Problem

In the following solitaire game, you are given an m × m board. On each of its m^2 positions lies either a blue stone, a red stone, or nothing at all. You play by removing stones from the board until each column contains only stones of a single color and each row contains at least one stone. You win if you achieve this objective. Winning may or may not be possible, depending upon the initial configuration. Let *SOLITAIRE* = { $\langle G\rangle$ | G is a winnable game configuration}. Prove that *SOLITAIRE* is NP-complete.

Step-by-step solution

Step 1 of 3

Definition of NP- Complete:

A language B is NP- Complete if it satisfies two conditions.

1. B is in NP

2. Every A is NP is polynomial time reducible to B i.e. B is NP-hard.

Objective of the SOLITAIRE Game:

• SOLITAIRE Game requires a $m\times m$ board.

• On each n^2 positions of $m\times m$ board a blue stone or a red stone or nothing is placed.

• Now the game is to remove the stones so that each column contains only stones of single color and each row contains at least one stone.

• The people who achieve this objective will win the game.

Now we have to show that SOLITAIRE is NP-Complete. Before this, we have to show that SOLITAIRE is in NP.

Comment

Step 2 of 3

SOLITAIRE is in NP:

SOLITAIRE $\in NP$ because it can be verified that a solution works in polynomial time.

Every Language in NP is polynomial time reducible to SOLITAIRE:

• We know that $\text{"3SAT} = \{\langle\phi\rangle \mid \phi \text{ is a satisfiable 3 cnf - formula}\}\text{"}$, three variables.

And $\text{"3SAT is NP- complete"}$

So, if we show that $\text{3SAT} \leq_P$ SOLITAIRE then SOLITAIRE is also NP- Complete.

• Given ϕ with m variables $V_1,\ldots,V_m$ and k clauses $C_1,\ldots,C_k$

• Now construct the following game g with $k\times m$ board.

Construction of $k\times m$ game of G:

Let us assume that ϕ has no clauses that contain both V_i *and* $\overline{V_i}$ because such clauses may

If the variable V_i is in clause C_i then put a blue stone in row C_i column V_i

If the variable $\overline{V_i}$ is in clause C_i then put a red stone in row C_i, column V_i then

$k\times m$ board can be changed to square board necessary without affecting solvability.

Comment

Step 3 of 3

Now we need to show that ϕ is satisfiable if and only if G has a solution:

If ϕ is satisfiable then G has a solution(Forward direction):

• A Satisfying assignment is taken.

• If V_i is true, remove the rod stone from the corresponding column.

• If V_i is false, remove the blue stone from corresponding column.

• So, stones corresponding to true literals remains.

• Because every clause has a true literal, every row has a stone.

• Therefore G has a solute or.

If G has a solution then ϕ is satisfiable(*backward direction*):

• Take a game solution.

• If the red stone removed from a column, set the corresponding variable true.

• If the bluestone is removed from a column, set the corresponding variable false.

• Every row has a stone remaining, so every clause has a true literal.

• Therefore ϕ is satisfied

Thus, SOLITAIRE is NP-Complete.

Comment