The complexity of Minesweeper and game playing strategies

Kasper Pedersen

March 10, 2004

Overview

- Minesweeper: the game and motivation
- The complexity of Minesweeper
- Some strategies for playing Minesweeper

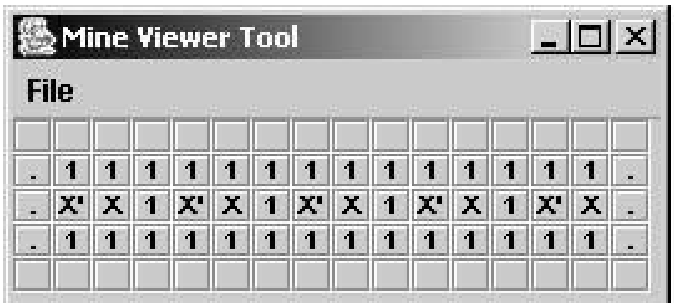
Minesweeper

- Played on *k* by *l* board with *m* hidden mines.
- To perform a move, the player must either choose an unlabelled square to *probe* or place a label on a free square.
- If a probed square contains a mine, the game is lost.
- If a probe is successful, the player is given information about the number mines adjacent to the probed square.
- The objective is to uncover all squares not containing a mine.

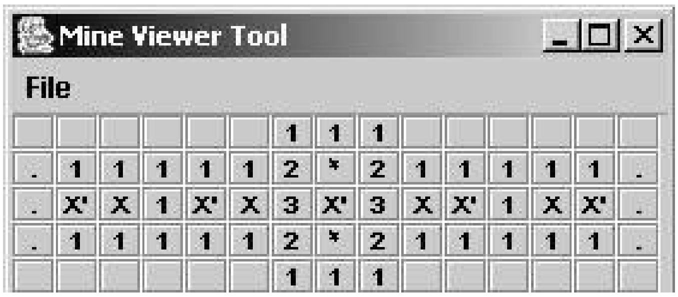
Motivation

- Minesweeper looks easy to play...
- ... Minesweeper is NP-complete
- **Def.** (Consistency) Given a rectangular grid partially marked with numbers and/or mines, some squares being blank, determine if there is *some* pattern of mines in the blank squares that give rise to the numbers seen (Kaye 2000).
- **Thm.** (Kaye) Consistency is NP-complete.

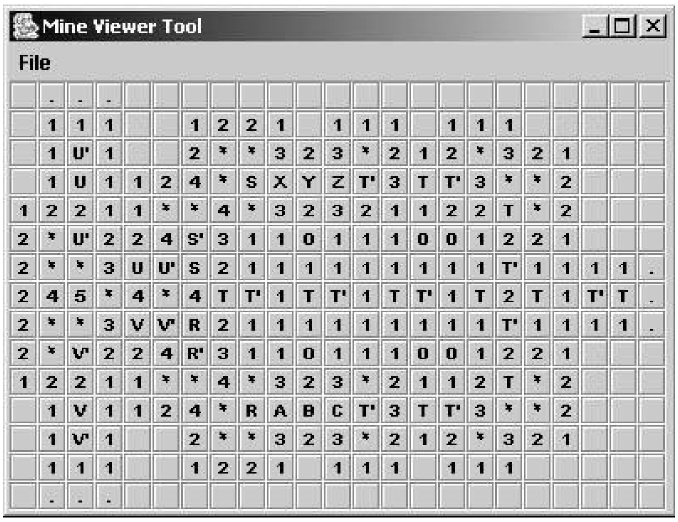
A wire



A NOT gate



An AND gate



The complexity of Minesweeper

- Is there a unique solution?
- Is there a winning strategy?

Minesweeper terminology

- **Def. 1** (Configuration) A Minesweeper configuration is a grid partially marked with numbers and/or mines, some squares remaining blank.
- **Def. 2** (Explanation) An explanation for a configuration *B* is an assignment of mines to the empty squares of the grid that gives rise to *B*.

Is there a unique solution?

- **Def. 3** (Solution) Input. A configuration *B*. Output. If a unique explanation of *B* exists, output it. Else, output “no“.

- **Def. 4** (#Consistency) Given a configuration *B*, output the number of explanations of *B*.

The complexity class #P

- Complexity class associated with finding the number of solutions to a computational problem.
- Contains complete problems.
- Problems that are #P-hard are much harder to solve than NP-hard problems.
- #SAT is #P-complete.

Is there a unique solution?

Thm. 1 #Consistency is #P-complete

Proof. (Outline) Reduction from #SAT.

1. Convert an instance *S* of SAT into an equivalent instance *S*‘ which only contains negations and conjunctions.
2. Convert *S*‘ into a Minesweeper configuration using the given gadgets.

The number of solutions to the resulting Minesweeper configuration = the number of solution to original SAT instance ■

The complexity class DP

- A language L is in the class DP iff there are two languages $L_1 \in$ NP and $L_2 \in$ coNP such that $L_1 \cap L_2 = L$.
- In general DP ≠ NP ∩ coNP.
- (DP is not likely to be contained in NP ∪ coNP.)
- DP is a syntactic class hence containing *complete problems*.
- Unique SAT is DP-complete.

Is there a unique solution?

