Discrete Applied Mathematics ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Algorithms, kernels and lower bounds for the Flood-It game parameterized by the vertex cover number☆

Michael Fellows [a], Fábio Protti [b], Frances Rosamond [a], Maise Dantas da Silva [b], Uéverton S. Souza [b,*]

[a] University of Bergen, Bergen, Norway
[b] Fluminense Federal University, Niterói, Brazil

ARTICLE INFO

Article history:
Received 30 December 2015
Received in revised form 22 June 2017
Accepted 3 July 2017
Available online xxxx

Keywords:
Flood-It
Flood-filling
Vertex cover
FPT
Kernel
ETH

ABSTRACT

Flood-It is a combinatorial problem on a colored graph whose aim is to make the graph monochromatic using the minimum number of *flooding moves*, relatively to a pivot vertex p. A flooding move consists of changing the color of the monochromatic component (maximal monochromatic connected subgraph) containing p. This problem generalizes a combinatorial game named alike which is played on $m \times n$ grids. It is known that Flood-It is NP-hard even for $3 \times n$ grids and for instances with bounded number of colors, diameter, treewidth, or pathwidth. In [Fellows, Souza, Protti, Dantas da Silva, Tractability and hardness of flood-filling games on trees, Theoretical Computer Science, 576, 102-116, 2015] it is shown that Flood-It is W[1]-hard when played on trees with bounded number of colors, and the number of leaves is a single parameter. Contrasting with such results, in this work we show that Flood-It is fixed-parameter tractable when parameterized by either the vertex cover number or the neighborhood diversity. Additionally, we prove that Flood-It does not admit a polynomial kernel when the vertex cover number is a single parameter, unless $coNP \subseteq NP/poly$. Finally, lower bounds based on the (Strong) Exponential Time Hypothesis as well as an upper bound for the required time to solve Flood-It are also provided.

1. Introduction

Let G be a vertex-colored graph and let $p \in V(G)$ be a pivot vertex of G. A flooding move $m = (p, c)$ in G consists of changing to c the color of p and of all vertices in the monochromatic component (maximal monochromatic connected subgraph) containing p in G. The problem of determining the minimum number of flooding moves to make the graph monochromatic is called *Flood-It*. Fig. 1 shows a sequence of moves to flood a graph.

As shown in [13], Flood-It played on trees is analogous to a restricted case of the *Shortest Common Supersequence Problem* (SCS) [18], where no string has the same symbol in consecutive positions. Consequently, Flood-It inherits many applications from this special case of SCS, such as: microarray production [27], DNA sequence assembly [2], and multiple sequence alignment [28]. In particular, when each symbol occurs at most once in any path from the pivot to a leaf of the tree, each path is analogous to a phylogenetic sequence (see [14]).

☆ This project was partially supported by the FAPERJ-Brazil (grant no. E-26/010.001578/2016) and CNPq-Brazil (grant no. 459051/2014-8).
* Corresponding author.
E-mail addresses: michael.fellows@uib.no (M. Fellows), fabio@ic.uff.br (F. Protti), Frances.Rosamond@uib.no (F. Rosamond), maisedantas@id.uff.br (M.D. da Silva), ueverton@ic.uff.br (U.S. Souza).

http://dx.doi.org/10.1016/j.dam.2017.07.004

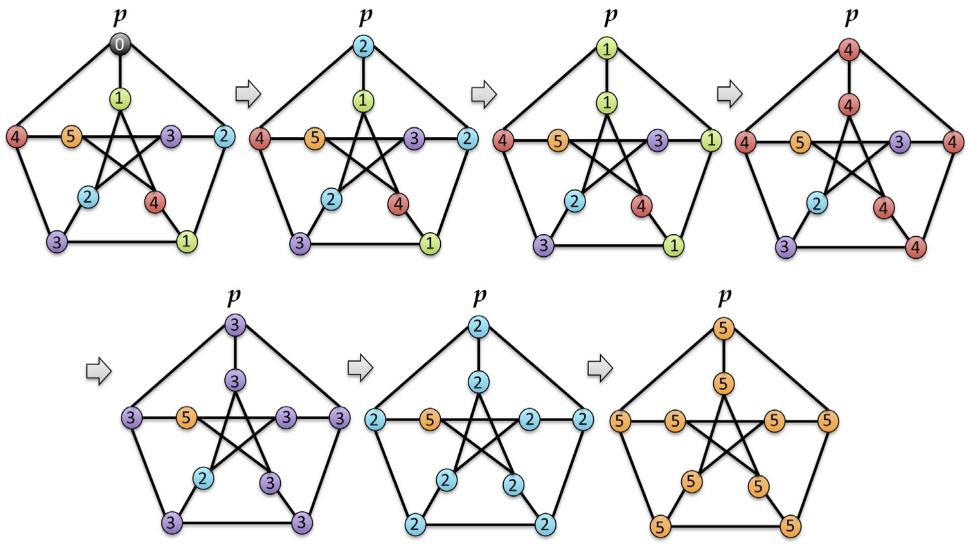

Fig. 1. An optimal sequence of moves to flood a 6-colored Petersen graph.

