Theoretical Computer Science ••• (••••) •••–•••

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Constructing an indeterminate string from its associated graph ☆

Joel Helling [a], P.J. Ryan [b], W.F. Smyth [b,c,1], Michael Soltys [a,*]

[a] Department of Computer Science, California State University Channel Islands, United States
[b] Algorithms Research Group, Dept. of Computing and Software, McMaster University, Canada
[c] School of Engineering and Information Technology, Murdoch University, Australia

ARTICLE INFO

Article history:
Received 21 May 2016
Received in revised form 2 February 2017
Accepted 14 February 2017
Available online xxxx

Keywords:
String algorithms
Indeterminate strings
Cliques
Graph labeling

ABSTRACT

As discussed at length in Christodoulakis et al. (2015) [3], there is a natural one-many correspondence between simple undirected graphs $\mathcal{G}$ with vertex set $V = \{1, 2, \ldots, n\}$ and ***indeterminate strings*** $\boldsymbol{x} = \boldsymbol{x}[1..n]$ — that is, sequences of subsets of some alphabet Σ. In this paper, given $\mathcal{G}$, we consider the "reverse engineering" problem of computing a corresponding $\boldsymbol{x}$ on an alphabet $\Sigma_{\min}$ of minimum cardinality. This turns out to be equivalent to the NP-hard problem of computing the ***intersection number*** of $\mathcal{G}$, thus in turn equivalent to the ***clique cover*** problem. We describe a heuristic algorithm that computes an approximation to $\Sigma_{\min}$ and a corresponding $\boldsymbol{x}$. We give various properties of our algorithm, including some experimental evidence that on average it requires $\mathcal{O}(n^2 \log n)$ time. We compare it with other heuristics, and state some conjectures and open problems.

1. Introduction

In this paper we seek to extend the connections between graph theory and stringology explored in [3]. We consider a ***string*** $\boldsymbol{x} = \boldsymbol{x}[1..n]$ to be a sequence of ***letters*** $\boldsymbol{x}[i], 1 \le i \le n$, that are nonempty subsets of a given finite set Σ, called the ***alphabet***. If $\boldsymbol{x}[i]$ is a subset of cardinality 1, it is said to be a ***regular*** letter; otherwise, ***indeterminate***. Similarly, if $\boldsymbol{x}$ contains only regular letters, it is said to be ***regular***; otherwise, ***indeterminate***. For example, on $\Sigma = \{a, b, c\}$, $\boldsymbol{x} = ababc$ is regular,[2] while $\boldsymbol{y} = \{a, b\}ba\{b, c\}b$ is indeterminate. Indeterminate strings are useful in various application areas, notably bioinformatics, where under certain circumstances DNA sequences can be regarded as indeterminate strings on nucleotides $\{a, c, g, t\}$. In recent years indeterminate strings have been the subject of much study [12,11,17,1].

Given string $\boldsymbol{x} = \boldsymbol{x}[1..n]$, we say that for $1 \le i, j \le n$, $\boldsymbol{x}[i]$ ***matches*** $\boldsymbol{x}[j]$ (written $\boldsymbol{x}[i] \approx \boldsymbol{x}[j]$) if and only if $\boldsymbol{x}[i] \cap \boldsymbol{x}[j] \neq \emptyset$. Thus in particular $\boldsymbol{x}[i] = \boldsymbol{x}[j] \Longrightarrow \boldsymbol{x}[i] \approx \boldsymbol{x}[j]$. As defined in [3], the ***associated graph*** $\mathcal{G}_{\boldsymbol{x}} = (V_{\boldsymbol{x}}, E_{\boldsymbol{x}})$ of $\boldsymbol{x}$ is the simple graph whose vertices are positions $1, 2, \ldots, n$ in $\boldsymbol{x}$ and whose edges are the pairs (i, j) such that $\boldsymbol{x}[i] \approx \boldsymbol{x}[j]$. Suppose that for some position $i_0 \in 1..n$, $\boldsymbol{x}[i_0]$ matches $\boldsymbol{x}[i_1], \boldsymbol{x}[i_2], \ldots, \boldsymbol{x}[i_k]$ for some $k \ge 0$, and matches no other elements of $\boldsymbol{x}$. We say that position i_0 is ***essentially regular*** if and only if the entries in positions $i_1, i_2, \ldots, i_k$ match each other pairwise. If every

