

SAT Encodings for Flood-It Puzzle

Milan van Stiphout

Faculty of Science

Radboud University, Nijmegen, The Netherlands
milanstiphout@gmail.com

Muhammad Osama and Anton Wijs

Department of Mathematics and Computer Science

Eindhoven University of Technology, Eindhoven, The Netherlands
{o.m.m.muhammad, a.j.wijs}@tue.nl

Abstract—Flood-it is a puzzle in which, the player aims to fill the game board (flood) with a minimum number of flooding moves. Finding an optimal solution to this game is an NP-Complete problem. In this paper, we describe a possible encoding of the game to propositional Satisfiability problem.

I. INTRODUCTION

Flood-It is a board game in which the player is given an $n \times n$ board of squares (we call them fields), where each one is allocated one of c colors. The goal is to fill the entire board with the same color via the shortest possible sequence of flooding moves from the top left. If $c \geq 3$, Flood-It complexity becomes NP-Hard [1], [2].

Each move dictates the player to choose a color. In turn, all flooded fields change their color to the chosen color. This procedure is recursive as any field touching the flooded region with the same color is also flooded. As the flooded region grows, new neighbors will be eligible for flooding in the next move. The process continues until the whole board is filled with a single color with lowest possible moves.

II. SAT ENCODING

The intuition of solving the game in SAT, is to let a SAT solver find the shortest sequence of moves that completes the puzzle. In this section, we briefly describe a strategy which is proposed by the first author in his bachelor thesis [3] to formulate the game rules with a set of predicates in Conjunctive Normal Form (CNF).

Nevertheless, solving a single encoding gives only part of the answer to the question “how many minimal moves is needed to solve Flood-It puzzle?”. The minimal solution is rather logarithmically approached by choosing a value m that allows the corresponding SAT formula satisfiable.

A. Game Properties

The properties of the game are expressed in the following predicates:

- $\mathcal{F}(t, f)$ is true if a field f is flooded at turn t .
- $\mathcal{M}(t, c)$ is true if a color c is chosen at turn t .
- $\mathcal{T}(t, f)$ is true if f is next to a flooded field at t .

These predicates define the Boolean variables of the SAT formula that encodes Flood-It puzzle. The assignments of the *move* predicates \mathcal{M} , along with the proposed conditions, give a possible sequence of moves that we need to solve the game.

B. Flood Conditions

The following conditions are designed in such a way to strictly follow the rules of Flood-It. This grants a tangible solution of the puzzle via a satisfying assignment to the corresponding SAT problem.

- 1) A cluster \mathcal{C} of orthogonal fields is flooded at $t = 0$ starting from the top-left-corner field $f = 0$ (precondition).

$$\bigwedge_{f=0}^{n^2-1} \begin{cases} \mathcal{F}(0, f) & , \text{if } f \in \mathcal{C} \\ \neg \mathcal{F}(0, f) & , \text{otherwise} \end{cases} \quad (1)$$

- 2) One move can be made at a turn with exactly one color. The notations m and k denote the maximum number of moves allowed and the number of colors available respectively.

$$\bigwedge_{t=0}^{m-1} \bigvee_{c=0}^{k-1} \mathcal{M}(t, c) \quad (2)$$

$$\bigwedge_{t=0}^{m-1} \bigwedge_{c=0}^{k-2} \bigwedge_{d=c+1}^{k-1} \neg \mathcal{M}(t, c) \vee \neg \mathcal{M}(t, d) \quad (3)$$

- 3) A flooded field will remain flooded to the end of the game

$$\bigwedge_{t=0}^{m-1} \bigwedge_{f=0}^{n^2-1} \mathcal{F}(t, f) \Rightarrow \mathcal{F}(t+1, f) \quad (4)$$

- 4) If a field f is not already flooded and not touched by any other field g_i such that $1 \leq i \leq k$, then f will not be flooded.

$$\bigwedge_{t=0}^m \bigwedge_{f=0}^{n^2-1} \neg \mathcal{F}(t, f) \wedge \neg \mathcal{T}(t, g_1) \wedge \dots \wedge \neg \mathcal{T}(t, g_k) \Rightarrow \neg \mathcal{F}(t+1, f) \quad (5)$$

- 5) If a move is made with an f 's color (c_f), and that field touches the flooded region, it is flooded.

$$\bigwedge_{t=0}^{m-1} \bigwedge_{f=0}^{n^2-1} \mathcal{M}(t, c_f) \wedge \mathcal{T}(t, f) \Rightarrow \mathcal{F}(t+1, f) \quad (6)$$

- 6) If a move is made of not c_f and f is not flooded already, then f is not flooded.

$$\bigwedge_{t=0}^{m-1} \bigwedge_{f=0}^{n^2-1} \neg \mathcal{M}(t, c_f) \wedge \neg \mathcal{F}(t, f) \Rightarrow \neg \mathcal{F}(t+1, f) \quad (7)$$

- 7) Whenever a field is flooded, all neighbors $(g_1 \dots g_k)$ with the same color are flooded as well.

$$\bigwedge_{t=0}^m \bigwedge_{f=0}^{n^2-1} \mathcal{F}(t, f) \Leftrightarrow \mathcal{F}(t, g_1) \vee \dots \vee \mathcal{F}(t, g_k) \quad (8)$$

- 8) A field is flooded iff it touches a direct orthogonal neighbor $(n_1 \dots n_k)$.

$$\bigwedge_{t=0}^m \bigwedge_{f=0}^{n^2-1} \mathcal{T}(t, f) \Leftrightarrow \mathcal{F}(t, n_1) \vee \dots \vee \mathcal{F}(t, n_k) \quad (9)$$

- 9) All fields are flooded after a maximum number of moves m at $t = m + 1$ (postcondition).

$$\bigwedge_{f=0}^{n^2-1} \mathcal{F}(m + 1, f) \quad (10)$$

III. BENCHMARKS

In benchmarks generation, we chose $k = 10, n = 70$, and $m = \{225, 229, 230\}$. The maximum number of moves m can be used to control the hardness of the encoded problem. For these settings, 20 instances are generated randomly.

REFERENCES

- [1] D. Arthur, R. Clifford, M. Jalsenius, A. Montanaro, and B. Sach, “The complexity of flood filling games,” in *Fun with Algorithms*, P. Boldi and L. Gargano, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 307–318.
- [2] U. dos Santos Souza, F. Rosamond, M. R. Fellows, F. Protti, and M. D. da Silva, “The flood-it game parameterized by the vertex cover number,” *Electronic Notes in Discrete Mathematics*, vol. 50, pp. 35 – 40, 2015, IAGOS’15 – VIII Latin-American Algorithms, Graphs and Optimization Symposium.
- [3] M. van Stiphout, “Flood-it as a sat problem,” bachelor thesis, Faculty of Science, Radboud University, Nijmegen, The Netherlands, 2020. [Online]. Available: <https://www.cs.ru.nl/bachelors-theses>