

# Definitions for Minesweeper board solvability

Thomas Kolar

November 16, 2012

## Abstract

This set of definitions was inspired by the following discussion on the minesweeper forum:

<http://www.minesweeper.info/forum/viewtopic.php?f=15&t=194>

I am creating this as a  $\text{\LaTeX}$  document so that I will be able to use math formatting.

This is - obviously - work in progress, so feel free to provide feedback!

## 1 Motivation

I feel that this discussion would benefit greatly from a couple of formal definitions that can be used to significantly increase both brevity and clarity of our discussion. This document may grow in the future to accomodate concrete results that arise from the definitions contained in it; But for now, it is intended as a collection of definitions only.

## 2 Preliminary definitions

**Definition 1.** A *clone* is a program that is used to play minesweeper.

## 3 Representing Game States as Sets

### 3.1 Visible State versus Hidden State

Two highly different types of game state exist. One is the complete “real” state of the game as known by the clone used; It contains all mine positions,

as well as which squares are open, which squares are flagged, etc. The other, incomplete type of game state is the state of the board as seen by the player. It consists of all information that the clone used exposes to the player.

Obviously, the visible state is implied by the hidden state, but not necessarily vice versa; As a matter of fact, deducing the hidden state from the visible state is the act of solving a minesweeper board.

Therefore, we will state many problems in the form “given a visible state  $X$ , what possible hidden states  $Y$  are there, and what do they have in common?”.

Also, since these definitions are supposed to facilitate reasoning about solvability in principle - as opposed to solving efficiency - considerations like flags or mouse events will be omitted entirely. Instead, we will focus on valid transitions between states, ie, opening single squares without presuming a method of doing so. For instance, the question whether squares are opened by flags or chords remains unanswered, as it is not relevant to our discussion. Similarly, solving time is entirely irrelevant.

**Definition 2.** We define

$$\mathcal{V} := \{0, 1, 2, 3, 4, 5, 6, 7, 8, ?\}$$

to be *the set of single-square visible states*, (? represents an unknown square) and

$$\mathcal{H} := \{*, 1, 0\}$$

to be *the set of single-square hidden states* (Remark: \*,1,0 stand for *mine, visible* and *hidden*, respectively. A square that contains a mine will always be hidden (we do not distinguish between flagged and unflagged squares)).

**Definition 3.** Let

$$H \in \mathcal{H}^{h \times w}$$

be a matrix over  $\mathcal{H}$  so that

$$\sum_{i=1}^h \sum_{j=1}^w \delta_{(H)_{ij},*} = m$$

(ie,  $H$  contains  $m$  mines). Then  $H$  is a *hidden state with height  $h$ , width  $w$  and  $m$  mines*, or, alternatively, a *hidden state with dimension  $(h, w, m)$* . We also write  $\dim H = (h, w, m)$ .

**Definition 4.** Let  $\mathcal{H}^{h \times w \times m}$  be the set of all hidden states  $H$  with  $\dim H = (h, w, m)$ .

**Definition 5.** Let  $H \in \mathcal{H}^{h \times w \times m}$ , and let  $1 \leq i \leq h$ ,  $1 \leq j \leq w$ . Then  $(i, j)$  are *coordinates* of  $H$ .

**Definition 6.** Let  $H \in \mathcal{H}^{h \times w \times m}$ , and let  $(i, j)$  be coordinates of  $H$ .

$$\begin{aligned} \text{adj}_H(i, j) := \{ & (k, l) \mid \max\{1, i-1\} \leq k \leq \min\{h, i+1\}, \\ & \max\{1, j-1\} \leq l \leq \min\{w, j+1\}, \\ & (k, l) \neq (i, j)\} \end{aligned}$$

is the set of *neighbors* of  $(i, j)$ , and

$$\text{num}_H(i, j) := |\{(k, l) \in \text{adj}_H(i, j) \mid (H)_{kl} = *\}|$$

is the *number* of  $(i, j)$ .

**Definition 7.** Let  $\rho_{h,w,m} : \mathcal{H}^{h \times w \times m} \rightarrow \mathcal{V}^{h \times w}$  so that

$$H \mapsto V, (V)_{ij} = \begin{cases} \text{num}_H(i, j) & ((H)_{ij} = 1) \\ ? & (\text{otherwise}) \end{cases}$$

$\rho$  is the *visible representation map*. Remark: Given a hidden state, it will expose exactly the information that a clone would expose to the player.

**Definition 8.** Let

$$\mathcal{V}^{h \times w \times m} := \rho_{h,w,m}(\mathcal{H}^{h \times w \times m})$$

be the set of (*valid*) *visible states*, and let  $V \in \mathcal{V}^{h \times w \times m}$ .

Then  $V$  is a *visible state with height  $h$ , width  $w$  and  $m$  mines*, or, alternatively, a *visible state with dimension  $(h, w, m)$* . We also write  $\dim V = (h, w, m)$ . Remark:  $\dim$  is NOT a map, because the third component is not well-defined. However, this notation will most likely prove useful, since we will assume that we know the total number of mines most of the time.

**Definition 9.** Let  $s \in \mathcal{V} \times \mathcal{H} =: \mathcal{S}$ . Then  $s$  is a *square*.

**Definition 10.** Let  $S = (H, V) \in \mathcal{H}^{h \times w \times m} \times \mathcal{V}^{h \times w \times m}$ . If

$$\rho_{h,w,m}(H) = V$$

then  $S$  is a *game state*. We also define

$$\mathcal{S}^{h \times w \times m} := \{(H, \rho_{h,w,m}(H)) \mid H \in \mathcal{H}^{h \times w \times m}\}$$

**Definition 11.** If a square has visible state  $?$ , we also say that it is *unknown*. Otherwise, we say that it is *known*.