☆ We are grateful to Manolis Christodoulakis and Shu Wang for helpful discussions.
* Corresponding author.
E-mail addresses: joel.helling904@csuci.edu (J. Helling), ryanpj@mcmaster.ca (P.J. Ryan), smyth@mcmaster.ca (W.F. Smyth), michael.soltys@csuci.edu (M. Soltys).
1 Supported in part by a Grant No. 14284 from the Natural Sciences and Engineering Research Council of Canada.
2 For singleton sets such as $\{a\}, \{b\}, \{c\}$, we write a, b, c for simplicity.

http://dx.doi.org/10.1016/j.tcs.2017.02.016
0304-3975/

position in $\boldsymbol{x}$ is essentially regular, we say that $\boldsymbol{x}$ itself is ***essentially regular***. Hence every essentially regular string can be replaced by an equivalent regular one, and the associated graph $\mathcal{G}_{\boldsymbol{x}}$ is a collection of disjoint cliques if and only if $\boldsymbol{x}$ is essentially regular.

For general indeterminate strings, however, $\mathcal{G}_{\boldsymbol{x}}$ is more interesting. In Section 2 we discuss a conjecture stated in [3], that given a finite simple graph $\mathcal{G}$ whose maximal cliques have basis $\mathcal{B}$, $|\mathcal{B}|$ is the minimum alphabet size of any string $\boldsymbol{x}$ whose associated graph $\mathcal{G}_{\boldsymbol{x}} = \mathcal{G}$. We discover that this conjecture is just a reformulation, in a slightly different context, of the problem of computing the intersection number of $\mathcal{G}$, which is NP-hard, and hence computing the minimum alphabet size of any string is also NP-hard. Section 3 describes an algorithm that approximates a basis of $\mathcal{G}$ by assigning symbols to the vertices of cliques until all vertices are labeled, thus effectively computing a string $\boldsymbol{x}$ whose associated graph $\mathcal{G}_{\boldsymbol{x}} = \mathcal{G}$. This is an example of the "reverse engineering" of a data structure, a class of problems initiated in [8,7] for the border array, and extended to other structures in, for example, [2,9,4]. In Section 4 we discuss our algorithm's results and execution, especially in the context of other algorithms that perform closely-related computations. Section 5 discusses a few conjectures and open problems.

2. Maximal cliques in the associated graph $\mathcal{G}_{\boldsymbol{x}}$

Suppose a collection $\mathcal{F} = F_1, F_2, \ldots, F_n$ of sets is given. Then the ***intersection graph*** $\mathcal{G}_{\mathcal{F}}$ of $\mathcal{F}$ is a simple undirected graph on $|\mathcal{F}| = n$ vertices $1, 2, \ldots, n$, with an edge (i, j), $1 \le i, j \le n$, if and only if $F_i \cap F_j \ne \emptyset$. Conversely, it was shown in [18] that, given a simple undirected graph $\mathcal{G}$ on vertices $1, 2, \ldots, n$, a collection $\mathcal{F}$ of n sets can be found such that $\mathcal{G}$ is the intersection graph of $\mathcal{F}$. (For example, for each (i, j) in $\mathcal{G}$, $i < j$, place a unique symbol $\lambda_{i,j}$ in F_i and F_j.) The ***intersection number*** $\theta(\mathcal{G})$ of $\mathcal{G}$ is the smallest number of distinct symbols that can be placed in the sets of $\mathcal{F}$ such that $\mathcal{G} = \mathcal{G}_{\mathcal{F}}$. In our application, the collection $\mathcal{F}$ becomes a string $\boldsymbol{x} = \boldsymbol{x}[1..n]$ with $\boldsymbol{x}[i] = F_i$ (necessarily nonempty). The associated graph and the intersection graph are thus the same, and we seek a smallest alphabet $\Sigma_{\min}$ that produces it. Let $\sigma_{\min} = |\Sigma_{\min}|$, the cardinality of such an alphabet.