Thm. 2 Solution ∈ DP

Proof. (Outline)

$L_1 = \{X \mid$ There exists an explanation for $X\}$

$L_2 = \{X \mid X$ has at most 1 explanation$\}$

$L_1 \cap L_2 = \{X \mid X$ has exactly 1 solution$\}$ ■

Is there a unique solution?

Thm. 3 Solution is DP-complete

Proof. (outline)

- DP-membership proved in Thm 2.
- DP-hardness by reduction from Unique SAT.

1. Convert an instance *S* of Unique SAT into an equivalent instance *S*‘ which only contains negations and conjunctions.
2. Convert *S*‘ into a Minesweeper configuration using the given gadgets.

■

Is there a winning strategy?

Def. 5 (Move) A move *M* is a pair ($x \in X, m \in \{0,1\}$), *X* is the set of positions:

- *M* = (*x*,0) => probing square *x*
- *M* = (*x*,1) => placing a mine on square *x*.

Def. 6 (Move safety)

- (*x*,0) is safe from *B* iff *B* has no explanations with a mine on *x*.
- (x,1) is safe from *B* iff *B* has no explanations with *x* mine-free.

Is there a winning strategy?

Def. 7 (Safety) Input. A config. *B* and a move *M*. Output:

- For $(B,(x,0))$ return yes if *B* has an explanation with no mine at *x* and no explanation with a mine at *x*. Else, “no“.
- For $(B,(x,1))$ return yes if *B* has an explanation with a mine at *x* and no explanation with no mine at *x*. Else, “no“.

Is there a winning strategy?

Thm. 4 Safety $\in$ DP.

Proof. (Outline)

$S = \{(B,(x,0)) \mid B$ has exp with x mine-free$\}$

$R = \{(B,(x,1)) \mid B$ has exp with x mine$\}$

$L_1 = S \cup R$

$L_2 = \{(B,M) \mid M$ is safe from $B\}$

$L = L_1 \cap L_2$. ■

Is there a winning strategy?

Thm. 5 Safety is DP-complete.

Proof. Reduction from solution (Def. 3)

Consider a configuration *B*.

for each unknown square $s \in B$

  b_1 = safety(*B*,(*s*,1)); b_2 = safety(*B*,(*s*,0))

  if $b_1 = b_2$ **return** “no“

  make appropriate assignment to *s*

return *B* ■

Game playing strategies

Aims

- Never perform a risky move unless necessary.
- When guessing, maximise the probability of a successful guess.

The first move

- First move safe!
- Maximise probability of getting a 0 => probe a corner square.

Single point strategy

- Maintains a set *S* of known safe moves.
- If $S \neq \{\}$ select a move (*x*,0) from *S* and perform it, otherwise guess *x* randomly.
- If all mines have been located around *x* => add all neighbours(*x*) to *S*.
- Checks if # adjacent mines(*x*) + # free adjacent squares(*x*) = label(*x*) => place mine symbols on all adj(*x*).

Single Point Strategy- Evaluation

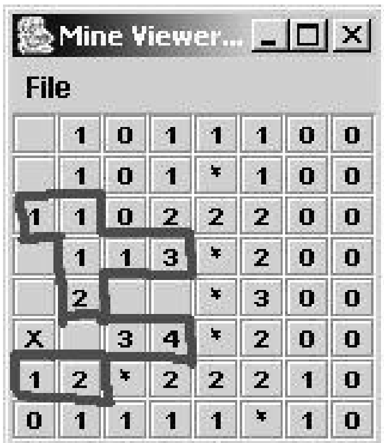
Success rate for playing 100000 games:

- Beginner (10 × 10, 10 mines): 74.9%
- Intermediate (16 × 16, 40 mines): 27.8%
- Expert (30 × 16, 99 mines): 0.4%

Limited Search strategy

- Performs a depth-first search on a local area around the square in question.
- Initially assumes the square contains a mine
- Goal is to deduce that this assumption is false => the square is safe to probe.
- Local area = zone of interest defined by Peña and Wrobel.
- Zone: set of labelled squares adj to *x* and set of labelled squares adj to unlabeled neighbour of *x*.

Zone of interest



Limited Search - Evaluation

Success rate for playing 100 000 games:

- Beginner: 91.5%
- Intermediate: 64.7%
- Expert: 16.9%

Adding probability estimation

- Poor performance on the advanced level caused by frequent guessing.
- Can improve guessing by estimating the prob of a square being a mine (note: #P-completeness => cannot calculate exact prob).
- Extension of limited search: instead of just terminating when contradiction is reached, returns number of solutions found (if no solutions found, this implies a contradiction).
- Use static probabilities as estimate when no other information is available.

Prob estimation - Evaluation

Success rate for playing 100 000 games:

- Beginner: 92.6%
- Intermediate: 67.5%
- Expert: 24.2%

Large search strategy

- Extend limited search by searching an arbitrarily large area rather than just zone.
- Expect this to be slow, since branching factor is 8.
- Want to search at several different levels away from the square.
- Zone of interest has level $\preceq 3$.

Large search - Evaluation

Results from playing 1000 ‘expert‘ games:

- Level 2: 275
- Level 3: 262
- Level 4: 282
- Level 5: 276

Summary

- Determining whether there is a unique solution is DP-complete.
- Determining whether a strategy exists is DP-complete.
- Local search is a relatively successful strategy for playing Minesweeper.