Discrete Applied Mathematics 170 (2014) 33–45

Contents lists available at ScienceDirect

Discrete Applied Mathematics

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

Brushing without capacity restrictions

Darryn Bryant [a], Nevena Francetić [b], Przemysław Gordinowicz [c], David A. Pike [d], Paweł Prałat [e,*]

[a] The University of Queensland, Department of Mathematics, QLD 4072, Australia
[b] School of Mathematics & Statistics, Carleton University, Ottawa, ON, Canada
[c] Institute of Mathematics, Technical University of Lodz, Łódź, Poland
[d] Department of Mathematics and Statistics, Memorial University of Newfoundland, St. John's, NL, Canada
[e] Department of Mathematics, Ryerson University, Toronto, ON, Canada

ARTICLE INFO

Article history:
Received 19 November 2012
Received in revised form 22 January 2014
Accepted 30 January 2014
Available online 20 February 2014

Keywords:
Network cleaning
Brush number
Directed path covering

ABSTRACT

In graph cleaning problems, brushes clean a graph by traversing it subject to certain rules. Various problems arise, such as determining the minimum number of brushes that are required to clean the entire graph. This number is called the brushing number. Here, we study a new variant of the brushing problem in which one vertex is cleaned at a time, but more than one brush may traverse a dirty edge. In particular, we obtain results on the brushing number of Cartesian products of graphs and trees, as well as upper and lower bounds on the brushing number in the general case.

1. Introduction

In this paper, we study a natural variant of the graph cleaning problem that was introduced by McKeil [9] in which all edges and vertices of a graph are initially considered to be contaminated. Cleaning agents called brushes travel throughout the graph, decontaminating as they go. Once each vertex has been visited, and each edge has been traversed by a brush, the graph has then been cleaned, although in certain models recontamination may also occur. The model considered in Chapter 3 of [9] permits edges to be traversed by more than one brush at a time, and also permits edges to be traversed on multiple occasions.

In [10], restrictions are imposed whereby only dirty edges can be traversed, and each edge can be traversed by at most one brush. This model corresponds to the minimum total imbalance of the graph which is used in the graph drawing theory [4], and is well studied, especially when it is performed on random graphs [1,13,17]. (See also [8] for the algorithmic side, [11,16] for a related model of cleaning with Brooms, [12] for the relationship with other elimination schemes, and a combinatorial game [6].) In the present paper we relax one of these restrictions and allow dirty edges to be traversed by multiple brushes, although we retain the condition that an edge can be traversed on only one occasion.

Having been inspired by chip firing processes [2], the manner in which brushes disperse from an individual vertex is such that they do so in unison, provided that their vertex meets the criteria to fire. Models in which multiple vertices may fire simultaneously are called parallel cleaning models (see [5] for more details). In contrast, sequential parallel models mandate that vertices fire one at a time. The variant considered in [10] and the one we consider in this paper are sequential in nature.

* Corresponding author. Tel.: +1 416 979 5000; fax: +1 416 598 5917.
E-mail addresses: db@maths.uq.edu.au (D. Bryant), nfrancet@uottawa.ca (N. Francetić), pgordin@p.lodz.pl (P. Gordinowicz), dapike@mun.ca (D.A. Pike), pralat@ryerson.ca (P. Prałat).

http://dx.doi.org/10.1016/j.dam.2014.01.024

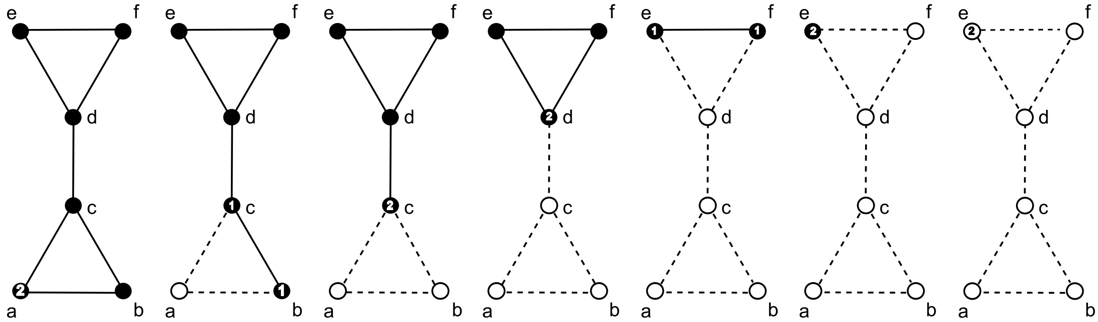

Fig. 1. A graph G with an initial configuration of 2 brushes.