As described in [13], Flood-It on trees can also be applied to scheduling. Each color corresponds to an operation in a sequential process of manufacturing an object. In the input tree T, paths from the pivot to the leaves correspond to the manufacturing sequences for a number of different objects that share the same production line. A flooding of T then corresponds to a schedule of operations for the production line that allows all of the different objects to be manufactured. Beyond these applications when the underlying graph is a tree, some disease spreading models described in [1] work in a similar way as the Flood-It game.

The computational problem Flood-It is generalization of a combinatorial game named alike, which is originally played on a colored board consisting of an $m \times n$ grid, where each tile of the board has an initial color from a fixed color set. Many complexity issues on Flood-It have recently been investigated. In [9], Clifford, Jalsenius, Montanaro, and Sach show that Flood-It is NP-hard on $n \times n$ grids colored with at least three colors. Meeks and Scott [22] prove that Flood-It remains NP-hard on $3 \times n$ grids colored with at least four colors. Clifford, Jalsenius, Montanaro, and Sach present in [9] a polynomial-time algorithm for Flood-It on $2 \times n$ grids. Regarding the complexity of Flood-It played on general graphs, Fleischer and Woeginger [15] proved that Flood-It (denoted by Honey-Bee-Solitaire) remains NP-hard even when the game is restricted to trees or split graphs, but it is polynomial-time solvable on co-comparability graphs. In [13,29], Fellows, Souza, Protti, and Dantas da Silva show that Flood-It played on trees is analogous to an important subcase of SCS, as indicated earlier in this section. In [30], Souza, Protti and Dantas da Silva describe polynomial-time algorithms to play Flood-It on C_n^2 or P_n^2 (the second power of a cycle or a path on n vertices) and $2 \times n$ circular grids, and Fellows, Souza, Protti, and Dantas da Silva [13] develop a multivariate investigation of the complexity of Flood-It when played on trees, analyzing the complexity consequences of parameterizing flood-filling problems in various ways. Besides that, a variant of Flood-It, where in each move the player can choose a new pivot vertex (so-called Free-Flood-It) was also studied in [9,13,20,23,24,30].

Flood-It remains NP-hard even assuming constant values for: number of colors [9]; diameter [29]; or treewidth [15]. In [13,29], Fellows, Souza, Protti, and Dantas da Silva show parameterized complexity results on Flood-It on trees; for instance, Flood-It on trees is W[1]-hard when parameterized by the aggregate parameter (number of leaves, number of colors). Therefore, finding interesting parameters for which Flood-It is fixed-parameter tractable seems to be a challenge. The main goal of this paper is to analyze the parameterized complexity of Flood-It when parameterized by the vertex cover number.

Results. We describe an FPT-algorithm for Flood-It with either the vertex cover number or the neighborhood diversity as a single parameter, and we present a polynomial kernelization algorithm when the neighborhood diversity and the number of colors of the input graph form an aggregate parameter. In addition, we show the following results: Flood-It does not admit polynomial kernel when the vertex cover number is a single parameter, unless $coNP \subseteq NP/poly$; no $2^{o(k+i_c)}n^{\mathcal{O}(1)}$ time algorithm for Flood-It is possible unless the Exponential Time Hypothesis (ETH) fails; and no $(2-\varepsilon)^{i_c}n^{\mathcal{O}(1)}$ time algorithm for Flood-It exists unless the Strong Exponential Time Hypothesis (SETH) fails, where k is the cardinality of a minimum vertex cover and i_c is the number of colors of a maximum independent set. On the other hand, an $\mathcal{O}(2^{\mathcal{O}(k\log(i_c\, k))}n^{\mathcal{O}(1)})$ algorithm for Flood-It is provided.

Definitions and notation.

- Two vertices a and b are *m-connected* when there is a monochromatic path between them.
- A subgraph H is said to be *flooded* when H becomes monochromatic.