PSPACE, EXPTIME, EXPSPACE, 3SUM, approximation, fixed parameter, games & puzzles, 3SUM, key problems, gadgets, and proof styles.

6.890 taught by Professor Erik Demaine
Grad H, AUS, and Theoretical CS Concentration
Tuesday & Thursday 3:30-5:00pm in room 2-105
http://courses.csail.mit.edu/6.890/
sign up for our mailing list to join the class

Fall 2014

* Easiness not guaranteed. Side effects such as open problems and a heightened sense of complexity may occur. Ask your advisor if 6.890 is right for you!
6.890: Algorithmic Lower Bounds
"Hardness made Easy"

Prof. Erik Demaine
TAs: Sarah Eisenstat & Jayson Lynch
http://courses.csail.mit.edu/6.890/fall14/

What is this class?
- practical guide to proving computational problems are formally hard/intractable
- NOT a complexity course
  (but we will use/refer to needed results)
- (anti)algorithmic perspective

Why take this class?
- know your limits in algorithmic design
- master techniques for proving hardness
- cool connections between problems
- fun problems like Mario & Tetris
  (serious problems too)
- solve puzzles → publishable papers

Background: algorithms, asymptotics, combinatorics
- no complexity background needed
  (but also little overlap with a complexity class)
Classic Nintendo Games are (NP-)Hard

Greg Aloupis*       Erik D. Demaine†       Alan Guo‡

March 26, 2012

Science Proves Old Video Games Were Super Hard
Super Mario Bros. is NP-Hard
[Aloupis, Demaine, Guo 2012]

The Lost Levels
Super Mario Bros. is NP-Hard

[Aloupis, Demaine, Guo 2012]
Super Mario Bros. is NP-Hard

[Aloupis, Demaine, Guo 2012]
Super Mario Bros. is NP-Hard
[Aloupis, Demaine, Guo, Viglietta 2014]
Super Mario Bros. is NP-Hard
[Aloupis, Demaine, Guo, Viglietta 2014]

\[(x \lor \neg y \lor z) \land (x \lor y \lor \neg y) \land (\neg x \lor \neg y \lor \neg z) \land (\neg x \lor \neg y \lor \neg z)\]
Super Mario Bros. is NP-Hard [Aloupis, Demaine, Guo, Viglietta 2014] crossover
Constraint Graphs

Machine = graph, red & blue edges
Constraint Graphs

Machine state = orientation

constraint graph
Constraint Logic

Rule: at least 2 units incoming at a vertex

Move: reverse an edge, preserving Rule
AND vertex

not your usual AND gate!

Rule: at least 2 units incoming at a vertex
**OR vertex**

not your usual OR gate!

**Rule:** at least 2 units incoming at a vertex
Decision Problem

can you reverse this edge?
Rush Hour is PSPACE-complete
[Flake & Baum 2002; Hearn & Demaine 2002]
Rush Hour is PSPACE-complete

[Flake & Baum 2002; Hearn & Demaine 2002]
Complexity of Games & Puzzles

- **0 players (simulation)**: \( \text{P} \)
- **1 player (puzzle)**: \( \text{NP} \)
- **2 players (game)**: \( \text{PSPACE} \)
- **team, imperfect info**: \( \text{NEXPTIME} \)

- **PSPACE**
- **PSPACE**
- **EXPTIME**
- **Undecidable bridge?**

Additional games and puzzles mentioned:
- *Rengo Kriegspiel?*
Constraint Logic
[Hearn & Demaine 2009]

0 players (simulation)
1 player (puzzle)
2 players (game)
team, imperfect info

PSPACE
PSPACE
EXPTIME
Undecidable

P
NP
PSPACE
NEXPTIME

(a) AND
(b) FANOUT
(c) OR

unbounded
bounded

true variable
false
slow win
slower win
White win
Black win
true
false
true
false