When considering a graph or network, the central question under investigation is that of determining the brush number for the graph (i.e., the minimum number of brushes that enable the graph to be cleaned), as well as to describe a corresponding cleaning strategy. Subsequent to presenting a more formal definition of our model in Section 2, we establish some general bounds for the brush number of an arbitrary graph, including bounds that are expressed in terms of parameters such as cutwidth and bisection width. We then investigate two specific classes of graphs, namely Cartesian products and trees.

For Cartesian products we prove a general upper bound on the brush number, and then establish exact values for the brush numbers of m by n grids and hypercubes. For trees, we prove that if a tree T has $d_\ell(T)$ vertices of degree 1, then exactly $(d_\ell(T)+1)/2$ brushes are required for an optimal cleaning strategy when $d_\ell(T)$ is odd. If $d_\ell(T)$ is even, then the minimum number of brushes that are required is either $d_\ell(T)/2$ or $d_\ell(T)/2+1$, which is to say that trees with an even number of vertices of degree 1 are partitioned into two sets depending on whether their brush number equals $d_\ell(T)/2$ or exceeds it by 1.

2. Definitions

As already noted, the graph cleaning model we consider here differs from the one presented in [10] in that we allow edges to be traversed by multiple brushes. Before we define the model rigorously, we present a simple example illustrated in Fig. 1.

In this example, all edges and vertices are initially dirty and we place two brushes at vertex a. As vertex a contains at least as many brushes as dirty incident edges, it is able to fire. Thus, at Step 1 vertex a is cleaned and a brush is sent down each dirty edge: edges ab and ac have been cleaned; vertices b and c have one fewer dirty incident edges. At Step 2, vertices c, d, e and f cannot be cleaned as they each have fewer brushes than dirty incident edges; vertex b is cleaned instead and one brush is sent to c. At Step 3, the only vertex ready to be cleaned is vertex c. It is at this point that the cleaning process considered in this paper differs from the process described in [10]. In [10], only one brush is permitted to traverse an edge. So one brush would be moved from c to d, whilst the other would remain at c. With our variant of the cleaning process, more than one brush can be moved through a dirty edge. Thus, the two brushes are moved from c to d (no advantage can be gained by leaving a brush behind). At Step 4, d is cleaned and the remaining two vertices (e and f) can be cleaned at the next two steps (although all edges are clean prior to the final step). Under our cleaning model this example graph can be cleaned with just two brushes, whereas under the model considered in [10] three brushes would be required.

Now we formally define the cleaning process we are considering. Let $G=(V,E)$ be any finite, undirected graph. The initial configuration of brushes is given by the function $\omega_0 : V \to \mathbb{N}_0$, where $\mathbb{N}_0=\{0,1,2,\ldots\}$, $\omega_0(v)$ is the number of brushes initially at vertex v, and all vertices and edges of the graph are initially dirty. At each step t of the process, $\omega_t(v)$ denotes the number of brushes at vertex $v \in V$, and $D_t \subseteq V$ denotes the set of dirty vertices. An edge $uv \in E$ is dirty if and only if both u and v are dirty; that is, $\{u,v\} \subseteq D_t$. Finally, let $D_t(v)$ denote the number of dirty edges incident to v at step t; that is,

$$D_t(v) = \begin{cases} |N(v) \cap D_t| & \text{if } v \in D_t \\ 0 & \text{otherwise} \end{cases}$$

(where $N(v)$ denotes, as usual, the neighbourhood of v).

Definition 2.1. The **(generalized) cleaning process** $\mathfrak{P}(G,\omega_0)=\{(\omega_t,D_t)\}_{t=0}^{T}$ of an undirected graph $G=(V,E)$ with an **initial configuration of brushes** ω_0 is as follows:

⓪ Initially, all vertices are dirty: $D_0=V$; set $t=0$.

① Let α_{t+1} be any vertex in D_t such that $\omega_t(\alpha_{t+1}) \geq D_t(\alpha_{t+1})$. If no such vertex exists, then stop the process (set $T=t$), return the **cleaning sequence** $\alpha=(\alpha_1,\alpha_2,\ldots,\alpha_T)$, the **final set of dirty vertices** D_T, the **final configuration of brushes** ω_T, and the **distribution of brushes** δ.