The standard way of efficiently representing a finite simple graph $\mathcal{G}$ as an intersection graph is by covering the graph by cliques. Take any set of cliques covering all edges of $\mathcal{G}$. For each vertex v, let F_v be the set of those cliques containing the vertex v. Then the intersection graph of $\{F_v\}$ coincides with $\mathcal{G}$. As a result, the intersection number $\theta(\mathcal{G})$ is equal to the ***edge clique cover number*** $ec(\mathcal{G})$, the cardinality of a minimum size set of the cliques that covers all the edges of $\mathcal{G}$. Erdős et al. [6,16] proved that $\theta(\mathcal{G}) \le \lfloor n^2/4 \rfloor$, an upper bound that is achieved when $\mathcal{G}$ is a triangle-free graph on $\lfloor n^2/4 \rfloor$ edges [15]. An instructive example is given by the complete bipartite graphs $K_{m,m}$ (for even $n = 2m$) and $K_{m,m+1}$ (for odd $n = 2m + 1$) for which the minimal covering by cliques consists of all edges, the number of which is precisely $\lfloor n^2/4 \rfloor$.

Erdős et al. use coverings that cover all vertices as well as all edges. If $\mathcal{G}$ has no isolated points, this is equivalent to the "edge covering" approach discussed above. For this case, they prove that $ec(\mathcal{G}) \le \lfloor n^2/4 \rfloor$ and that one need only use 2-cliques and 3-cliques (edges and triangles) in a minimal covering.[3]

More recently, Conjecture 21 in [3] formulated the problem in a slightly different way, in terms of the ***maximal*** cliques in $\mathcal{G}$; that is, those that are not proper subgraphs of any other clique. We provide here a proof of this conjecture, thus validating the several following remarks made in that paper. To be consistent with [3], we use the notion of ***basis***. A basis is a minimum size set of maximal cliques that covers all edges and all vertices of $\mathcal{G}$.

Lemma 1. *Suppose that a finite simple graph $\mathcal{G}$ with vertex set $V = \{1, 2, \ldots, n\}$ has a basis $\mathcal{B}$ of maximal cliques of cardinality $\sigma_{\min}$. Then there is a string $\boldsymbol{x}$ on a base alphabet of size $\sigma_{\min}$ whose associated graph $\mathcal{G}_{\boldsymbol{x}} = \mathcal{G}$. No string on a smaller alphabet has this property.*

Proof. Let $\mathcal{B} = \{C_1, C_2, \ldots, C_\sigma\}$. Let $\{\lambda_s\}_{s=1}^{\sigma}$ be distinct letters. We construct a string $\boldsymbol{x}$ as follows. For each ordered pair (s, i) with $1 \le s \le \sigma$ and $1 \le i \le n$, assign λ_s to $\boldsymbol{x}[i]$ if vertex i occurs in the maximal clique C_s. It is clear from the definitions that the string $\boldsymbol{x}$ so constructed satisfies $\mathcal{G}_{\boldsymbol{x}} = \mathcal{G}$.

Now consider any string $\boldsymbol{x}$ of length n for which $\mathcal{G}_{\boldsymbol{x}} = \mathcal{G}$ and let τ be the number of distinct (ordinary) letters occurring in $\boldsymbol{x}$. For each such letter λ, there is a clique C_λ of $\mathcal{G}$ whose vertices are those i for which $\lambda \in \boldsymbol{x}[i]$. Of course, these cliques may not be maximal, but each C_λ can be extended to a maximal clique C'_λ. Note that every vertex and edge of $\mathcal{G}$ occurs in one of the cliques C_λ and *a fortiori* in one of the maximal cliques C'_λ. However, the C'_λ might not all be distinct. Let τ' be the number of distinct C'_λ. Then $\tau \ge \tau' \ge \sigma$, the latter inequality following from the fact that there is a basis of cardinality $\sigma_{\min}$. This shows that τ cannot be less than $\sigma_{\min}$ and completes the proof. □

It turns out that maximality is irrelevant in the specification of basis. Let $\phi'(\mathcal{G})$ be the cardinality of a basis (of maximal cliques) in $\mathcal{G}$ and let $\phi(\mathcal{G})$ be the cardinality of a smallest set of cliques that cover all edges and vertices of $\mathcal{G}$. Then:

Observation 2. $\phi'(\mathcal{G}) = \phi(\mathcal{G})$.

[3] The authors thank an anonymous referee who drew their attention to the available material on intersection graphs and clique edge covers.

Please cite this article in press as: J. Helling et al., Constructing an indeterminate string from its associated graph, Theoret. Comput. Sci. (2017), http://dx.doi.org/10.1016/j.tcs.2017.02.016